**Mémoire**

présenté en vue de l'obtention de

# L'Habilitation à Diriger des Recherches

Université Paul Sabatier Toulouse 3

Spécialité : Mathématiques Appliquées

par

**Didier Auroux**

# Algorithmes rapides pour le traitement d'images et l'assimilation de données

Soutenue le 26 Novembre 2008

**Après avis de :**

| | |
|---|---|
| G. Aubert, Professeur | Université de Nice Sophia-Antipolis |
| P. Rouchon, Professeur | Mines ParisTech |
| F. Santosa, Professeur | University of Minnesota |

**Devant le jury composé de :**

| | | |
|---|---|---|
| G. Aubert, Professeur | Université de Nice Sophia-Antipolis | *(Rapporteur)* |
| J. Blum, Professeur | Université de Nice Sophia-Antipolis | *(Examinateur)* |
| L. Cohen, Directeur de Recherche | CNRS & Université Paris Dauphine | *(Examinateur)* |
| P. Degond, Directeur de Recherche | CNRS & Université Paul Sabatier | *(Examinateur)* |
| M. Masmoudi, Professeur | Université Paul Sabatier, Toulouse | *(Coordinateur)* |
| J.-P. Puel, Professeur | Université de Versailles Saint-Quentin | *(Examinateur)* |
| P. Rouchon, Professeur | Mines ParisTech | *(Rapporteur)* |

*“Tout le monde savait que c’était impossible.*
*Il est venu un imbécile qui ne le savait pas et qui l’a fait.”*
Marcel Pagnol (dans : *Papillotes Révillon*)

# Remerciements

Je tiens à remercier en tout premier lieu Gilles Aubert, Pierre Rouchon et Fadil Santosa, qui ont consacré une partie de leur temps à l’étude de mes travaux de recherche en acceptant de rapporter sur ce mémoire d’habilitation. Ils m’ont fourni par la même occasion de nombreuses pistes de recherche, toutes plus intéressantes les unes que les autres.

Je remercie également Laurent Cohen et Jean-Pierre Puel, qui ont accepté avec bienveillance de faire partie du jury et d’étudier avec attention mon travail.

Je suis très reconnaissant à Pierre Degond pour m’avoir fait l’amitié de participer à mon jury.

Je souhaite maintenant exprimer toute ma gratitude et reconnaissance (quasi-éternelle) aux deux personnes qui ont rendu possible ce travail : Jacques Blum et Mohamed Masmoudi. Je connais le premier depuis dix ans maintenant, il m’a formé et fait découvrir les problèmes inverses et l’assimilation de données lors de ma thèse sous sa direction. Cinq années ont passé depuis ma soutenance de thèse et je les ai passées à étudier le traitement d’images auprès du second, qui m’a montré tous les jours à quel point ces problèmes étaient difficiles pour lui et source de bonheur pour moi (ou l’inverse, peut-être). Leurs vastes activités de recherche et leur détermination impressionante à résoudre les problèmes (aussi bien théoriques qu’industriels) en font des directeurs de recherche et collaborateurs inestimables (quand ils n’essaient pas de se débarasser de moi, ou quand ils ne sont pas enfermés dans leur bureau).

Une partie importante de ce travail n’aurait pas été possible sans plusieurs autres collaborateurs (et néanmoins amis). En premier lieu, Lamia Jaafar Belaid, avec qui il est toujours facile et agréable de travailler, à Toulouse comme à Tunis, et grâce à qui je me sens un peu comme chez moi au LAMSIN. Je suis également ravi d’avoir travaillé avec Maëlle Nodet, après tant d’années passées ensemble aux quatre coins du sud-est de la France… J’espère que ce n’est là que le début d’une longue collaboration. Merci à Jérôme Fehrenbach, avec qui les discussions nombreuses et variées dans le couloir ont fini par engendrer une collaboration passionnante. Merci aussi à Silvère Bonnabel, qui a contribué à me faire découvrir la théorie des observateurs. Enfin, j’ai pris énormément de plaisir à travailler avec Patrick Bansart, Sébastien Marinesque, Thomas Migliore, Badreddine Rjaibi, Edith Taillefer, et Thierry Touya pendant leur thèse (terminée, en cours, ou tout juste commencée).

Certaines personnes ont rendu ces quelques années toulousaines particulièrement agréables par leur disponibilité, leur bonne humeur et surtout leur amitié : Marjolaine Puel (ma jumelle de poste, qui m'a nourri un nombre incalculable de mercredis soirs, et que j'ai appris à ne pas contrarier), Pierre Raphaël (même s'il est un assez mauvais supporter de l'OL, nous avons souvent refait le monde autour d'une bonne bouteille) et leurs deux magnifiques enfants Paul et Maud, Guillaume Chèze (mon partenaire de blagues en tous genres, tondeuse, tête de cerf, et plus si affinités) et son harem (Véronique, Émèlie et Éloïse), Luca Amodei (dont j'ai rompu la tranquilité au bureau, malgré son passé trouble), Jean-Pierre Dedieu (fournisseur officiel de miel et contrepéteur professionnel), Pierre Maréchal (qui a désormais peur des cartons et des clés sur les portes), Jean-Michel Roquejoffre (le doyen du labo, que je regrette de ne pas avoir connu quand il était jeune), Jean-Claude Yakoubsohn (qui nous a régalé jusqu'à Hong-Kong avec le 4408), Jean-Pierre Raymond (son plus beau coup : le chien de Mohamed), Komla Domelevo (probablement commercial chez Apple dans une autre vie), Luc Mieussens (pour ses précieux conseils), Marie-Hélène Vignal (merci de ne pas avoir accouché dans mon bureau !), Naoufel Ben Abdallah (pour avoir annoncé le colloque du 1er Avril), Eric Lombardi (champion de vitesse toutes catégories au 205), Marcela Szopos (à qui je dois certainement plein d'heures de TD à l'IUT), Michel Fournié (SOS dépannage bricolage), quelques Mat(t)hieu, les FARC pour m'avoir libéré à temps, Bianca pour ses gâteaux au chocolat, et plein d'autres encore, parmi lesquels de nombreux collègues du département GMP de l'IUT.

Un grand merci à tout le personnel administratif de l'équipe (ex-laboratoire) MIP de l'Institut de Mathématiques de Toulouse, d'une compétence (sauf pour les fournitures de bureau, j'attends toujours le baby-foot) et disponibilité rares, que j'ai embêté tant de fois à la dernière minute : Christine et Marie-Louise, Valérie, Marie-Line, Sarah, Françoise, Janani, Zohra, Anne, Marie-Laure, Delphine, Yannick. Par la même occasion, je remercie les membres de la cellule informatique, ainsi que Monique qui exporte sa bonne humeur permanente jusqu'au 3ème étage.

Mes remerciements s'adressent également à mes collègues de l'équipe-projet MOÏSE de l'INRIA Grenoble - Rhône-Alpes qui m'accueillent régulièrement (surtout en période hivernale), et tout particulièrement à Antoine, Arthur, Innocent, Maëlle et Olivier, à FX pour m'avoir laissé un coin de table, ainsi qu'au grand chef (que dis-je, notre guide spirituel !) Éric et l'irremplaçable Imma.

Merci en vrac à Alexis, Jacqueline, Alexandre et Aurélien pour l'hébergement 5$*$, à Caroline, Céline et Florence pour leur amitié et amour, à mes frères, mes parents, et ma grand-mère, irremplaçable.

Je finirai par mes sponsors principaux : Air France (une bonne dizaine de tours du monde depuis que je suis à Toulouse), différentes marques de café sans qui je n'aurais pas pu travailler avant 15h du matin, et l'imprimante Dell3.

# Table des matières

# Chapitre 1

# Introduction

Plusieurs thématiques ont été étudiées dans le cadre de ce travail, avec le point commun et la volonté de fournir des algorithmes efficaces, robustes, faciles à implémenter et particulièrement rapides et performants. La rapidité des algorithmes est rendue nécessaire par le contexte généralement opérationnel des méthodes, et par un besoin de traiter de plus en plus d'informations en un temps de plus en plus court. L'autre contrainte que nous avons particulièrement prise en compte est la facilité d'utilisation et de mise en œuvre des méthodes développées.

Dans le chapitre 2, nous aborderons différents problèmes de traitement d'images par une approche originale dans le domaine : l'analyse asymptotique topologique, ou plus simplement le gradient topologique.

Le traitement d'images connaît à l'heure actuelle un nouvel éclairage, grâce d'une part aux nouvelles technologies de télécommunication et de diffusion de l'information, qui reposent désormais sur l'envoi et la réception de flux massifs de données numériques sous forme d'images, et d'autre part au monde médical, dans lequel de très grandes avancées ont été réalisées, notamment pour la détection précoce de tumeurs, grâce à de nouvelles techniques d'imagerie plus performantes.

Notre étude est motivée par plusieurs constatations. La première est que le gradient topologique est généralement utilisé pour des problèmes de mécanique des structures, design, conception et optimisation de formes. Il a également été appliqué avec succès en électromagnétisme pour la détection de fissures ou d'objets enfouis. Or de très nombreuses problématiques en traitement d'images reposent sur l'identification de formes, par exemples les contours ou un objet caractéristique de l'image. Ce point commun nous a paru intéressant, et nous a permis d'adapter la méthode du gradient topologique initialement utilisée pour la détection de fissures, à différents problèmes d'imagerie (restauration, classification, segmentation, inpainting).

Le deuxième aspect qui nous a paru intéressant est la rapidité de la méthode. L'analyse asymptotique topologique a permis dans de très nombreux domaines d'obtenir des résultats extrêmement rapidement. Or que ce soit dans l'imagerie médicale ou dans la diffusion audiovisuelle grand public (par exemple la télévision par satellite ou internet), le temps de traitement doit être négligeable pour ne pas retarder respectivement le diagnostic médical ou la diffusion du flux. Il est donc important d'apporter une réponse extrêmement rapide à ces différentes questions, en temps réel pour des films et en un temps négligeable (inférieur à la seconde) pour une image.

Comme nous le verrons par la suite, la méthode du gradient topologique a effectivement pu s'adapter parfaitement au traitement d'images, permettant d'obtenir des résultats très intéressants, et pour un coût de calcul particulièrement faible.

Dans le chapitre 3, nous nous intéresserons à l'assimilation de données pour les problèmes géophysiques et environnementaux, et plus particulièrement dans le cadre de l'atmosphère et des océans. Depuis plusieurs années, une des préoccupations majeures consiste à améliorer sensiblement la connaissance de ces systèmes réputés turbulents, l'un des buts étant de pouvoir prédire leur évolution à plus ou moins court terme avec une grande fiabilité.

Depuis la prévision météorologique grand public pour les prochains jours, jusqu'à l'étude du réchauffement climatique, en passant par la détection de phénomènes climatiques extrêmes plusieurs semaines à l'avance, les enjeux sont sensiblement les mêmes, et se résume au problème suivant. Il s'agit d'estimer rapidement et avec une très grande précision l'état d'un système turbulent, à partir d'une part d'équations mathématiques qui essayent de modéliser les phénomènes physiques régissant le système atmosphère-océans, et d'autre part d'observations de différente nature (in situ et satellitaires), portant sur différentes quantités physiques, et éparpillées dans le temps et l'espace.

Au delà de la taille extrême du problème à traiter (plusieurs milliards de valeurs à estimer, à partir de centaines de millions d'observations) et de fait du temps de calcul nécessaire pour le résoudre, un autre facteur rentre en jeu : le coût humain de développement et d'utilisation d'une méthode d'assimilation de données. À l'heure actuelle, il est extrêmement difficile de mettre en œuvre une telle méthode, même sur un problème relativement simple. Il nous a donc paru intéressant d'étudier la possibilité d'améliorer une des méthodes les plus simples d'assimilation de données, le nudging (ou relaxation newtonienne), afin d'obtenir de bien meilleurs résultats sans toutefois compliquer la méthode.

En appliquant la méthode du nudging au problème rétrograde en temps, nous avons constaté qu'il est possible de stabiliser le système rétrograde, a priori instable du fait de l'irréversibilité du problème physique. Ainsi, comme l'étude présentée au chapitre 3 le montre, nous pouvons remonter le temps, et récupérer une estimation plus fiable du système à un instant passé, à partir duquel des prévisions peuvent être déduites. En appliquant alternativement et itérativement la méthode du nudging standard au modèle direct et au modèle rétrograde en temps, nous obtenons un algorithme itératif extrêmement simple à mettre en œuvre, et qui fournit des résultats nettement meilleurs que le nudging simple. En effet, les résultats sont comparables en qualité et obtenus souvent beaucoup plus rapidement qu'en utilisant la méthode variationnelle d'assimilation de données.

Le chapitre 4 présente une étude à l'interface de ces deux disciplines, la problématique étant l'assimilation de données images. À l'heure actuelle, une très grande quantité d'observations provenant d'images satellitaires n'est quasiment pas utilisée pour améliorer la connaissance de l'état du système. Pourtant, sur les séquences d'images ainsi obtenues, on peut voir très nettement certaines structures caractéristiques (cyclones, tourbillons, courants d'eau chaude, pollution, ...) évoluer et se déplacer au fil du temps.

Plusieurs approches peuvent être considérées pour résoudre ce type de problème, et nous avons fait le choix d'essayer d'identifier et extraire des champs de vitesse à partir de séquences d'images. Cela nous a paru être le choix le plus approprié pour à la fois extraire rapidement des données conventionnelles (car directement reliées aux variables du modèle), et pouvoir les utiliser ensuite dans un système d'assimilation standard.

L'idée que nous développons au chapitre 4 repose sur la méthode du flot optique, qui consiste à chercher un champ de déplacement qui envoie une image sur une autre. L'originalité de notre approche réside dans la non linéarisation de la fonctionnelle à minimiser, combinée à une façon rapide d'assembler la matrice jacobienne. Une approche multi-grille permet enfin de garantir la qualité du minimum. Grâce à tout cela, nous arrivons à extraire des champs de vitesse complets en un temps très court, et il est également possible de fournir un estimateur de la qualité du champ de vitesse obtenu, ce qui peut être vu comme une information sur les statistiques d'erreur de ces pseudo-observations dans le cadre de l'assimilation de données.

Enfin quelques conclusions générales et perspectives de recherche sont données au chapitre 5.

# Chapitre 2

# Traitement d'images par analyse asymptotique topologique

Ce chapitre résume les travaux contenus dans [9, 10, 12, 13, 16, 22, 26].

## 2.1 Introduction

Le gradient topologique consiste à étudier la variation d'un critère lorsqu'on perturbe le domaine d'étude. Nous considérons ici l'approche qui a été introduite pour étudier les problèmes d'optimisation topologique, dans lequel il faut généralement trouver une forme optimale et son complémentaire dans un domaine donné [133, 98, 104].

Cette méthode semble particulièrement adaptée au traitement d'images, puisque de nombreux problèmes d'imagerie (tels que la classification, la segmentation, le débruitage, l'inpainting, ...) reposent sur l'identification d'un sous-domaine particulier de l'image, celui des contours.

La difficulté généralement rencontrée dans un problème d'optimisation de formes est la non différentiabilité. En effet, rechercher un domaine optimal est équivalent à rechercher sa fonction caractéristique. Plusieurs méthodes classiques ont été développées pour rendre ce problème différentiable. Nous pouvons citer par exemple les méthodes de relaxation, permettant à la fonction caractéristique de prendre toutes les valeurs entre 0 et 1, et l'approche par courbes de niveaux (ou level set) qui consiste à remplacer la fonction caractéristique par une fonction régulière qui prend des valeurs positives dans le domaine recherché et négatives en dehors [133, 34, 33, 36, 51, 157].

L'optimisation topologique consiste à faire varier la fonction caractéristique de 0 à 1, mais uniquement dans des zones de taille infinitésimale. De cette façon, la variation de la fonction coût est faible quand une petite zone du domaine passe du sous-domaine recherché à son complémentaire (ou l'inverse). L'analyse asymptotique topologique permet justement d'étudier cette variation, et de dériver un gradient, dit topologique, de la fonction coût [133, 98, 158, 157].

Au cours de ce chapitre, nous rappelons dans un premier temps les outils de base liés à l'analyse asymptotique topologique, avant d'étudier plusieurs applications en traitement d'images : inpainting (reconstruction de l'image dans une zone où elle n'était pas connue), restauration et débruitage, classification, segmentation. Par la suite, nous présentons une méthode d'accélération des algorithmes que nous avons introduite, en s'appuyant sur la transformée de cosinus discrète et un préconditionnement approprié. Enfin, nous présentons un couplage entre le gradient topologique et la méthode des chemins minimaux pour améliorer la détection des contours et éviter que ceux-ci ne soient pas connexes.

## 2.2 Analyse asymptotique topologique

### 2.2.1 Présentation de la méthode

Soit $\Omega$ un ouvert borné régulier de $\mathbb{R}^2$ (ou $\mathbb{R}^3$). On considère une équation aux dérivées partielles définie sur $\Omega$, que l'on peut représenter sous forme variationnelle :

$$\text{trouver } u \in \mathcal{V} \text{ tel que } a(u, w) = l(w), \forall w \in \mathcal{V}, \tag{2.1}$$

où $\mathcal{V}$ représente un espace de Hilbert sur $\Omega$, généralement $H^1(\Omega)$, $a$ est une forme bilinéaire coercive continue sur $\mathcal{V}$, et $l$ est une forme linéaire continue sur $\mathcal{V}$. On considère ensuite une fonction coût $J$ mesurant un critère sur la solution $u$ de (2.1) : $J(\Omega, u)$.

Considérons maintenant une perturbation du domaine, par exemple représentée par l'insertion d'une fissure de petite taille $\sigma_\rho = x_0 + \rho\sigma(n)$, où $x_0 \in \Omega$ représente le point d'insertion de la fissure, $\sigma(n)$ est une fissure plane de normale unitaire $n$ contenant l'origine du domaine. Enfin $\rho > 0$ représente la taille de la perturbation, qui sera supposée petite. En notant $\Omega_\rho = \Omega \backslash \sigma_\rho$ le domaine ainsi perturbé, nous pouvons alors chercher la solution de l'EDP sur ce nouveau domaine :

$$\text{trouver } u_\rho \in \mathcal{V}_\rho \text{ tel que } a_\rho(u_\rho, w) = l_\rho(w), \forall w \in \mathcal{V}_\rho, \tag{2.2}$$

où $\mathcal{V}_\rho$, $a_\rho$ et $l_\rho$ représentent respectivement l'espace de Hilbert $\mathcal{V}$ restreint à $\Omega_\rho$, et les formes bilinéaire et linéaire perturbées.

La fonction coût peut alors être réécrite comme une fonction de $\rho$ uniquement, en considérant la carte suivante :

$$j : \rho \mapsto \Omega_\rho \mapsto u_\rho \text{ solution de (2.2) } \mapsto j(\rho) := J(\Omega_\rho, u_\rho). \tag{2.3}$$

La théorie de l'analyse asymptotique topologique donne alors un développement asymptotique de la fonctionnelle $j$ lorsque $\rho$ tend vers 0 :

$$j(\rho) - j(0) = f(\rho)G(x_0) + o(f(\rho)), \tag{2.4}$$

où $f(\rho)$ est une fonction positive tendant vers 0 lorsque $\rho$ tend vers 0, et $G(x_0)$ est appelé le gradient topologique au point $x_0$ d'insertion de la perturbation [133].

La minimisation du critère $j$ consiste alors à perturber le domaine aux endroits où le gradient topologique $G$ est le plus négatif, en s'appuyant sur le développement asymptotique (2.4).

### 2.2.2 Résultat principal

Dans la suite des travaux, nous nous appuierons sur le résultat suivant [37] :

**Théorème 2.1** *S'il existe une forme linéaire $L_\rho$ définie sur $\mathcal{V}_\rho$, une fonction $f$ : $\mathbb{R}^+ \to \mathbb{R}^+$, et 4 nombres réels $\delta J_1$, $\delta J_2$, $\delta a$ et $\delta l$ tels que*

- $\lim_{\rho \to 0} f(\rho) = 0,$
- $J(\Omega_\rho, u_\rho) - J(\Omega_\rho, u_0) = L_\rho(u_\rho - u_0) + f(\rho)\delta J_1 + o(f(\rho)),$
- $J(\Omega_\rho, u_0) - J(\Omega, u_0) = f(\rho)\delta J_2 + o(f(\rho)),$
- $(a_\rho - a_0)(u_0, p_\rho) = f(\rho)\delta a + o(f(\rho)),$
- $(l_\rho - l_0)(p_\rho) = f(\rho)\delta l + o(f(\rho)),$

*où $p_\rho$ est l'état adjoint, solution de*

$$a_\rho(w, p_\rho) = -L_\rho(w), \forall w \in \mathcal{V}_\rho, \tag{2.5}$$

*et $u_\rho$ est solution de (2.2), alors la fonction coût $j$ admet le développement asymptotique (2.4), où le gradient topologique est défini par*

$$G(x) = \delta J_1 + \delta J_2 + \delta a - \delta l. \tag{2.6}$$

## 2.3 Inpainting [9, 13]

Dans cette section, nous appliquons la méthode du gradient topologique au problème de l'"inpainting" d'une image. L'inpainting consiste à reconstituer les parties cachées d'une image, par exemple par un masque, ou à cause d'une détérioration, afin de récupérer l'image entière et originale. Les applications sont nombreuses, par exemple pour enlever les tâches sur une image ou sur un film mal conservé, pour supprimer un logo ou une image incrustée sur une autre image, ...

Ce problème est classiquement résolu par l'une des approches suivantes : méthodes d'apprentissage (e.g. réseaux de neurones) [177, 178], minimisation d'une fonctionnelle d'énergie reposant sur la variation totale [67, 68], analyse morphologique pour séparer la texture du décor [87], ...

Pour étudier ce problème, nous allons dans un premier temps essayer d'identifier et reconstituer les contours (ou arêtes) principaux de l'image dans la zone cachée (et donc inconnue). La technique de détection des fissures est basée sur la diffusion thermique classique [142, 66, 174, 175, 150], que nous avons améliorée en modélisant les contours de l'image comme des fissures. À l'aide de la connaissance à la fois de la condition de Dirichlet et de celle de Neumann au bord de la zone cachée, nous pouvons définir un critère, mesurant l'écart entre la solution d'un problème de Dirichlet et celle d'un problème de Neumann [118]. Ce problème est analogue au problème de conductivité inverse, aussi connu sous le nom de problème de Calderón [65]. Seulement deux mesures sont nécessaires pour retrouver des fissures simples [30, 31, 48]. Plusieurs approches numériques ont été proposées dans [40, 49, 50, 64, 96, 152, 151], mais la plus performante semble être l'approche par analyse asymptotique topologique. La minimisation du critère nous permettra alors d'identifier les principales fissures de l'image dans la partie inconnue. Enfin, l'image sera reconstituée entre les fissures grâce au Laplacien.

Cette section résume les travaux présentés dans [9, 13]. De nombreux tests numériques sont également présentés dans ces références.

### 2.3.1 Problème de localisation des fissures

Dans cette partie, nous supposons que le domaine $\Omega$ contient une fissure $\sigma^*$. On impose un flux $\phi \in H^{-1/2}(\Gamma)$ sur le bord $\Gamma$ du domaine $\Omega$, et on veut trouver $\sigma \subset \Omega$ telle que la solution $u \in H^1(\Omega \backslash \sigma)$ de

$$\left\{ \begin{array}{ll} \Delta u = 0 & dans\ \Omega \backslash \sigma, \\ \partial_n u = \phi & sur\ \Gamma, \\ \partial_n u = 0 & sur\ \sigma, \end{array} \right. \tag{2.7}$$

vérifie $u|_\Gamma = T$ où $T \in H^{1/2}(\Gamma)$ est donnée. Certaines conditions de compatibilité sur les données assurent que le problème est bien posé.

Une approche par gradient topologique a été introduite dans [37], et consiste à définir deux solutions différentes à partir des deux données de Dirichlet et Neumann :

$$u_D \in H^1(\Omega \backslash \sigma) \text{ telle que } \left\{ \begin{array}{ll} \Delta u_D = 0 & dans\ \Omega \backslash \sigma, \\ u_D = T & sur\ \Gamma, \\ \partial_n u_D = 0 & sur\ \sigma, \end{array} \right. \tag{2.8}$$

$$u_N \in H^1(\Omega \backslash \sigma) \text{ telle que } \left\{ \begin{array}{ll} \Delta u_N = 0 & dans\ \Omega \backslash \sigma, \\ \partial_n u_N = \phi & sur\ \Gamma, \\ \partial_n u_N = 0 & sur\ \sigma. \end{array} \right. \tag{2.9}$$

En remarquant que les deux solutions coïncident si $\sigma = \sigma^*$, l'idée consiste à minimiser la fonction coût

$$J(\sigma) = \frac{1}{2} \|u_D - u_N\|^2_{L^2(\Omega)}. \tag{2.10}$$

Le développement asymptotique topologique de cette fonctionnelle est détaillé dans [37].

### 2.3.2 Formulation des problèmes de Dirichlet et Neumann pour l'inpainting

Dans notre approche, $\Omega$ représente désormais l'image, on note $\Gamma$ sa frontière, et on dénote par $\omega \subset \Omega$ la partie cachée (inconnue) de l'image, et $\gamma$ sa frontière. Soit $v$ l'image dégradée et que l'on souhaite reconstituer. On suppose donc que $v$ est connue sur $\Omega \backslash \omega$, et inconnue sur $\omega$.

L'idée consiste à adapter la méthode de localisation des fissures à l'inpainting : nous allons chercher à identifier les fissures, ou contours, $\sigma$ dans la partie cachée $\omega$ de l'image, puis imposer que le Laplacien de l'image reconstruite soit nul dans $\omega\backslash\sigma$. Pour une fissure $\sigma$ donnée (i.e. pour un contour donné dans la partie cachée de l'image), et en utilisant la connaissance de $v$ (donnée de Dirichlet) et de son flux (donnée de Neumann) sur le bord $\gamma$ de $\omega$, on peut reconstruire deux solutions différentes à l'intérieur de $\omega$.

Pour une fissure $\sigma$ donnée dans $\omega$, la solution $u_D(\sigma) \in H^1(\Omega\backslash\sigma)$ du problème de Dirichlet vérifie

$$\begin{cases} \Delta u_D = 0 \quad dans\ \omega\backslash\sigma, \\ u_D = v \quad sur\ \gamma, \\ \partial_n u_D = 0 \quad sur\ \sigma, \\ u_D = v \quad dans\ \Omega\backslash\omega. \end{cases} \tag{2.11}$$

En dehors de $\omega$, la solution est égale à l'image originale, et à l'intérieur de $\omega$, nous reprenons l'équation (2.8). De même, en supposant $v$ régulière, on peut définir une solution $u_N \in H^1(\Omega\backslash\sigma)$ du problème de Neumann :

$$\begin{cases} \Delta u_N = 0 \quad dans\ \omega\backslash\sigma, \\ \partial_n u_N = \partial_n v \quad sur\ \gamma, \\ \partial_n u_N = 0 \quad sur\ \sigma, \\ u_N = v \quad dans\ \Omega\backslash\omega. \end{cases} \tag{2.12}$$

À noter que d'un point de vue numérique, il est beaucoup plus simple de résoudre un problème de Neumann approché :

$$\begin{cases} \Delta u_N = 0 \quad dans\ \omega\backslash\sigma, \\ \partial_n u_N = \partial_n v \quad sur\ \gamma, \\ \partial_n u_N = 0 \quad sur\ \sigma, \\ -\alpha\Delta u_N + u_N = v \quad dans\ \Omega\backslash\omega, \end{cases} \tag{2.13}$$

où $\alpha > 0$ est petit.

### 2.3.3 Développement asymptotique

La fonction coût reste inchangée et est définie par (2.10), puisque nous souhaitons trouver les fissures $\sigma \subset \omega$ qui minimisent l'écart entre ces deux solutions. En supposant que la fissure $\sigma$ est égale à $x + \rho\overline{\sigma}$, où $x$ est le point d'insertion de la fissure, $\rho$ est la taille de la fissure insérée (supposée petite) et $\overline{\sigma}$ est une fissure de référence, de normale $n$, alors on peut réécrire la fonction coût $J$ définie par (2.10) comme étant une fonction $j(\rho)$ de $\rho$ uniquement. Le développement asymptotique est alors donné par

$$j(\rho) - j(0) = f(\rho)g(x,n) + o(f(\rho)), \tag{2.14}$$

où le gradient topologique $g$ est donné par

$$g(x,n) = -\left[(\nabla u_D(x).n)(\nabla p_D(x).n) + (\nabla u_N(x).n)(\nabla p_N(x).n)\right], \tag{2.15}$$

où $u_D$ et $u_N$ sont solutions de (2.11) et (2.12) respectivement, mais sans fissure $\sigma$ insérée ($\sigma = \emptyset$). $p_D$ et $p_N$ sont les états adjoints associés, respectivement solutions dans $H^1(\Omega)$ de

$$\begin{cases} p_D = 0 \quad dans\ \Omega\backslash\omega, \\ p_D = 0 \quad sur\ \gamma, \\ -\Delta p_D = -(u_D - u_N) \quad dans\ \omega, \end{cases} \tag{2.16}$$

$$\begin{cases} p_N = 0 \quad dans\ \Omega\backslash\omega, \\ \partial_n p_N = 0 \quad sur\ \gamma, \\ -\Delta p_N = +(u_D - u_N) \quad dans\ \omega. \end{cases} \tag{2.17}$$

Le gradient topologique défini par (2.15) peut se réécrire sous la forme

$$g(x, n) = n^T M(x) n, \tag{2.18}$$

où $M(x)$ est une matrice symétrique $2 \times 2$ (ou $3 \times 3$ dans le cas d'images tridimensionnelles, ou films) définie par

$$M(x) = -sym(\nabla u_D(x) \otimes \nabla p_D(x) + \nabla u_N(x) \otimes \nabla p_N(x)). \tag{2.19}$$

On en déduit que la valeur minimale de $g(x, n)$ est atteinte lorsque $n$ est le vecteur propre associé à la valeur propre $\lambda_{min}(M(x))$ la plus négative de $M(x)$.

### 2.3.4 Algorithme

L'algorithme que nous avons défini est donc le suivant :

- Résolution des deux problèmes directs (2.11) et (2.12) non perturbés ($\sigma = \emptyset$).
- Résolution des deux problèmes adjoints (2.16) et (2.17) non perturbés.
- Assemblage en chaque point $x \in \omega$ de la matrice $M(x)$ définie par (2.19).
- Définition de l'ensemble des fissures identifiées :

$$\sigma = \{x \in \omega; \lambda_{min}(M(x)) < \delta < 0\}, \tag{2.20}$$

  où $\delta$ est un seuil négatif.
- Résolution du problème de Neumann direct (2.12) perturbé par $\sigma$.

L'image ainsi obtenue coïncide avec l'image originale en dehors de $\omega$, et a été reconstruite à l'intérieur de $\omega$.

### 2.3.5 Remarques

D'un point de vue numérique, les fissures sont généralement modélisées par une conductivité très faible plutôt que par un trou effectif dans le domaine de calcul. L'algorithme défini précédemment a une complexité en $\mathcal{O}(n.\log(n))$, où $n$ est le nombre de pixels de l'image, comme expliqué au paragraphe 2.7.

L'avantage de cette méthode est de permettre la reconstruction de l'image à l'aide d'uniquement 5 résolutions d'EDPs, les deux problèmes directs, les deux problèmes adjoints, puis un problème direct perturbé. Les résultats présentés dans [9] montrent la qualité des résultats obtenus en appliquant ainsi seulement une itération du gradient topologique.

Le paramètre de contrôle de la méthode se situe dans le seuillage du gradient : en dessous d'un certain seuil, nous estimons que les pixels considérés sont des contours de l'image, alors qu'au-dessus, nous estimons que ce n'est pas le cas. La résolution du dernier problème perturbé (2.12) pour trouver l'image reconstruite repose sur la résolution du problème de Poisson, et la netteté de l'image obtenue dépend de la connexité des contours identifiés. En effet, si un contour n'est pas fermé, le Laplacien crée une zone floue. De fait, le cœfficient de seuillage est généralement calé pour assurer la fermeture des principaux contours de l'image, quitte à prendre en compte des pixels qui ne sont en fait pas sur les contours de l'image. Nous avons constaté que numériquement, ce paramètre dépend très peu des images traitées et peut être fixé a priori.

Une solution à ce problème de seuillage est étudiée en section 2.8.

## 2.4 Restauration [16, 22]

Nous considérons dans cette section le problème de la restauration d'une image, essentiellement dans le but de la débruiter. L'idée principale de la méthode est de détecter les contours de l'image bruitée, en utilisant le gradient topologique, afin de les préserver dans le processus de restauration.

La méthode est également dérivée de la diffusion thermique. Les méthodes variationnelles généralement considérées reposent sur la diffusion isotrope ou anisotrope non linéaire, et sur la minimisation de la variation totale [142, 66, 124, 174, 175, 43]. D'autres approches non variationnelles ont également été développées, et notamment des méthodes statistiques [86].

Cette section résume les travaux présentés dans [16, 22]. Nous ne détaillons pas ici les résultats des tests numériques qui se trouvent dans ces références.

### 2.4.1 Formulation variationnelle

Soit $\Omega \subset \mathbb{R}^2$ un ouvert borné, et soit $v \in L^2(\Omega)$ l'image bruitée. Le débruitage d'une image passe généralement par la résolution du problème suivant :

$$\text{trouver } u \in H^1(\Omega) \text{ telle que } \begin{cases} -div(c\nabla u) + u = v & dans \quad \Omega, \\ \partial_n u = 0 & sur \quad \partial\Omega, \end{cases} \tag{2.21}$$

où $n$ représente la normale unitaire extérieure à $\partial\Omega$, et $c$ est la conductivité, que nous allons définir par la suite. Différents choix de conductivité sont possibles, essentiellement $c$ constant (méthode de diffusion linéaire, rapide mais qui rend l'image floue), et $c$ défini par une fonction non linéaire de $\nabla u$ (diffusion non linéaire, qui préserve les contours [175, 43]). Dans notre approche, $c$ ne prendra que 2 valeurs, soit une valeur de l'ordre de 1 en dehors des contours de l'image afin de lisser l'image, soit 0 sur les contours afin de les préserver.

Imposer $c = 0$ sur une partie de l'image revient à perturber le domaine en insérant des fissures. Pour un point $x_0 \in \Omega$ fixé, et pour un paramètre $\rho > 0$ supposé petit, on considère le domaine perturbé $\Omega_\rho = \Omega \backslash \sigma_\rho$ par l'insertion d'une fissure $\sigma_\rho = x_0 + \rho\sigma(n)$, où $\sigma(n)$ est une fissure de normale unitaire $n$ contenant l'origine du domaine. Le problème perturbé peut s'écrire sous la forme variationnelle suivante :

$$\text{trouver } u_\rho \in H^1(\Omega_\rho) \text{ telle que } a_\rho(u_\rho, w) = l_\rho(w),\ \forall w \in H^1(\Omega_\rho), \tag{2.22}$$

où $a_\rho$ (resp. $l_\rho$) est une forme bilinéaire (resp. linéaire) définie sur $H^1(\Omega_\rho)$ (resp. $L^2(\Omega_\rho)$) par

$$a_\rho(u, w) = \int_{\Omega_\rho} (c\nabla u \nabla w + uw)\, dx, \qquad l_\rho(w) = \int_{\Omega_\rho} vw\, dx. \tag{2.23}$$

La détection des contours de l'image revient à minimiser par rapport au domaine la fonctionnelle d'énergie suivante :

$$j(\rho) = J(\Omega_\rho, u_\rho) = \int_{\Omega_\rho} \|\nabla u_\rho\|^2. \tag{2.24}$$

### 2.4.2 Gradient topologique

En s'appuyant sur le théorème 2.1, un développement asymptotique peut être dérivé pour la fonction coût (2.24) et s'écrit sous la forme :

$$j(\rho) - j(0) = \rho^2 G(x_0, n) + o(\rho^2), \tag{2.25}$$

avec

$$G(x_0, n) = -\pi c(\nabla u_0(x_0).n)(\nabla p_0(x_0).n) - \pi|\nabla u_0(x_0).n|^2, \tag{2.26}$$

où $p_0$ est la solution du problème adjoint non perturbé :

$$\begin{cases} -div(c\nabla p_0) + p_0 = -\partial_u J(\Omega, u_0) & dans \quad \Omega, \\ \partial_n p_0 = 0 & sur \quad \partial\Omega. \end{cases} \tag{2.27}$$

Comme précédemment, le gradient topologique peut se réécrire sous la forme $G(x, n) = \langle M(x)n, n\rangle$, où $M(x)$ est la matrice symétrique $2 \times 2$ (en dimension 2) définie par

$$M(x) = -\pi c \frac{\nabla u_0(x)\nabla p_0(x)^T + \nabla p_0(x)\nabla u_0(x)^T}{2} - \pi \nabla u_0(x)\nabla u_0(x)^T. \tag{2.28}$$

### 2.4.3 Algorithme

L'algorithme consiste à insérer des petites inhomogénéités (ou fissures) dans les zones de l'image où le gradient topologique est inférieur à un certain seuil, ce qui signifiera qu'il s'agit vraisemblablement de contours de l'image. Notre algorithme est le suivant :

- Initialisation : $c = c_0$ (valeur constante, imposée partout).
- Résolution des problèmes direct (2.21) et adjoint (2.27) non perturbés.
- Assemblage de la matrice $2 \times 2$ $M(x)$ définie par (2.28) en chaque point de l'image, et calcul de sa valeur propre la plus négative $\lambda_{min}(M(x))$.
- On définit la nouvelle conductivité :

$$c_1 = \begin{cases} \varepsilon \text{ si } x \in \Omega \text{ est tel que } \lambda_{min}(M(x)) < \alpha < 0, \\ c_0 \text{ sinon,} \end{cases} \tag{2.29}$$

  où $\varepsilon > 0$ est supposé petit, et $\alpha$ est un cœfficient de seuillage négatif.
- Résolution du problème direct (2.21) en utilisant $c = c_1$.

L'image ainsi obtenue est l'image débruitée.

### 2.4.4 Remarques

D'un point de vue numérique, il est beaucoup plus facile d'imposer $c$ égal à une valeur très faible (ici $\varepsilon$) plutôt que de travailler sur un domaine perturbé $\Omega_\rho$. La résolution du problème (2.21) avec $c = c_1$ est une approximation de la résolution du problème perturbé (2.22), d'autant plus précise que $\varepsilon$ est petit.

Comme précédemment, cet algorithme est extrêmement efficace, et ne nécessite que 3 résolutions d'une EDP sur le domaine de l'image : les problèmes direct et adjoint non perturbés, puis le problème direct perturbé. La complexité de cet algorithme est une fois encore en $\mathcal{O}(n.\log(n))$ (voir le paragraphe 2.7).

Les résultats ainsi obtenus, présentés par exemple dans [16], sont de très bonne qualité, comme on peut le constater visuellement, ou en utilisant le ratio signal sur bruit (SNR). Il faut noter que là encore, l'algorithme nécessite le seuillage du gradient topologique, afin de décider si les points considérés font partie des contours de l'image ou non. Contrairement à l'inpainting, le fait de ne pas identifier des contours fermés n'est pas essentiel, au sens où cela influe assez peu sur le résultat. Toutefois, la méthode présentée au paragraphe 2.8 permet ici aussi d'obtenir des contours fermés, avec moins de points faussement identifiés comme appartenant aux contours. La qualité de l'image restaurée est ainsi meilleure.

### 2.4.5 Couplage des canaux dans le cas d'images couleur

Il existe plusieurs façons de considérer une image en couleur. En utilisant un espace adapté pour la représentation multi-dimensionnelle, par exemple l'espace RGB (rouge-vert-bleu), l'image en couleur peut être modélisée par une fonction de $\Omega$ dans $\mathbb{R}^3$ au lieu de $\mathbb{R}$. Une première approche consiste à découpler les canaux, et résoudre un problème direct et un problème adjoint pour chacun des canaux. Mais ces équations peuvent également être résolues directement de façon vectorielle pour identifier les images vectorielles $u$ et $p$. Le développement asymptotique topologique reste donné par les équations (2.25-2.26) et (2.28), où les fonctions impliquées sont vectorielles, i.e. le gradient topologique vectoriel est la somme des expressions correspondant à chacun des canaux [22].

Une autre approche a été étudiée dans [22], en utilisant une norme qui permet de coupler les différents canaux entre eux. Dans l'approche définie par Di Zenzo [83], l'idée consiste à considérer le tenseur de structure

$$T = \begin{pmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{pmatrix}, \qquad t_{ij} = \sum_{k=1}^{3} \frac{\partial u^k}{\partial x_i} \frac{\partial u^k}{\partial x_j}, \quad 1 \leq i, j \leq 2, \tag{2.30}$$

dans le cas d'images bi-dimensionnelles. Ce tenseur décrit la structure différentielle de l'image, et le gradient de Di Zenzo est défini par la plus grande valeur propre de ce tenseur :

$$\|\nabla u\|_{DZ} = \frac{1}{\sqrt{2}} \left[ t_{11} + t_{22} + \sqrt{(t_{11} - t_{22})^2 + 4t_{12}^2} \right]^{\frac{1}{2}}. \tag{2.31}$$

Grâce à une réécriture du gradient à l'aide de la fonction suivante :

$$H(\nabla u) = \frac{1 + \sqrt{1 - 4f(\nabla u)}}{2}, \tag{2.32}$$

avec

$$f(\nabla u) = \frac{det^2(\nabla u^1, \nabla u^2) + det^2(\nabla u^1, \nabla u^3) + det^2(\nabla u^2, \nabla u^3)}{(|\nabla u^1|^2 + |\nabla u^2|^2 + |\nabla u^3|^2)^2}, \tag{2.33}$$

$$det^2(\nabla u^s, \nabla u^t) = \left( \frac{\partial u^s}{\partial x_1} \frac{\partial u^t}{\partial x_2} - \frac{\partial u^t}{\partial x_1} \frac{\partial u^s}{\partial x_2} \right)^2, \tag{2.34}$$

il est possible de dériver le développement asymptotique de la fonction coût définie par l'équation (2.24) dans laquelle la norme utilisée est celle définie en (2.31) :

$$G(x_0, n) = \sum_{k=1}^{3} \left[ -\pi c(\nabla u_0^k(x_0).n)(\nabla v_0^k(x_0).n) - \pi H(\nabla u_0(x_0)) |\nabla u_0^k(x_0).n|^2 \right] \tag{2.35}$$

en utilisant les mêmes notations que précédemment.

Dans [22], nous montrons que cette approche a le même coût de calcul que l'algorithme où les différents canaux sont découplés, alors qu'il permet d'améliorer la détection des contours de l'image, et par conséquent d'obtenir une image restaurée plus précise aux abords des contours de l'image.

## 2.5 Classification [10, 16]

Dans cette section, nous nous intéressons au problème de la classification d'une image en différentes classes, de façon régularisée et non supervisée.

L'idée repose sur une idée introduite dans [150, 43], consistant à coupler la classification avec la restauration. Nous adaptons ici cette idée à notre approche par analyse asymptotique topologique.

Cette section résume les travaux présentés dans [16, 10] sans détailler les résultats numériques.

### 2.5.1 Introduction du problème

Soit $v$ l'image originale définie sur $\Omega$, et soient $n$ classes (i.e. niveaux de gris, ou couleurs) $C_i, 1 \leq i \leq n$, que l'on suppose prédéfinies pour le moment. Le but de la classification est de trouver une partition de $\Omega$ en sous-ensembles $\Omega_i$, recouvrant tout le domaine $\Omega$, et tels que $v$ soit proche de $C_i$ dans $\Omega_i$.

Une approche variationnelle peut être définie à l'aide de la fonction coût suivante :

$$J((\Omega_i)_{i=1\ldots n}) = \sum_{i=1}^{n} \int_{\Omega_i} (v(x) - C_i)^2 \, dx + \alpha \sum_{i \neq j} |\Gamma_{ij}|, \tag{2.36}$$

où $\Gamma_{ij}$ représente l'interface entre deux sous-domaines $\Omega_i \cap \Omega_j$.

La difficulté majeure de cette approche est que les inconnues sont des ensembles et non des variables, ce qui rend l'approche par analyse asymptotique topologique particulièrement appropriée pour traiter ce problème. Le développement du gradient topologique ainsi que les résultats numériques sont présentés dans [10].

### 2.5.2 Approche couplée restauration-classification

Une autre solution consiste à coupler la classification avec la restauration, en reprenant l'approche présentée dans la section précédente 2.4. L'idée est de lisser beaucoup plus fortement l'image, puis de la classifier sans aucune régularisation. En effet, si l'on enlève le terme de régularisation de l'équation (2.36), cela revient à définir $\Omega_i$ comme étant l'ensemble des pixels qui sont plus proches de $C_i$ que des autres valeurs prédéfinies. En d'autres termes, les pixels sont assignés aux sous-domaines représentant leur classe la plus proche.

Au lieu de choisir $c = \varepsilon$ ou $c = c_0$ pour le problème perturbé (équation (2.29)), nous choisissons de définir $c = c_1$ avec

$$c_1 = \begin{cases} \varepsilon \text{ sur les contours identifiés,} \\ \dfrac{c_0}{\varepsilon} \text{ en dehors.} \end{cases} \tag{2.37}$$

L'algorithme est alors le suivant :

- Application de l'algorithme de restauration défini en section 2.4, en remplaçant (2.29) par (2.37).
- Classification non régularisée de l'image $u_1$ ainsi obtenue, en assignant par exemple chaque pixel à sa classe la plus proche.

Comme précédemment, la complexité de l'algorithme est en $\mathcal{O}(n.\log(n))$, et les résultats numériques présentés dans [10] comparent l'efficacité relative des différentes approches proposées. De plus, comme cela est présenté, en jouant sur le paramètre $c_1$, il est possible de régulariser plus ou moins la solution obtenue, et cela permet également d'obtenir de très bons résultats sur des images bruitées.

### 2.5.3 Extension au cas non supervisé

En supposant que le nombre de classes $n$ soit donné, mais pas leurs valeurs $C_i$, il est possible de les déterminer de façon optimale en les intégrant dans la minimisation de la fonction coût :

$$\min_{(\Omega_i),(C_i)} j((\Omega_i),(C_i)) = \sum_{i=1}^{n} \int_{\Omega_i} |v(x) - C_i|^2 \, dx + \alpha \sum_{i \neq j} |\Gamma_{ij}|. \tag{2.38}$$

L'idée est de minimiser la fonctionnelle $j$ alternativement par rapport aux sous-domaines et par rapport aux classes. La minimisation par rapport aux sous-domaines revient à classifier l'image pour les valeurs $C_i$, et la minimisation par rapport aux classes revient à imposer

$$C_i = \frac{1}{|\Omega_i|} \int_{\Omega_i} v(x) \, dx. \tag{2.39}$$

L'algorithme de classification non supervisée est alors :

- Initialisation : choisir un jeu de classes $C_1, \ldots, C_n$, par exemple équi-réparties.
- Répéter jusqu'à convergence :
  - Trouver l'image classifiée pour les classes $C_1, \ldots, C_n$ en utilisant l'algorithme précédent.
  - Mettre à jour les classes avec (2.39).

De même, si le nombre $n$ de classes n'est pas donné, on peut ajouter un terme supplémentaire "$+\beta n$" dans la fonction coût (2.38), et minimiser également par rapport à $n$. Le choix du paramètre de régularisation $\beta$ influe directement sur le nombre de classes qui sera identifié.

## 2.6 Segmentation [12, 13]

La segmentation consiste à détecter automatiquement les différentes composantes d'une image. L'idée de base reste la même, à savoir qu'il faut d'abord commencer par chercher les contours de l'image, ce qui revient à séparer les différentes composantes de l'image.

Plusieurs approches ont été proposées dans la littérature. On peut citer les approches variationnelles, par exemple basées sur la minimisation de la fonctionnelle de Mumford-Shah [136], les contours actifs et méthodes de serpent [55, 156], les approches stochastiques [54, 61], les transformées d'ondelettes, ...[43, 45, 42, 140, 149, 150, 176].

Cette section présente les principaux résultats obtenus dans [12, 13]. Des tests numériques ont également été réalisés dans ces références afin de montrer l'efficacité de la méthode.

### 2.6.1 De la restauration à la segmentation

On considère à nouveau l'algorithme de restauration dans lequel on utilise la conductivité suivante pour le problème perturbé :

$$c(\varepsilon) = \begin{cases} \varepsilon \text{ sur } \omega, \\ \dfrac{1}{\varepsilon} \text{ en dehors de } \omega, \end{cases} \tag{2.40}$$

où $\omega \subset \Omega$ représentera par la suite l'ensemble des contours de l'image. Dans un premier temps, on suppose que $\omega$ n'est pas d'intérieur vide, i.e. qu'il est de co-dimension 0 dans $\Omega$. En utilisant (2.40), l'algorithme revient alors à considérer le problème suivant :

$$(\mathcal{P}_\varepsilon) \quad \begin{cases} -div(\varepsilon \nabla u_\varepsilon) + u_\varepsilon = v & dans \quad \omega, \\ -div\left(\dfrac{1}{\varepsilon}\nabla u_\varepsilon\right) + u_\varepsilon = v & dans \quad \Omega\backslash\omega, \\ \partial_n u_\varepsilon = 0 & sur \quad \partial\Omega, \end{cases} \tag{2.41}$$

où $u_\varepsilon \in H^1(\Omega)$, i.e. avec la condition implicite de recollement de $c(\varepsilon)\partial_n u_\varepsilon$ des deux côtés de $\partial\omega$.

Nous avons alors le résultat asymptotique suivant [12] :

**Théorème 2.2** *Si $u_\varepsilon$ est l'unique solution du problème $(\mathcal{P}_\varepsilon)$ dans $H^1(\Omega)$, alors*

$$\lim_{\varepsilon \to 0}(\|\nabla u_\varepsilon - \nabla u_0\|_{L^2(\Omega\backslash\omega)} + \|u_\varepsilon - u_0\|_{L^2(\omega)}) = 0, \tag{2.42}$$

*où $u_0 \in H^1(\Omega\backslash\omega) \cap L^2(\Omega)$ est la solution de*

$$(\mathcal{P}_0) \quad \begin{cases} u_0 = v & dans \quad \omega, \\ -div\,(\nabla u_0) = 0 & dans \quad \Omega\backslash\omega, \\ \partial_n u_0 = 0 & sur \quad \partial\omega, \\ \partial_n u_0 = 0 & sur \quad \partial\Omega. \end{cases} \tag{2.43}$$

Ce résultat nous indique qu'on peut approcher l'image segmentée $u_0$ à l'aide de $u_\varepsilon$. Désormais, on suppose que $\omega$ est de co-dimension 1 dans $\Omega$, ce qui permet de mieux gérer la situation réelle. En effet, du point de vue des applications, il est naturel de considérer que les contours forment un ensemble de dimension 1 lorsque l'image est de dimension 2 par exemple. Pour garder la cohérence avec les sections précédentes, nous noterons désormais $\sigma$ cet ensemble, qui désigne donc les contours de l'image. On suppose que cet ensemble est connu, grâce à la méthode de détection des contours que nous avons vue à plusieurs reprises précédemment.

Le problème $(\mathcal{P}_\varepsilon)$ devient désormais

$$(\tilde{\mathcal{P}}_\varepsilon) \quad \begin{cases} -div\left(\dfrac{1}{\varepsilon}\nabla u_\varepsilon\right) + u_\varepsilon = v & dans \quad \Omega\backslash\sigma, \\ \partial_n u_\varepsilon = 0 & sur \quad \sigma, \\ \partial_n u_\varepsilon = 0 & sur \quad \partial\Omega, \end{cases} \tag{2.44}$$

où $u_\varepsilon \in H^1(\Omega\backslash\sigma)$. L'existence et unicité de la solution est garantie si $v \in L^2(\Omega)$. Le résultat asymptotique devient alors [12] :

**Théorème 2.3** *Si $u_\varepsilon$ est l'unique solution du problème $(\tilde{\mathcal{P}}_\varepsilon)$ dans $H^1(\Omega\backslash\sigma)$, alors*

$$\|u_\varepsilon\|_{L^2(\Omega)} \leq \|v\|_{L^2(\Omega)}, \quad \|\nabla u_\varepsilon\|_{L^2(\Omega\backslash\sigma)} \leq \sqrt{\varepsilon}\|v\|_{L^2(\Omega)}, \tag{2.45}$$

*et*

$$\lim_{\varepsilon\to 0} \|\nabla u_\varepsilon - \nabla u_0\|_{L^2(\Omega\backslash\sigma)} = 0, \tag{2.46}$$

*où $u_0 \in H^1(\Omega\backslash\sigma)$ est l'unique solution de*

$$(\tilde{\mathcal{P}}_0) \quad \begin{cases} -div\,(\nabla u_0) = 0 & dans \quad \Omega\backslash\sigma, \\ \displaystyle\int_{\Omega_i} u_0 = \int_{\Omega_i} v & \forall\,\Omega_i \text{ composante connexe de } \Omega\backslash\sigma, \\ \partial_n u_0 = 0 & sur \quad \sigma, \\ \partial_n u_0 = 0 & sur \quad \partial\Omega. \end{cases} \tag{2.47}$$

La résolution directe du problème $(\tilde{\mathcal{P}}_0)$ peut s'avérer extrêmement coûteuse en pratique, à cause du très mauvais conditionnement du système. L'idée est alors de résoudre des approximations $(\tilde{\mathcal{P}}_\varepsilon)$, et d'approcher la solution $u_0$ à l'aide des solutions $u_\varepsilon$ ainsi construites.

### 2.6.2 Développement en série entière

Nous allons nous appuyer sur le développement en série entière de la solution $u_\varepsilon$ pour construire une suite d'approximations [12] :

**Théorème 2.4** *Il existe une constante $\varepsilon_R > 0$ et une famille de fonction $(u_n)_{n\in\mathbb{N}}$ de $H^1(\Omega\backslash\sigma)$ telle que pour tout $0 \leq \varepsilon \leq \varepsilon_R$,*

$$u_\varepsilon = \sum_{n=0}^{\infty} u_n \varepsilon^n. \tag{2.48}$$

*De plus, $u_0$ est l'unique solution du problème $(\tilde{\mathcal{P}}_0)$, et les autres fonctions sont les uniques solutions dans $H^1(\Omega\backslash\sigma)$ des problèmes respectifs suivants :*

$$(\tilde{\mathcal{P}}_1) \quad \begin{cases} -div\,(\nabla u_1) = -u_0 + v & dans \quad \Omega\backslash\sigma, \\ \displaystyle\int_{\Omega_i} u_1 = 0 & \forall\,\Omega_i \text{ composante connexe de } \Omega\backslash\sigma, \\ \partial_n u_1 = 0 & sur \quad \sigma, \\ \partial_n u_1 = 0 & sur \quad \partial\Omega, \end{cases} \tag{2.49}$$

$$n \geq 2, (\tilde{\mathcal{P}}_n) \quad \begin{cases} -div\,(\nabla u_n) = -u_{n-1} & dans \quad \Omega\backslash\sigma, \\ \displaystyle\int_{\Omega_i} u_n = 0 & \forall\,\Omega_i \text{ composante connexe de } \Omega\backslash\sigma, \\ \partial_n u_n = 0 & sur \quad \sigma, \\ \partial_n u_n = 0 & sur \quad \partial\Omega. \end{cases} \tag{2.50}$$

On peut alors définir la fonction suivante :

$$f(\varepsilon) := u_\varepsilon \in H^1(\Omega\backslash\sigma), \tag{2.51}$$

qui admet le développement en série entière (2.48) autour de l'origine. On considère alors une famille de points $(\varepsilon_i)$ choisis dans un intervalle $[\varepsilon_c, \varepsilon_R]$, où $\varepsilon_c$ est une valeur critique pour laquelle on estime qu'il n'est plus raisonnable de résoudre numériquement le problème $(\tilde{\mathcal{P}}_\varepsilon)$, et $\varepsilon_R$ est inférieur au rayon de convergence de la série. À l'aide de ces points, nous pouvons définir des polynômes d'interpolation de degré arbitraire :

$$g_N(\varepsilon) = \sum_{i=1}^{N} \left( \prod_{j=1, j\neq i}^{N} \frac{\varepsilon - \varepsilon_j}{\varepsilon_i - \varepsilon_j} \right) u_{\varepsilon_i}, \tag{2.52}$$

où $N$ est le nombre de points $\varepsilon_i$ choisis, et $g_N$ est alors de degré $N-1$.

L'analycité de la fonction $f$ permet d'estimer l'erreur d'approximation, et d'affirmer que

$$\|u_0 - g_N(0)\|_{H^1(\Omega\backslash\sigma)} = \mathcal{O}(\varepsilon_c^N). \tag{2.53}$$

### 2.6.3 Algorithme

On peut alors définir l'algorithme suivant, basé en partie sur l'algorithme de restauration défini en section 2.4 :

- Résoudre les problèmes direct (2.21) et adjoint (2.27) non perturbés (i.e. avec $c = c_0$ partout).
- Assembler la matrice $M(x)$ définie par (2.28) et calculer la valeur propre $\lambda_{min}(M(x))$ la plus négative en chaque point du domaine.
- Définir l'ensemble des contours : $\sigma = \{x \in \Omega;\ \lambda_{min} < \alpha < 0\}$ où $\alpha$ est un cœfficient de seuillage négatif.
- Définir la valeur critique $\varepsilon_c > 0$ en dessous de laquelle la résolution numérique du problème $(\tilde{\mathcal{P}}_\varepsilon)$ est difficile.
- Choisir le degré $N$ de l'approximation pour que la norme du résidu soit en $\mathcal{O}(\varepsilon_c^N)$, et $N$ valeurs différentes $(\varepsilon_i)$.
- Résoudre les $N$ problèmes $(\tilde{\mathcal{P}}_{\varepsilon_i})$.
- Calculer la valeur en 0 du polynôme d'interpolation $g_N$ défini par (2.52).

La complexité de cet algorithme est donc en $\mathcal{O}(N.n.\log(n))$, où $n$ est le nombre de pixels de l'image, et $N$ est le degré de l'approximation par interpolation. Typiquement, dans les tests numériques, $N$ est de l'ordre de 2 à 5.

Des tests numériques ont été réalisés pour montrer l'intérêt de cet algorithme, et sont présentés dans [12].

## 2.7 Complexité des algorithmes [13, 16]

Nous présentons ici les techniques que nous avons utilisées pour résoudre les différentes équations aux dérivées partielles, afin d'obtenir une complexité théorique en

$\mathcal{O}(n.\log(n))$ [13]. Des expériences numériques ont confirmé cette complexité théorique [16, 13].

### 2.7.1 Transformée de cosinus discrète

Tous les algorithmes que nous avons utilisés reposent sur des résolutions de l'équation

$$\begin{cases} -div(c\nabla u) + u = v & dans \quad \Omega, \\ \partial_n u = 0 & sur \quad \partial\Omega, \end{cases} \tag{2.54}$$

avec différentes valeurs de $c$. Les premières résolutions concernent les systèmes direct et adjoint non perturbés, i.e. avec une conductivité $c$ constante. En utilisant une transformée de cosinus discrète (DCT, équivalente à une transformée de Fourier discrète mais en ne gardant que les cosinus), le problème (2.54) est équivalent à résoudre

$$\sum_{m,n} \left(1 + c(m\pi)^2 + c(n\pi)^2\right) u_{m,n}\phi_{m,n} = \sum_{m,n} v_{m,n}\phi_{m,n}, \tag{2.55}$$

où les fonctions $\phi_{m,n} = \delta_{m,n}\cos(m\pi x)\cos(n\pi y)$ forment une base de cosinus dans $\mathbb{R}^2$, et où $(v_{m,n})$ représente les cœfficients de la DCT de l'image originale $v$. Par identification dans l'équation (2.55), les cœfficients $(u_{m,n})$ de la DCT de l'image $u$ que l'on cherche sont :

$$u_{m,n} = \frac{v_{m,n}}{1 + c(m\pi)^2 + c(n\pi)^2}. \tag{2.56}$$

La complexité d'une DCT est $\mathcal{O}(n.\log(n))$ où $n$ est le nombre de pixels de l'image. La résolution des problèmes non perturbés se fait de la façon suivante :

- Calcul des cœfficients $v_{m,n}$ de la DCT de l'image originale $v$.
- Calcul des cœfficients $u_{m,n}$ en utilisant (2.56).
- Assemblage de l'image $u$ à partir de ses cœfficients $u_{m,n}$ par une DCT inverse.

### 2.7.2 Gradient conjugué préconditionné

La solution des différents problèmes étudiés provient alors de la résolution d'un problème perturbé. Mais il faut noter que la perturbation est finalement assez faible au sens où la conductivité $c$ est constante, sauf dans une zone généralement négligeable de l'image.

Le problème perturbé qu'il faut résoudre peut s'écrire sous la forme

$$A(c)u = B, \tag{2.57}$$

où $u$ est l'inconnue. Nous avons vu que le problème (2.57) se résout très rapidement dans le cas $c$ constant. L'idée est de préconditionner le cas où $c$ n'est pas constant à l'aide du cas constant. Le problème (2.57) est équivalent au problème suivant :

$$\left[A(c_0)^{-1}A(c)\right] u = \left[A(c_0)^{-1}B\right]. \tag{2.58}$$

L'intérêt est que pour $c$ proche de $c_0$, ce qui est généralement le cas, la matrice du système à inverser est proche de l'identité, ce qui accélère la résolution. En utilisant un gradient conjugué préconditionné de cette façon, nous pouvons espérer rester en $\mathcal{O}(n.\log(n))$ opérations pour résoudre le problème perturbé. Tous les tests numériques que nous avons réalisés, en petite comme en grande dimension, confirment cette complexité théorique.

L'avantage indéniable est que cela permet de traiter des films en temps réel, à condition toutefois de découper le film en séquences courtes (de l'ordre d'une à deux secondes) pour pouvoir utiliser une approche tri-dimensionnelle sans retarder la diffusion de plus que quelques secondes. Notre approche permet également de traiter des images de grande taille en un temps quasiment négligeable (par exemple, une image $1600 \times 1200$ pixels en moins d'une seconde avec un code en C++).

## 2.8 Couplage du gradient topologique avec la méthode des chemins minimaux [26]

Comme nous l'avons vu, dans certaines applications comme la segmentation ou surtout l'inpainting, il est extrêmement important d'identifier des contours fermés, afin de pouvoir utiliser le Laplacien dans des zones indépendantes. Or le système que nous avons décrit pour identifier les contours repose sur un seuillage du gradient topologique : en dessous d'un certain seuil, les points sont considérés comme appartenant à des contours.

Nous avons constaté que les contours coïncident en fait avec les lignes de fond de vallée du gradient topologique. En adaptant le seuil, il est possible de les récupérer mais cela nous oblige à considérer beaucoup de points supplémentaires qui ne sont en fait pas des points appartenant aux contours. Pour identifier rapidement les lignes de fond de vallée du gradient topologique, nous proposons d'utiliser la méthode des chemins minimaux et du *fast marching* [71, 73, 82, 84, 179, 145, 164].

Dans la suite, nous considérons n'importe laquelle des applications en traitement d'image précédemment traitées. Nous supposons uniquement que le gradient topologique $g$ a été défini et calculé, et nous souhaitons identifier les lignes de fond de vallée, correspondant aux zones les plus négatives du gradient topologique.

Cette étude est tirée de [26], qui présente quelques résultats numériques dans le cadre de l'inpainting et de la segmentation.

### 2.8.1 Chemins minimaux

Soit $g$ le gradient topologique, l'idée des chemins minimaux est de définir une fonction potentiel, permettant de mesurer en chaque point du domaine $\Omega$ le coût pour un chemin de passer par ce point. Comme le potentiel doit être positif, nous proposons d'utiliser la fonction suivante :

$$P(x) = g(x) - \min_{y \in \Omega} \{g(y)\} . \tag{2.59}$$

L'avantage de ce choix est que plus le gradient topologique est négatif, plus la fonction potentielle est proche de 0, et moins c'est coûteux pour un chemin de passer à cet endroit. À l'inverse, il est extrêmement coûteux pour un chemin de passer par un point dont le gradient topologique est positif (ou faiblement négatif).

En notant $C(s)$ un chemin (ou une courbe) sur $\Omega$, où $s$ représente l'abscisse curviligne, on peut définir le coût total du chemin de la façon suivante :

$$J(C) = \int_C (P(C(s)) + \alpha)\, ds, \tag{2.60}$$

où $\alpha > 0$ est un cœfficient de régularisation, mesurant la longueur réelle du chemin.

Le but est de minimiser la fonction coût $J$, afin par exemple de trouver le chemin le plus court (en distance) et le moins onéreux (au sens de la fonction potentiel), reliant deux points. Pour ce faire, on définit la distance suivante sur l'image :

$$D(x; x_0) = \inf_{C \in \mathcal{A}(x,x_0)} J(C), \tag{2.61}$$

où $\mathcal{A}(x, x_0)$ est l'ensemble des chemins reliant $x$ à $x_0$.

### 2.8.2 Fast marching

Une façon extrêmement performante de construire la fonction distance définie par (2.61) est d'utiliser une équation de propagation d'un front :

$$\frac{\partial C(s,t)}{\partial t} = \frac{1}{P(C(s,t)) + \alpha}\, \mathbf{n}_C(s,t), \tag{2.62}$$

où $\mathbf{n}_C(s,t)$ est la normale unitaire extérieure au front $C$. L'initialisation est généralement faite avec $C(s, 0)$ égal à un cercle de taille infinitésimale autour du point $x_0$.

La méthode du fast marching est la suivante : d'après la théorie des équations Eikonales, la distance $D(x; x_0)$ est exactement l'instant $t$ pour lequel le front $C$ initialisé en $x_0$ atteint le point $x$ [179, 84].

En choisissant un point $x_0$ de référence, la résolution de l'équation (2.62) pour un temps suffisamment long (i.e. jusqu'à ce que tous les points aient été atteints, et donc jusqu'à ce que $C$ coïncide avec le bord du domaine $\partial\Omega$) donne directement la distance $D(x; x_0)$ pour tout point $x$ du domaine.

### 2.8.3 Algorithme couplé

En considérant que le minimum global du gradient topologique est certainement un point de contour, on l'utilise généralement comme point $x_0$ de référence. On peut alors construire la distance entre $x_0$ et n'importe quel point du domaine. À partir de là, un simple algorithme de descente sur la fonction distance en partant d'un point $x_1$ donnera le chemin minimal entre $x_0$ et $x_1$, c'est-à-dire le chemin le plus court et le moins coûteux entre ces deux points. Si $x_0$ et $x_1$ sont deux points sur le même contour, comme la fonction potentiel est choisie de telle sorte qu'il soit peu coûteux de rester sur le contour (grâce au gradient topologique), le chemin minimal sera une très bonne approximation du contour entre $x_0$ et $x_1$. L'avantage est que cette fois-ci, le contour identifié entre ces deux points sera fermé.

Une amélioration très peu coûteuse consiste à utiliser plusieurs points d'initialisation, et de construire la fonction distance à l'ensemble de ces points. Le diagramme de Voronoï correspondant peut être vu comme le graphe dual de celui de la fonction distance, et le point de distance minimale sur chaque arête du diagramme donne le point équidistant le plus proche des deux points-clés correspondant à l'arête.

L'algorithme est finalement le suivant :

- Construire le gradient topologique pour la détection de contours, comme vu dans les sections précédentes.
- Choisir $N$ points-clés. Le principal point-clé est le minimum global du gradient topologique, et les autres sont par exemple d'autres points ayant pour gradient topologique des valeurs proches du minimum global.
- Construire la fonction distance à cet ensemble de points-clés.
- En déduire le diagramme de Voronoï correspondant.
- Pour chaque arête, trouver le point de distance minimale.
- Pour chacun de ces points-selles, classés par distance croissante, vérifier qu'il ne servira pas à relier des points-clés déjà connectés à plusieurs autres points-clés.
- Si ce n'est pas le cas, utiliser ce point de distance minimale pour initialiser un algorithme de descente vers chacun des deux points clés voisins, afin de récupérer le chemin minimal correspondant entre les deux points-clés.

Cet algorithme converge et assure que chaque point-clé est connecté à au plus deux autres points-clés. Cela fournit donc un contour fermé, reliant les points-clés, et qui est une approximation d'un contour de l'image puisqu'il passe dans un fond de vallée du gradient topologique.

Comme nous l'avons constaté dans [26], cela permet d'améliorer sensiblement notre algorithme d'inpainting. Cela permet également de segmenter l'image plus efficacement. Pour les autres applications, nous n'avons pas constaté de gain réellement significatif en utilisant cette méthode.

## 2.9 Conclusions et perspectives

Nous avons utilisé la technique de détection de fissures, qui s'appuie sur le gradient topologique, dans le but d'étudier plusieurs aspects du traitement d'images : inpainting, restauration, segmentation, classification. Dans toutes ces applications, nous avons obtenu de très bons résultats, et surtout dans un temps de calcul très court.

Les méthodes présentées ici peuvent s'appliquer indifféremment aux images en noir et blanc, en couleur, bi- ou tri-dimensionnelles, ainsi qu'aux films. En effet, la complexité à la fois théorique et numérique que nous obtenons pour tous ces algorithmes nous a permis de traiter des films quasiment en temps réel (sur une machine bi-processeur, avec un programme écrit en $\text{c}_{++}$).

Un autre avantage réside dans le fait que tous les algorithmes que nous avons présentés ici s'appuient sur le même noyau, puisque les équations aux dérivées partielles à résoudre sont toutes du même type. Cela simplifie particulièrement la mise en œuvre pratique de ces algorithmes grâce à cette base commune.

Plusieurs pistes restent à l'étude, comme par exemple la possibilité d'utiliser des opérateurs différentiels d'ordre plus élevé que le Laplacien, dans le but de reconstituer une information sur le gradient de l'image. Par exemple, dans le cas de l'inpainting, l'image reconstruite est affine par morceaux dans chaque zone délimitée par les contours identifiés. En utilisant le même type de technique, la reconstruction du gradient de l'image devrait permettre de reconstruire plus précisément l'image.

# Chapitre 3

# Assimilation de données : le nudging direct et rétrograde

Ce chapitre résume les travaux contenus dans [8, 11, 14, 18, 21, 24, 25].

## 3.1 Introduction

L'assimilation de données, en météorologie comme en océanographie, consiste souvent à identifier l'état initial d'un système dynamique à partir d'observations. Le but principal est d'améliorer la connaissance de l'état actuel du système, pour en déduire des prévisions fiables de son évolution future [116, 53, 163].

Le nudging est une méthode d'assimilation de données basée sur la relaxation dynamique, dans le but d'ajuster le modèle et de le contraindre vers les observations. L'algorithme standard du nudging consiste à ajouter aux équations d'état du système un terme de rappel, proportionnel à la différence entre les observations et la quantité correspondante calculée par la résolution des équations d'état. Le modèle apparaît alors comme une contrainte faible et le terme de rappel force les variables du modèle à coller aux observations. Ce terme de rappel peut être ajusté grâce à un cœfficient (ou une matrice, suivant les cas). Il est généralement choisi de la façon suivante dans les expériences numériques : suffisamment petit pour que le terme de nudging reste faible en comparaison avec les autres termes des équations du modèle, et également assez grand pour contraindre suffisamment le modèle vers les observations. Le terme de nudging peut être vu comme un terme de pénalisation du modèle lorsqu'il s'éloigne trop des données. À noter que la méthode du nudging direct est également appelée observateur de Luenberger dans le domaine de l'automatique et du contrôle [129].

Le nudging est une méthode d'assimilation de données très simple, et nettement plus économique (d'un point de vue de la mise en œuvre pratique) que la plupart des autres méthodes, et en particulier des algorithmes variationnels comme le 4D-VAR [123]. Initialement introduit en météorologie [106], le nudging (encore appelé relaxation newtonienne) a ensuite été utilisé avec succès en océanographie sur un modèle quasi-géostrophique [168, 170, 57], puis appliqué à un système océanographique mésoéchelle opérationnel [160]. Les cœfficients du nudging peuvent être choisis de façon optimale en utilisant une méthode d'estimation de paramètres par approche variationnelle [159, 180]. Les cœfficients ainsi obtenus sont optimaux au sens où ils permettent d'obtenir la plus petite différence entre la trajectoire du modèle et les observations. Une comparaison entre le nudging optimal et les méthodes basées sur le filtre de Kalman a été réalisée dans [172]. Le principal inconvénient de ces méthodes de nudging optimal est qu'elles nécessitent la mise en œuvre et la résolution des équations du modèle adjoint, ce qui n'est pas utile pour le nudging standard.

Le nudging rétrograde consiste à résoudre les équations d'état du modèle de façon rétrograde en temps, en partant d'une observation de l'état du système à l'instant final. Un terme de rappel, avec un signe opposé à celui introduit dans le nudging direct, est ajouté aux équations du système, et l'état obtenu à l'instant final est en fait un état initial du système [1].

L'algorithme du nudging direct et rétrograde (Back and Forth Nudging, BFN) introduit dans [18] consiste à résoudre d'abord les équations directes du modèle avec le terme de nudging, puis en repartant de l'état final ainsi obtenu, à résoudre les mêmes équations de façon rétrograde avec un terme de rappel opposé à celui du nudging direct. On obtient ainsi à la fin de la résolution rétrograde une première estimation de l'état initial. Ce procédé est alors répété de façon itérative jusqu'à obtenir la convergence de l'état initial.

Cet algorithme peut être comparé au 4D-VAR (algorithme variationnel d'assimilation de données en dimension 3 d'espace et 1 de temps [123]), qui consiste aussi en une succession de résolutions de systèmes directs et rétrogrades. Mais dans l'algorithme BFN, il est inutile, même pour des problèmes non linéaires, de linéariser le modèle comme dans le 4D-VAR, et le système rétrograde n'est pas l'équation adjointe mais le système des équations du modèle avec un terme de rappel qui en fait un problème bien posé.

Nous pouvons citer ici quelques autres algorithmes reposant également sur des résolution directes et rétrogrades. Une approche similaire est étudiée dans [162, 161], à la différence principale qu'à chaque instant d'observation, la trajectoire du modèle est remplacée par les observations. Cela correspond à notre algorithme dans le cas particulier de cœfficients de nudging infinis et d'un système physique réversible. L'algorithme du quasi-inverse est une autre méthode directe-rétrograde, mais sans rapport avec le nudging [116]. En effet, dans cet algorithme, le signe des termes dissipatifs est changé dans les équations rétrogrades pour des raisons de stabilité. Notre algorithme utilise un terme de relaxation pour stabiliser les équations rétrogrades, et ainsi conserver le bon signe pour les termes dissipatifs.

Au cours de ce chapitre, nous définissons tout d'abord l'algorithme du nudging direct et rétrograde, ou BFN, dans un cadre général. Puis nous présentons des résultats théoriques de convergence dans des cas simplifiés (observations complètes et non bruitées), sur différents types de modèles : modèles linéaires, équations de transport (linéaires ou non, diffusives ou non). Puis nous listons les différentes expériences numériques qui ont été réalisées, et donnons les principales conclusions qui peuvent en être tirées. Enfin, en s'appuyant sur des méthodes développées dans le monde de l'automatique (où le nudging peut être relié à certains observateurs), nous montrons qu'il est notamment possible de corriger les variables du modèle non observées à partir de celles qui le sont, et grâce à celà améliorer les résultats obtenus. Enfin, quelques conclusions et perspectives sont données à la fin de ce chapitre.

## 3.2 Algorithme du "nudging direct et rétrograde", ou "Back and Forth Nudging" (BFN) [8, 11, 18]

### 3.2.1 Nudging direct

Pour simplifier les notations, nous supposons que les équations du modèle ont été discrétisées spatialement, à l'aide d'une méthode aux différences finies, éléments finis ou une méthode spectrale. Nous considérons ainsi un modèle continu en temps et régi par des équations dynamiques du type

$$\frac{dX}{dt} = F(X), \quad 0 < t < T, \tag{3.1}$$

avec une condition initiale $X(0) = x_0$. Dans cette équation, $F$ représente l'ensemble des opérateurs différentiels spatiaux et autres termes (linéaires ou non linéaires) du modèle.

Soit $C$ l'opérateur d'observation, permettant de relier les observations du système $X_{obs}(t)$ aux quantités correspondantes $C(X(t))$ calculées à partir des trajectoires $X(t)$ du modèle. Dans la pratique, l'opérateur d'observation $C$ modélise essentiellement deux phénomènes. D'une part, les observations ne sont pas faites exactement sur les points de grille du modèle numérique, et il faut donc interpoler entre les points du maillage. D'autre part, les observations ne correspondent pas forcément aux variables du modèle, mais à une autre quantité physique pouvant s'y relier. Par exemple, les satellites mesurent des radiances, qui peuvent être reliées aux variables du modèle comme la température ou la hauteur d'eau. Sauf mention contraire, nous ne supposons pas dans cette étude que $C$ est un opérateur linéaire.

Le nudging standard appliqué à l'équation (3.1) donne le modèle suivant :

$$\frac{dX}{dt} = F(X) + K(X_{obs} - C(X)), \quad 0 < t < T, \tag{3.2}$$

en conservant la même condition initiale, et où $K$ représente la matrice (ou éventuellement un cœfficient) de nudging, ou gain. Le modèle devient ainsi une contrainte faible, puisqu'il n'est plus satisfait exactement. Le terme de nudging a pour but de forcer le modèle vers les observations. Dans un cadre linéaire, c'est exactement l'observateur de Luenberger, ou observateur asymptotique, dans lequel la matrice $K$ de nudging peut être choisie de sorte que l'erreur tende vers 0 en temps infini [129]. Mais dans les applications géophysiques, nous ne pouvons pas toujours considérer un temps suffisamment long pour que le nudging standard donne de bons résultats.

### 3.2.2 Nudging rétrograde

L'idée du nudging rétrograde consiste à repartir de la condition finale obtenue à la fin de la résolution de (3.2) et à revenir à l'instant initial grâce aux équations du modèle. Si l'on considère le modèle rétrograde correspondant à (3.1), on obtient

$$\frac{d\tilde{X}}{dt} = F(\tilde{X}), \quad T > t > 0, \tag{3.3}$$

avec une condition finale $\tilde{X}(T) = \tilde{x}_T$. Lorsque le modèle direct est irréversible (ou dissipatif), ce problème est généralement mal posé. Pour le stabiliser, nous ajoutons un terme de rappel aux observations avec un signe opposé :

$$\frac{d\tilde{X}}{dt} = F(\tilde{X}) - K'(X_{obs} - C(\tilde{X})), \quad T > t > 0, \tag{3.4}$$

où $K'$ est la matrice de nudging rétrograde.

La résolution de ce modèle rétrograde permet d'obtenir une solution à l'instant $t = 0$, qui peut être vue comme une nouvelle estimation de l'état initial du système.

### 3.2.3 Algorithme BFN

L'algorithme du nudging direct et rétrograde (Back and Forth Nudging) a été introduit dans [18]. Il consiste à résoudre alternativement et itérativement l'équation directe avec nudging (3.2) et l'équation rétrograde avec nudging (3.4), en repartant à chaque fois de la dernière solution calculée. L'algorithme peut s'écrire sous la forme suivante :

$$\begin{aligned} k \geq 1 \quad & \begin{cases} \dfrac{dX_k}{dt} = F(X_k) + K(X_{obs} - C(X_k)), \\ X_k(0) = \tilde{X}_{k-1}(0), \end{cases} \\ k \geq 1 \quad & \begin{cases} \dfrac{d\tilde{X}_k}{dt} = F(\tilde{X}_k) - K'(X_{obs} - C(\tilde{X}_k)), \\ \tilde{X}_k(T) = X_k(T), \end{cases} \end{aligned} \tag{3.5}$$

avec la notation $\tilde{X}_0(0) = x_0$.

On peut remarquer que si les trajectoires directe et rétrograde convergent vers la même limite, alors en faisant la somme et la différence des deux équations dans (3.5), la trajectoire limite est solution du modèle direct (3.1) et elle coïncide avec les observations à travers l'opérateur d'observation $C$ et la matrice de gain $K$.

Dans la pratique, les observations sont discrètes en temps, ce qui revient à dire que le vecteur $X_{obs}$ n'est disponible et utilisable qu'à certains instants $(t_i)_{i=1\ldots N}$. Le terme de nudging est alors ajouté uniquement à ces instants-là :

$$\frac{dX}{dt} = F(X) + \sum_{i=1}^{N} K(X_{obs} - C(X))\delta(t - t_i), \quad 0 < t < T. \tag{3.6}$$

### 3.2.4 Choix des matrices de nudging et interprétations

Nous allons maintenant expliquer et justifier le choix des matrices de nudging, à travers différentes interprétations du nudging.

#### Interprétation variationnelle du nudging

Concernant la matrice de nudging direct, la méthode ayant été beaucoup étudiée et utilisée dans les 30 dernières années [106, 168, 160, 159], il existe plusieurs possibilités pour fixer la matrice $K$ de nudging direct. La plus connue est la méthode du nudging optimal, permettant de déterminer les cœfficients optimaux grâce à une méthode variationnelle d'estimation de paramètres [180, 172].

Nous pouvons déjà noter qu'imposer $K = 0$ dans (3.2) nous ramène aux équations du modèle (3.1). D'un autre côté, imposer $K = +\infty$ revient à substituer les observations à la trajectoire du modèle à chaque instant d'observation. L'un comme l'autre ont l'inconvénient de privilégier l'une des deux sources d'information (modèle et données) aux dépens de l'autre.

Soit $R$ la matrice de covariance des erreurs d'observation. Cette matrice intervient dans les méthodes classiques d'assimilation de données, aussi bien variationnelles (3D-VAR, 4D-VAR, 4D-PSAS, ...) que séquentielles (filtres dérivés du filtre de Kalman) [78, 90, 144, 99, 100]. En pratique, on ne connaît pas exactement les statistiques d'erreur, et on considère donc une approximation de cette matrice, qu'on suppose symétrique définie positive.

On suppose ici que le modèle est linéaire, i.e. que $F$ est un opérateur linéaire. On suppose également que $F$ représente un opérateur symétrique. Une discrétisation temporelle classique (méthode d'Euler implicite) de l'équation (3.2) est la suivante :

$$\frac{X^{n+1} - X^n}{\Delta t} = FX^{n+1} + K(X_{obs} - CX^{n+1}), \tag{3.7}$$

où $\Delta t$ représente le pas de discrétisation temporelle, et $X^n$ représente la solution du problème discret à l'insant $t_n$.

En choisissant

$$K = C^T R^{-1}, \tag{3.8}$$

alors le problème (3.7) est équivalent au problème d'optimisation suivant :

$$\begin{aligned} X^{n+1} = \arg\min_X \ \Bigg[ & \frac{1}{2}\langle X - X^n, X - X^n\rangle - \frac{\Delta t}{2}\langle FX, X\rangle \\ & + \frac{\Delta t}{2}\langle R^{-1}(X_{obs} - CX), X_{obs} - CX\rangle \Bigg]. \end{aligned} \tag{3.9}$$

Les deux premiers termes correspondent exactement à l'énergie du modèle direct discrétisé, et le dernier terme est la partie relative aux observations de la fonctionnelle coût de l'algorithme 4D-VAR [123]. Ce principe variationnel montre qu'à chaque pas de temps, le nudging direct revient à faire un compromis entre minimiser l'énergie du système et minimiser l'écart aux données.

Comme conséquence directe, nous constatons qu'il n'y a aucune nécessité d'introduire un terme supplémentaire de régularisation, comme c'est le cas dans le 4D-VAR avec un terme de rappel à une ébauche de la condition initiale. Dans l'algorithme BFN, il suffit d'initialiser avec l'ébauche, sans besoin d'avoir une information sur ses statistiques d'erreur, et sans la prendre en considération dans le reste de l'algorithme.

Le BFN fournit automatiquement une correction du modèle grâce aux observations, et le modèle est alors une contrainte faible. L'énorme avantage est que cette méthode est moins sensible au fait que le modèle théorique ne modélise pas forcément bien la réalité.

À noter que dans les cas non linéaires, le terme $\langle FX, X\rangle$ de l'équation (3.9) peut être remplacé par un terme $-G(X)$, où $G$ est l'énergie du système à l'équilibre.

#### Interprétation séquentielle

Une interprétation stochastique (ou séquentielle) du nudging est également possible, en le rapprochant du filtre de Kalman. En effet, dans les périodes de temps où les observations ne sont pas disponibles, le nudging se résume, comme le filtre de Kalman, à intégrer les équations du modèle pour propager la solution dans le temps [8].

Lorsque des observations sont disponibles, la solution du modèle est corrigée en utilisant l'écart entre les observations et la trajectoire modèle, comme dans le filtre de Kalman. Toutefois, nous ne considérons pas dans le BFN de matrices de gain optimales dans le but d'alléger la mise en œuvre pratique de l'algorithme, ainsi que son temps de calcul. Le nudging peut ainsi être vu comme un filtre de Kalman sous-optimal. Toutefois, le fait de résoudre alternativement des équations directes et rétrogrades en temps permet d'améliorer très sensiblement les résultats par rapport à la méthode du nudging standard.

#### Méthode de placement de pôles et matrice de nudging rétrograde

Le terme de nudging dans l'équation rétrograde a un double rôle : contraindre le modèle à rester proche des observations, et stabiliser la résolution rétrograde du modèle. En effet, l'irréversibilité des phénomènes physiques considérés fait qu'en général le système rétrograde est instable.

En faisant un changement de variable dans l'équation (3.4) pour se ramener à un temps croissant, on obtient (en supposant une fois encore les opérateurs linéaires) :

$$-\frac{d\tilde{X}}{dt'} = F\tilde{X} - K'(X_{obs} - C\tilde{X}), \tag{3.10}$$

avec $t' = T - t$. La matrice de stabilité du système est donc $-F - K'C$, et la stabilité numérique est garantie si toutes les valeurs propres de cette matrice sont à partie réelle négative.

Le théorème suivant [80, 41, 60, 167], connu sous le nom de méthode de placement de pôles, permet d'obtenir l'existence d'une matrice de nudging rétrograde $K'$ qui stabilise le système :

**Théorème 3.1** *Si le système $(F, C)$ est observable, alors il existe une matrice $K'$ telle que toutes les valeurs propres de $-F - K'C$ soient à partie réelle négative.*

L'observabilité du système est équivalente au fait que le rang de la matrice $[C, CF, \ldots, CF^{n-1}]$ est égal à $n$, où $n$ est la dimension du problème physique discrétisé.

Une telle matrice $K'$ peut parfois être exhibée, à condition de résoudre au préalable des équations du type Riccati, ce qui s'avère relativement coûteux en pratique.

## 3.3 Expériences numériques [11, 14, 21]

### 3.3.1 Valeurs numériques des matrices de nudging

D'un point de vue pratique, les expériences numériques ont pour la plupart été réalisées avec une matrice de nudging facile et rapide à implémenter :

$$K = C^T(kI) = kC^T, \tag{3.11}$$

où $k$ est un cœfficient scalaire de gain, et $I$ est la matrice identité de l'espace des observations. Ce choix est motivé par plusieurs remarques. Premièrement, les matrices de covariance telles que $R$ sont généralement mal déterminées. Par ailleurs, ce choix est extrêmement simple et ne nécessite pas d'appliquer une méthode d'estimation de paramètres. Enfin, le choix naturel de la matrice de nudging est $K = C^T L$ où $L$ est un opérateur linéaire sur l'espace des observations. En effet, si les observations ne sont pas localisées aux points du maillage, la matrice $K$ aura le rôle de ramener la correction $X_{obs} - C(X)$ des points d'observation vers les points du maillage.

De même, pour la matrice rétrograde, nous choisissons généralement

$$K' = C^T(k'I) = k'C^T, \tag{3.12}$$

comme dans le cas direct. Ce choix est également motivé par la simplicité et la rapidité de notre méthode dans ce cas là.

Les deux uniques paramètres de cette méthode deviennent alors les scalaires $k$ et $k'$. Le scalaire $k > 0$ est généralement fixé de sorte que le terme de nudging soit petit par rapport aux termes du modèle afin de respecter le compromis entre le modèle et les observations. Le scalaire $k' > 0$ est quant à lui choisi comme étant le plus petit cœfficient qui stabilise la résolution numérique de l'équation rétrograde.

### 3.3.2 Méthodologie expérimentale

Nous avons ensuite testé notre algorithme sur de nombreux modèles linéaires et non linéaires, chaotiques ou géophysiques simplifiés. L'approche est la même dans tous les cas. Elle consiste à réaliser des expériences jumelles avec des données simulées. Une première résolution permet d'extraire des données discrètes en temps et en espace. Ces données simulées sont ensuite bruitées, généralement avec un bruit blanc gaussien.

La condition initiale utilisée pour initialiser l'algorithme correspond suivant les cas à un champ constant, ou à la solution du modèle de référence à un instant différent et avec du bruit. Cette technique permet de simuler des données avec une répartition spatio-temporelle et un bruit de mesure à peu près réalistes, avec l'avantage de pouvoir comparer les résultats à la solution exacte, i.e. celle utilisée pour générer les données.

### 3.3.3 Modèles étudiés

Nous présentons ici les différents modèles sur lesquels nous avons testé et comparé notre algorithme. Nous renvoyons dans chaque cas à des références pour le détail des expériences et les résultats correspondants.

#### Système de Lorenz

Nous avons considéré le système de Lorenz dans une configuration chaotique [127] :

$$\begin{cases} \dfrac{dx}{dt} = 10(y-x), \\ \dfrac{dy}{dt} = 28x - y - xz, \\ \dfrac{dz}{dt} = -\dfrac{8}{3}z + xy. \end{cases} \tag{3.13}$$

Ce système d'équations différentielles ordinaires en dimension 3, dont les trajectoires prennent la forme caractéristique des ailes d'un papillon, possède deux attracteurs, autour desquels la solution oscille, passant de l'un à l'autre de façon chaotique. Les tests numériques, résultats de convergence, et comparaison avec la méthode variationnelle, sont détaillés dans [11].

#### Équation de Burgers visqueux

Nous avons ensuite considéré un modèle non linéaire géophysique extrêmement simple, correspondant à l'équation de Burgers sur un domaine périodique en dimension 1 :

$$\frac{\partial X}{\partial t} + \frac{1}{2}\frac{\partial (X^2)}{\partial s} - \nu \frac{\partial^2 X}{\partial s^2} = 0, \tag{3.14}$$

où $s$ représente l'abscisse curviligne sur le $45^o$ parallèle nord, et $t$ est le temps. Ce modèle peut être vu comme la projection 1D d'un modèle bi-dimensionnel de circulation météorologique à une pression (ou altitude) fixée. Nous avons choisi la période du domaine, les intervalles de temps considérés, ainsi que la répartition spatio-temporelle des observations afin d'obtenir une situation comparable à la réalité.

Il faut noter que ce système est non linéaire, et surtout irréversible (car dissipatif) à cause du terme de diffusion. Néanmoins, nous avons constaté qu'il était possible de résoudre le problème rétrograde grâce aux observations. Les résultats numériques montrant la convergence et la comparaison de cet algorithme avec la méthode variationnelle sont détaillés dans [11]. D'autres expériences numériques dans une situation physique différente ont été réalisées dans [21].

### Modèle “shallow water” (ou équations de Saint-Venant)

Le modèle shallow water (ou équations de Saint-Venant) est généralement utilisé en océanographie, en hydrologie et en mécanique des fluides. Il décrit l'évolution d'un fluide bi-dimensionnel à l'aide de 3 équations. Ces équations proviennent d'une intégration verticale des champs tri-dimensionnels, dans le cadre de l'approximation hydrostatique (i.e. en négligeant l'accélération verticale). Plusieurs variantes de ce modèle existent, et nous nous limitons à la formulation suivante qui fait intervenir la hauteur du fluide :

$$\begin{cases} \partial_t u - (f+\zeta)v + \partial_x B = \dfrac{\tau}{\rho_0 h} - ru + \nu\Delta u, \\ \partial_t v + (f+\zeta)u + \partial_y B = \dfrac{\tau}{\rho_0 h} - rv + \nu\Delta v, \\ \partial_t h + \partial_x(hu) + \partial_y(hv) = 0, \end{cases} \tag{3.15}$$

où $\zeta = \partial_x v - \partial_y u$ est la vorticité relative, $B = g^* h + \dfrac{1}{2}(u^2+v^2)$ est le potentiel de Bernoulli, $g^*$ est la gravité réduite, $f$ la force de Coriolis, $\rho_0$ la densité du fluide, $r$ le cœfficient de friction latérale, et $\nu$ la viscosité. Les trois variables sont $(u, v, h)$, respectivement les deux composantes longitudinale et transversale de la vitesse horizontale, et la hauteur d'eau. Enfin, $\tau$ représente le terme de couplage du modèle, représentant un forçage par le vent.

De plus amples détails sur ce modèle sont disponibles dans [56]. Les résultats de nombreuses expériences numériques sur ce modèle sont regroupés dans [14]. Outre des études de convergence et de sensibilité, une hybridation avec la méthode variationnelle a également été étudiée.

### Modèle quasi-géostrophique multi-couches

Nous avons enfin considéré un modèle quasi-géotrophique barocline à plusieurs couches [107, 169, 57]. Ce modèle dérive des équations primitives (lois de conservation de la masse, du moment, de la température et de la salinité), en effectuant plusieurs simplifications. Premièrement, ce modèle suppose que les effets inertiels sont négligeables devant la rotation terrestre (force de Coriolis). Le nombre de Rossby, rapport entre les échelles temporelles caractéristiques de ces deux phénomènes, est donc négligeable devant 1. La quasi-géostrophie suppose également que l'océan est de petite taille par rapport à la Terre, avec un rapport de l'ordre du nombre de Rossby. Ce modèle suppose aussi que l'épaisseur (ou profondeur) de l'océan est faible devant sa largeur. Toutes ces hypothèses ne sont bien évidemment pas valides dans le cas de l'océan Atlantique nord, par exemple, mais il a été prouvé que ces équations permettaient néanmoins de très bien modéliser la plupart des phénomènes physiques qui apparaissent aux latitudes moyennes (par exemple le Gulf stream).

Ce modèle suppose que l'océan est découpé en plusieurs niveaux verticaux, où la densité du fluide est constante. Les équations proviennent de la conservation de la vorticité, et s'écrivent :

$$\frac{D_1\left(\theta_1(\Psi)+f\right)}{Dt}+A_4\nabla^6\Psi_1=F_1 \quad \text{dans } \Omega\times]0,T[, \tag{3.16}$$

pour la couche de surface ($k=1$) ;

$$\frac{D_k\left(\theta_k(\Psi)+f\right)}{Dt}+A_4\nabla^6\Psi_k=0 \quad \text{dans } \Omega\times]0,T[, \tag{3.17}$$

pour les niveaux intermédiaires ($k=2,\dots,n-1$) ;

$$\frac{D_n\left(\theta_n(\Psi)+f\right)}{Dt}+A_1\Delta\Psi_n+A_4\nabla^6\Psi_n=0, \tag{3.18}$$

dans $\Omega\times]0,T[$, pour la couche du fond ($k=n$).

$\Omega$ représente le bassin de circulation, $\psi_k$ est la fonction de courant dans le niveau $k$, $\theta_k$ est la somme des vorticités dynamique et thermique, $f$ est la force de Coriolis, les termes diffusifs correspondent à la dissipation par frottement latéral et au fond du domaine. Enfin $F_1$ représente l'unique terme source du modèle, provenant du forçage par le vent en surface.

Nous renvoyons à [107, 169, 57] pour plus de détails sur les équations, et à [11] pour les résultats de simulations numériques sur ce modèle (convergence et comparaison du BFN avec le 4D-VAR, études de sensibilité par rapport à différentes perturbations).

### 3.3.4 Conclusions relatives aux expériences numériques

L'algorithme BFN (nudging direct et rétrograde) semble être une méthode d'assimilation de données extrêmement intéressante. Elle est en effet très simple à mettre en œuvre : pas de linéarisation des équations, pas de modèle adjoint, pas d'optimisation. La seule tâche consiste à ajouter le terme de relaxation dans les équations du modèle.

Cet algorithme a été testé et comparé avec la méthode variationnelle sur plusieurs types de systèmes non linéaires : Lorenz (équation différentielle ordinaire, chaotique), Burgers (équation aux dérivées partielles en dimension 1), Saint-Venant ou shallow water (dimension 2), modèle quasi-géostrophique (dimension 3). La conclusion de toutes ces expériences numériques est que notre algorithme converge beaucoup plus rapidement que la méthode variationnelle, et fournit de bien meilleurs résultats dans le même temps, pour les toutes premières itérations. La condition initiale est généralement moyennement bien identifiée, mais la solution correspondante à la fin de la période d'assimilation est nettement meilleure. C'est le point clé pour la phase de prévision, et notre algorithme se révèle beaucoup plus efficace pour obtenir de très bonnes prévisions, qui restent fiables dans le temps.

Les deux algorithmes (BFN et variationnel) peuvent être combinés de la façon suivante : puisque le BFN identifie à chaque itération une estimation de la condition initiale, il est possible d'appliquer quelques itérations du BFN, puis utiliser la solution ainsi identifiée comme point de départ pour la minimisation de l'algorithme variationnel. Cela a pour effet d'accélérer assez nettement la convergence de la méthode variationnelle, puisqu'il faut généralement 2 fois moins d'itérations pour converger, alors que seules quelques (2 à 5) itérations de BFN ont été réalisées au départ. Sans atteindre la convergence (i.e. lorsque le temps de calcul imparti ne permet de faire qu'un nombre assez faible d'itérations), l'utilisation du BFN en préconditionneur de la méthode variationnelle permet d'obtenir de bien meilleurs résultats qu'en faisant le nombre total d'itérations avec uniquement la méthode variationnelle.

Enfin, le nudging direct et rétrograde permet de prendre en compte l'erreur modèle de façon inhérente (sans coût additionnel) puisqu'il consiste justement à voir le modèle comme une contrainte faible et non comme une contrainte forte (comme c'est généralement le cas dans les méthodes variationnelles), la correction du modèle étant réalisée à l'aide des observations.

## 3.4 Principaux résultats théoriques de convergence [18, 24]

### 3.4.1 Cas linéaire

Nous nous plaçons d'abord dans un cadre linéaire simple, qui permet de comprendre la situation. On suppose ici que l'opérateur d'observation $C$ est égal à l'identité, et que le modèle $F$ est linéaire. On suppose également que $K$ et $F$ commutent, ce qui est a priori le cas dans nos applications puisqu'on choisit souvent $K$ proportionnel à la matrice identité. Dans ce cadre là, il est possible d'expliciter la solution de l'algorithme du nudging direct et rétrograde. Afin de simplifier considérablement les équations, on suppose ici que $K' = K$, mais les résultats présentés ici restent vrais dans le cas général. À l'itération $n$, $n > 1$, en supposant $T > 0$, on a

$$\begin{aligned} X_n(0) &= \left(I - e^{-2KT}\right)^{-1}\left(I - e^{-2nKT}\right)\int_0^T \left(e^{-(K+F)s} + e^{-2KT}e^{(K-F)s}\right) K X_{obs}(s)ds \\ &\quad + e^{-2nKT}x_0 \end{aligned} \tag{3.19}$$

et pour tout $t \in [0, T]$,

$$X_n(t) = e^{-(K-F)t}\int_0^t e^{(K-F)s} K X_{obs}(s)ds + e^{-(K-F)t}X_n(0). \tag{3.20}$$

Le résultat suivant donne alors l'existence de la trajectoire limite, et explicite sa forme [18] :

**Théorème 3.2** *Si $n \to +\infty$, alors $X_n(0)$ admet une limite, et*

$$\lim_{n\to+\infty} X_n(0) = X_\infty(0) = \left(I - e^{-2KT}\right)^{-1}\int_0^T \left(e^{-(K+F)s} + e^{-2KT}e^{(K-F)s}\right) K X_{obs}(s)ds. \tag{3.21}$$

*De plus, si $T > 0$, pour tout $t \in [0, T]$,*

$$\lim_{n\to+\infty} X_n(t) = X_\infty(t) = e^{-(K-F)t}\int_0^t e^{(K-F)s} K X_{obs}(s)ds + e^{-(K-F)t}X_\infty(0). \tag{3.22}$$

Sous les mêmes hypothèses, on peut prouver un résultat similaire pour les trajectoires rétrogrades, i.e. il existe une trajectoire limite $\tilde{X}_\infty(t)$ telle que $\lim_{n\to+\infty} \tilde{X}_n(t) = \tilde{X}_\infty(t)$, pour tout $t \in [0,T]$. Cela prouve la convergence de l'algorithme BFN dans ce cadre là.

On peut remarquer que la solution limite (en direct comme en rétrograde) ne dépend plus de la condition initiale $x_0$ utilisée pour initialiser l'algorithme.

Supposons à présent que les observations $X_{obs}$ soient solutions du modèle direct (3.1). Cela implique notamment que pour tout $t \in [0,T]$,

$$X_{obs}(t) = e^{Ft} X_{obs}(0). \tag{3.23}$$

En substituant cette égalité dans les équations (3.21) et (3.22), on montre facilement que pour tout $t \in [0,T]$,

$$\lim_{n\to\infty} X_n(t) = X_{obs}(t). \tag{3.24}$$

À noter que l'algorithme BFN a un comportement similaire sur les opérateurs paraboliques linéaires en dimension infinie, comme par exemple sur l'équation de la chaleur. Une décomposition des trajectoires sur une base de Fourier nous ramène à l'étude d'équations différentielles ordinaires du premier ordre sur les cœfficients, et donne rapidement la convergence de l'algorithme dans ce cas là.

### 3.4.2 Équations de transport

Dans toute cette partie, nous ne considérons qu'une seule itération de l'algorithme BFN, c'est-à-dire une résolution du problème direct avec nudging et une résolution du problème rétrograde avec nudging. Les résultats de décroissance de l'erreur au cours d'une itération peuvent être étendus à un nombre quelconque d'itérations. En effet, les résultats présentés ci-dessous consistent à estimer le rapport entre l'erreur avant et après une itération. Ce rapport est indépendant des itérations, et s'il est strictement inférieur à 1, l'algorithme est contractant et l'erreur décroît exponentiellement vers 0 avec les itérations.

Nous renvoyons à la référence [24] pour les démonstrations de tous ces résultats.

#### Transport linéaire visqueux

On considère tout d'abord le cas d'une équation de transport linéaire avec viscosité.

$$\begin{array}{ll}
(F) & \left\{ \begin{array}{rcl} \partial_t u - \nu\partial_{xx}u + a(x)\partial_x u &=& -K(u-u_{obs}), \\ u|_{x=0} = u|_{x=1} &=& 0, \\ u|_{t=0} &=& u_0, \end{array} \right. \\
(B) & \left\{ \begin{array}{rcl} \partial_t \widetilde{u} - \nu\partial_{xx}\widetilde{u} + a(x)\partial_x \widetilde{u} &=& K'(\widetilde{u}-u_{obs}), \\ \widetilde{u}|_{x=0} = \widetilde{u}|_{x=1} &=& 0, \\ \widetilde{u}|_{t=T} &=& u(T), \end{array} \right.
\end{array} \tag{3.25}$$

avec les notations suivantes, qui restent valables dans la suite :

- la période de temps considérée est $t \in [0, T]$ ;
- la première équation $(F)$ est appelée directe (Forward), et la seconde (B) est appelée rétrograde (Backward) ;
- $K$ et $K'$ sont des fonctions scalaires positives, qui peuvent dépendre du temps $t$ et de la variable d'espace $x$. Pour des raisons de simplicité uniquement, on suppose qu'il existe une constante $\kappa \in \mathbb{R}_+^*$ telle que $K'(t,x) = \kappa K(t,x)$ ;
- le transport $a(x) \in W^{1,\infty}(\Omega)$, $\Omega$ étant le domaine spatial, ici soit l'intervalle $[0,1]$, soit le tore $[0,1]$ ;
- le cœfficient de viscosité $\nu > 0$ est constant ;
- la fonction d'observations $u_{obs}$ est solution de l'équation directe (sans nudging) avec une certaine condition initiale $u_{obs}^0$ :

$$\left\{ \begin{array}{rcl} \partial_t u_{obs} - \nu \partial_{xx} u_{obs} + a(x) \partial_x u_{obs} & = & 0, \\ u|_{x=0} = u|_{x=1} & = & 0, \\ u|_{t=0} & = & u_{obs}^0. \end{array} \right. \tag{3.26}$$

Alors, nous avons le résultat suivant sur les équations de transport linéaire visqueux [24] :

**Théorème 3.3** *On considère une itération de l'algorithme BFN (3.25) avec des observations $u_{obs}$ vérifiant l'équation (3.26). Soient*

$$\begin{array}{rcl} w(t) & = & u(t) - u_{obs}(t), \\ \widetilde{w}(t) & = & \widetilde{u}(t) - u_{obs}(t), \end{array} \tag{3.27}$$

*les erreurs directe et rétrograde.*

1. *Si $K(t,x) = K$, alors pour tout $t \in [0,T]$ :*

$$\widetilde{w}(t) = e^{(-K-K')(T-t)} w(t). \tag{3.28}$$

2. *Si $K(t,x) = K(x)$, avec Support $(K) \subset [a,b]$ où $a < b$ et $a \neq 0$ ou $b \neq 1$, alors l'équation (3.25) est mal posée : il n'existe en général pas de solution $(u, \widetilde{u})$.*
3. *Si $K(t,x) = K \mathbb{1}_{[t_1,t_2]}(t)$ avec $0 \leq t_1 < t_2 \leq T$, alors*

$$\widetilde{w}(0) = e^{(-K-K')(t_2-t_1)} w(0). \tag{3.29}$$

Ce résultat montre que dans le cas des équations de transport linéaire visqueux, l'algorithme BFN converge à condition que le terme de rappel agisse en tout point du domaine. Par exemple, dans le premier cas du théorème 3.3, l'équation (3.28) indique que l'erreur a été réduite d'un facteur $e^{(-K-K')T}$ au cours d'une itération. Par conséquent, l'erreur décroît d'un facteur $e^{-N(K+K')T}$ au cours de $N$ itérations, et la stricte positivité de $K$ (ou $K'$) et $T$ montre la convergence de l'algorithme dans ce cas. Si par contre les observations ne sont pas disponibles partout (i.e. le support de $K$ ne recouvre pas tout le domaine), l'algorithme ne converge pas car le terme de diffusion ne peut pas être stabilisé et le problème rétrograde est alors mal posé.

### Burgers visqueux

On considère désormais une équation classique de transport non linéaire, l'équation de Burgers, dans un cas visqueux. On se place également dans le cas d'une seule itération de l'algorithme BFN :

$$
\begin{aligned}
&(F)\quad \left\{\begin{array}{rcl}
\partial_t u - \nu\partial_{xx}u + u\partial_x u &=& -K(u-u_{obs}),\\
u|_{x=0} = u|_{x=1} &=& 0,\\
u|_{t=0} &=& u_0,
\end{array}\right.\\
&(B)\quad \left\{\begin{array}{rcl}
\partial_t \widetilde{u} - \nu\partial_{xx}\widetilde{u} + \widetilde{u}\partial_x \widetilde{u} &=& K'(\widetilde{u}-u_{obs}),\\
\widetilde{u}|_{x=0} = \widetilde{u}|_{x=1} &=& 0,\\
\widetilde{u}|_{t=T} &=& u(T),
\end{array}\right.
\end{aligned}
\tag{3.30}
$$

avec les mêmes notations que précédemment. On suppose également que les observations $u_{obs}$ sont solution de l'équation de Burgers directe :

$$
\left\{\begin{array}{rcl}
\partial_t u_{obs} - \nu\partial_{xx}u_{obs} + u_{obs}\partial_x u_{obs} &=& 0,\\
u|_{x=0} = u|_{x=1} &=& 0,\\
u|_{t=0} &=& u^0_{obs}.
\end{array}\right.
\tag{3.31}
$$

Alors nous avons le résultat suivant [24] :

**Théorème 3.4** *Si la fonction $K$ n'est pas identiquement nulle, une itération de BFN (3.30) pour l'équation de Burgers visqueux, avec des observations $u_{obs}$ vérifiant l'équation (3.31), est mal posée au sens où il n'existe pas de solution $(u, \widetilde{u})$.*

Dans le cas particulier où $K = K' = 0$, le problème rétrograde est mal posé au sens d'Hadamard (non continuité par rapport à la donnée finale), mais il admet une unique solution si la condition finale $\widetilde{u}|_{t=T}$ est égale à la solution finale du modèle direct. De plus, dans ce cas particulier, la solution rétrograde est exactement égale à la solution directe : $\widetilde{u}(t) = u(t)$ pour tout $t \in [0, T]$. Le résultat est le suivant [24] :

**Proposition 3.1** *Si $K = K' = 0$, alors le problème (3.30) est bien posé au sens de Hadamard, et il existe une unique solution $(u, \widetilde{u})$. De plus, $u = \widetilde{u}$.*

L'algorithme BFN est donc mal posé (sauf si $K = K' = 0$) dans le cas des équations de Burgers visqueux, au sens où il n'existe pas de solution au problème rétrograde. Toutefois, d'un point de vue numérique, nous avons obtenu de très bons résultats sur ce type de modèle [11]. Cela s'explique vraisemblablement par le caractère discret du problème effectivement résolu.

### Transport linéaire non visqueux

On considère désormais le cas non visqueux des équations de transport linéaire. Les équations du BFN sont :

$$
\begin{array}{ll}
(F) & \left\{ \begin{array}{rcl} \partial_t u + a(x)\partial_x u & = & -K(u - u_{obs}), \\ u|_{x=0} & = & u|_{x=1}, \\ \partial_x u|_{x=0} & = & \partial_x u|_{x=1}, \\ u|_{t=0} & = & u_0, \end{array} \right. \\
(B) & \left\{ \begin{array}{rcl} \partial_t \widetilde{u} + a(x)\partial_x \widetilde{u} & = & K'(\widetilde{u} - u_{obs}), \\ \widetilde{u}|_{x=0} & = & \widetilde{u}|_{x=1}, \\ \partial_x \widetilde{u}|_{x=0} & = & \partial_x \widetilde{u}|_{x=1}, \\ \widetilde{u}|_{t=T} & = & u(T), \end{array} \right.
\end{array}
\tag{3.32}
$$

où le transport $a(x)$ peut être constant ou non. Nous avons alors le résultat suivant [24] :

**Théorème 3.5** *On considère une itération du BFN sur le modèle (3.32), avec des observations $u_{obs}$ qui vérifient l'équation (3.32-F) avec $K = 0$. On note*

$$
\begin{array}{rcl} w(t) & = & u(t) - u_{obs}(t), \\ \widetilde{w}(t) & = & \widetilde{u}(t) - u_{obs}(t). \end{array}
\tag{3.33}
$$

*Soit*

$$(s, \psi(s, x)) \tag{3.34}$$

*la courbe caractéristique associée à l'équation (3.32-F) avec $K = 0$, issue de $x$ à l'instant $s = 0$, i.e. telle que*

$$(s, \psi(s, x))|_{s=0} = (0, x). \tag{3.35}$$

*On suppose que le temps final $T$ est tel que les caractéristiques sont bien définies et ne se croisent pas sur $[0, T]$. Alors nous avons les résultats suivants :*

1. *Si $K(t, x) = K$, alors pour tout $t \in [0, T]$,*

$$\widetilde{w}(t) = w(t) e^{(-K-K')(T-t)}. \tag{3.36}$$

2. *Si $K(t, x) = K \mathbb{1}_{[t_1, t_2]}(t)$ avec $0 \leq t_1 < t_2 \leq T$, alors*

$$\widetilde{w}(0) = w(0) e^{(-K-K')(t_2-t_1)}. \tag{3.37}$$

3. *Si $K(t, x) = K(x)$, alors pour tout $t \in [0, T]$,*

$$\widetilde{w}(t, \psi(t, x)) = w(t, \psi(t, x)) \exp\left( - \int_t^T K(\psi(s, x)) + K'(\psi(s, x))\, ds \right). \tag{3.38}$$

On en déduit que dans le cas où le terme de rappel agit sur tout le domaine (cas 1 et 2 du théorème précédent), l'algorithme BFN converge pour les équations de transport linéaire non visqueux. De plus, dans le cas où le support de $K$ ne recouvre pas tout le domaine (cas 3, par exemple lorsque les observations sont partielles), l'algorithme converge dès que toutes les courbes caractéristiques rencontrent le support de $K$, ce qui est par exemple garanti par l'observabilité du système (voir les remarques après la proposition 3.2 ci-dessous).

### Burgers non visqueux

On considère finalement le cas de l'équation de Burgers non visqueux, avec des conditions périodiques, pour un temps $T$ tel qu'aucun choc n'apparaisse sur l'intervalle $[0,T]$ :

$$\begin{array}{ll} (F) & \left\{ \begin{array}{rcl} \partial_t u + u\partial_x u &=& -K(u-u_{obs}), \\ u|_{x=0} &=& u|_{x=1}, \\ \partial_x u|_{x=0} &=& \partial_x u|_{x=1}, \\ u|_{t=0} &=& u_0, \end{array} \right. \\ (B) & \left\{ \begin{array}{rcl} \partial_t \widetilde{u} + \widetilde{u}\partial_x \widetilde{u} &=& K'(\widetilde{u}-u_{obs}), \\ \widetilde{u}|_{x=0} &=& \widetilde{u}|_{x=1}, \\ \partial_x \widetilde{u}|_{x=0} &=& \partial_x \widetilde{u}|_{x=1}, \\ \widetilde{u}|_{t=T} &=& u(T). \end{array} \right. \end{array} \tag{3.39}$$

On a alors les résultats suivants [24] :

**Théorème 3.6** *On considère une itération du BFN correspondant au problème (3.39), avec des observations $u_{obs}$ qui vérifient (3.39-F) avec $K=0$. On note*

$$\begin{array}{rcl} w(t) &=& u(t)-u_{obs}(t), \\ \widetilde{w}(t) &=& \widetilde{u}(t)-u_{obs}(t). \end{array} \tag{3.40}$$

*On suppose que $u_{obs} \in W^{1,\infty}([0,T]\times\Omega)$, i.e. qu'il existe $M>0$ telle que*

$$|\partial_x u_{obs}(t,x)| \leq M, \quad \forall t \in [0,T], \forall x \in \Omega. \tag{3.41}$$

*Alors, nous avons les résultats suivants :*

1. *Si $K(t,x)=K$, alors pour tout $t\in[0,T]$,*

$$\|\widetilde{w}(t)\| \leq e^{(-K-K'+M)(T-t)}\|w(t)\|. \tag{3.42}$$

2. *Si $K(t,x)=K\mathbb{1}_{[t_1,t_2]}(t)$ avec $0\leq t_1<t_2\leq T$, alors*

$$\|\widetilde{w}(0)\| \leq e^{(-K-K')(t_2-t_1)+MT}\|w(0)\|. \tag{3.43}$$

**Proposition 3.2** *On considère une résolution du problème direct (resp. rétrograde) avec nudging pour l'équation de Burgers non visqueux (3.39-F) (resp. (3.39-B)). Avec les notations du théorème 3.6, si $K(t,x)=K(x)$, alors*

$$w(T,\psi(T,x)) = w(0,x)\exp\left(-\int_0^T K(\psi(\sigma,x))d\sigma - \int_0^T \partial_x u_{obs}(\sigma,\psi(\sigma,x))d\sigma\right). \tag{3.44}$$

**Remarque :** Dans le cas particulier où $K(t,x) = K(x) = K\mathbb{1}_{[a,b]}(x)$, où $K$ est une constante et $[a,b]$ est un sous-intervalle de $[0,1]$, alors

$$w(T, \psi(T,x)) = w(0,x) \exp\left(-K\chi(x) - \int_0^T \partial_x u_{obs}(\sigma, \psi(\sigma,x)) d\sigma\right), \tag{3.45}$$

où la fonction $\chi$ est définie par

$$\chi(x) = \int_0^T \mathbb{1}_{Supp(K)}(\psi(\sigma,x)) d\sigma \tag{3.46}$$

et désigne le temps pendant lequel la caractéristique $\psi(\sigma,x)$ issue de $x$ correspondant à l'équation (3.39-F) avec $K = 0$ appartient au support de $K$.

On peut remarquer que le système est observable si et seulement si la fonction $\chi$ admet une borne inférieure strictement positive, i.e. $m := \min_x \chi(x) > 0$, l'observabilité étant définie par (voir [147]) :

$$\exists C, \forall u \text{ solution de (3.39-F) avec } K = 0, \quad \|u(T,.)\|^2 \leq C \int_0^T \|K(.)u(s,.)\|^2 \, ds. \tag{3.47}$$

Dans ce cas, la proposition 3.2 démontre la décroissance exponentielle globale (en espace) de l'erreur, du moment que $K$ est plus grand que $\frac{MT}{m}$, où $M$ est défini par l'équation (3.41).

Nous pouvons déduire de cette remarque que, si à chaque itération, à la fois pour la résolution du problème direct et du problème rétrograde, la condition d'observabilité est satisfaite, alors l'algorithme converge (l'erreur décroît exponentiellement vers 0). Il faut toutefois noter que c'est une condition suffisante mais non nécessaire, car même si $\chi(x) = 0$, la dernière exponentielle de l'équation (3.45) est bornée.

### Remarques sur les résultats théoriques

Dans certaines applications géophysiques (par exemple la météorologie), il n'y a pas de viscosité dans les équations des modèles numériques. Dans ce cas, en supposant que la condition d'observabilité est vérifiée, alors l'algorithme du nudging direct et rétrograde est bien posé, et le théorème 3.6 et la proposition 3.2 montrent que la solution ainsi reconstruite tend vers la trajectoire des observations partout, et pas seulement sur le support de $K$.

D'un point de vue numérique, on peut effectivement constater que même si les observations sont discrètes et éparpillées en temps et en espace, la solution numérique est corrigée partout et pas uniquement là où les observations agissent. Enfin, si le cœfficient de viscosité reste petit, nous avons également constaté que le comportement de l'algorithme reste le même, et la convergence a lieu partout [11].

## 3.5 Lien avec les observateurs [25]

Dans cette section, nous faisons le lien entre le nudging standard et les observateurs du type Luenberger [129, 114]. En utilisant les symétries intrinsèques des modèles physiques, il est possible de définir une classe d'observateurs invariants par ces symétries. Il paraît en effet judicieux de préserver les symétries du modèle lors de l'ajout d'un terme de rappel dans les équations. L'étude des observateurs nous permet également d'améliorer la méthode du nudging dans le cadre d'un modèle shallow water, en corrigeant notamment les variables non observées grâce à un observateur invariant. Cette section résume les travaux contenus dans [25].

### 3.5.1 Observateurs pour un modèle shallow water

Nous considérons ici un modèle shallow water, analogue à celui présenté dans la section 3.3.3. Nous le réécrivons toutefois sous une autre forme afin de mieux faire apparaître les symétries (nous renvoyons à [113] pour plus de détails). Dans la suite, $h$ désigne toujours la hauteur du fluide, tandis que $v$ est désormais le vecteur vitesse à deux composantes. Les équations considérées sont les suivantes, l'équation vectorielle pour la vitesse :

$$\frac{\partial(hv)}{\partial t} + (\nabla.(hv) + (hv).\nabla)\, v = -g'h\nabla h - k \times f(hv) + (A\nabla^2 - R)(hv) + \frac{\tilde{\tau}}{\rho} i, \tag{3.48}$$

et l'équation scalaire pour la hauteur d'eau :

$$\frac{\partial h}{\partial t} = -\nabla.(hv). \tag{3.49}$$

Dans ces équations, $g'$ désigne la gravité réduite, $\rho$ est la densité du fluide, $f$ la force de Coriolis, $i$ est le vecteur unitaire longitudinal (pointant vers l'est) et $k$ est le vecteur vertical (altitude). Enfin, $A$, $R$ et $\tilde{\tau}$ sont les cœfficients de viscosité latérale, friction et forçage par le vent respectivement.

On suppose que seule la hauteur d'eau $h$ est observée, afin de se placer dans une situation réaliste où l'immense majorité des observations d'un océan proviennent des observations satellitaires, directement reliées à la hauteur d'eau de surface.

Un observateur $(\hat{h}, \hat{v})$ du système (3.48-3.49) est solution des équations suivantes :

$$\frac{\partial(\hat{h}\hat{v})}{\partial t} + \Big(\nabla.(\hat{h}\hat{v}) + (\hat{h}\hat{v}).\nabla\Big)\, \hat{v} = -g'\hat{h}\nabla\hat{h} - k \times f(\hat{h}\hat{v}) + (A\nabla^2 - R)(\hat{h}\hat{v}) + \frac{\tilde{\tau}}{\rho} i + F_v(h, \hat{v}, \hat{h}) \tag{3.50}$$

et

$$\frac{\partial \hat{h}}{\partial t} = -\nabla.(\hat{h}\hat{v}) + F_h(h, \hat{v}, \hat{h}). \tag{3.51}$$

Les observateurs $(\hat{h}, \hat{v})$ vérifient les mêmes équations que les solutions réelles $(h, v)$, à ceci près qu'elles contiennent un terme de rappel, $F_v(h, \hat{v}, \hat{h})$ dans l'équation sur la vitesse, et $F_h(h, \hat{v}, \hat{h})$ dans l'équation sur la hauteur d'eau. Le choix de $F_v$ et $F_h$ est d'abord dicté par le fait que les observateurs doivent tendre vers la solution exacte et jouent donc le rôle de termes de rappel vers la solution, mais aussi par les symétries du système. Ces termes de rappel doivent notamment être nuls lorsque $\hat{h} = h$ et $\hat{v} = v$.

### 3.5.2 Utilisation des symétries du modèle

Le modèle shallow water que nous considérons est invariant par rotation et translation, puisqu'il ne dépend ni de l'orientation, ni de l'origine du repère. Par conséquent, nous allons chercher des termes de rappel qui conservent ces invariances. La conception d'observateurs invariants (ou qui préservent les symétries du modèle) a été très récemment introduite, essentiellement pour des problèmes industriels [28, 59]. L'idée sous-jacente est évidemment de ne pas perturber les symétries du modèle lors de l'ajout du terme de rappel vers les données dans les équations de l'observateur.

Pour le terme scalaire portant sur la hauteur d'eau, un résultat de calcul différentiel assure que tout opérateur différentiel scalaire invariant par rotation et translation s'écrit comme un polynôme du Laplacien [155]. Par des considérations d'invariance par rotation [139], on arrive ainsi à une famille d'opérateurs scalaires de la forme

$$F_h = Q_1(\Delta, h, |\hat{v}|^2, \hat{h} - h) + \nabla\left(Q_2(\Delta, h, |\hat{v}|^2, \hat{h} - h)\right).\hat{v} + f_h, \tag{3.52}$$

où $Q_1$ et $Q_2$ sont deux polynômes scalaires du Laplacien, et $f_h$ est un terme intégral. Plus précisément,

$$Q_i(\Delta, h, |\hat{v}|^2, \hat{h} - h) = \sum_{k=0}^{N} a_k^i(h, |\hat{v}|^2, \hat{h} - h)\Delta^k\left(b_k^i(h, |\hat{v}|^2, \hat{h} - h)\right), \tag{3.53}$$

où $a_k^i$ et $b_k^i$ sont des fonctions scalaires régulières qui s'annulent lorsque le troisième argument est nul. De même, le terme de rappel vectoriel sur la vitesse est de la forme

$$F_v = P_1(\Delta, h, |\hat{v}|^2, \hat{h} - h)\hat{v} + \nabla\left(P_2(\Delta, h, |\hat{v}|^2, \hat{h} - h)\right) + f_v, \tag{3.54}$$

où les polynômes $P_1$ et $P_2$ sont du même type que $Q_1$ et $Q_2$.

On définit alors les termes intégraux $f_v$ et $f_h$ de sorte qu'ils soient eux aussi invariants par rotation et translation. On obtient ainsi les formulations suivantes :

$$f_v(x,y,t) = \iint \left[R_1(\Delta, h, |\hat{v}|^2, \hat{h} - h)\hat{v} + \nabla\left(R_2(\Delta, h, |\hat{v}|^2, \hat{h} - h)\right)\right]_{(x-\xi, y-\zeta, t)} \phi_v(\xi^2+\zeta^2)\, d\xi d\zeta, \tag{3.55}$$

$$f_h(x,y,t) = \iint \left[S_1(\Delta, h, |\hat{v}|^2, \hat{h} - h) + \nabla\left(S_2(\Delta, h, |\hat{v}|^2, \hat{h} - h)\right).\hat{v}\right]_{(x-\xi, y-\zeta, t)} \phi_h(\xi^2+\zeta^2)\, d\xi d\zeta, \tag{3.56}$$

où les polynômes $R_i$ et $S_i$ sont définis commes les polynômes $Q_i$ et $P_i$.

Les supports de $\phi_v$ et $\phi_h$ permettent de définir une zone d'influence où cela a un sens de corriger l'observateur avec les valeurs observées. Dans le cas particulier où ces fonctions sont des fonctions Dirac, alors les fonctions $f_v$ et $f_h$ deviennent équivalentes aux autres termes de $F_v$ et $F_h$.

### 3.5.3 Étude d'une classe d'observateurs dans un cas linéarisé

Nous considérons maintenant un cas simple, où on ne garde que les termes intégraux des termes de correction : $Q_1 = Q_2 = P_1 = P_2 = 0$, $R_1 = S_2 = 0$ et $S_2 = R_1 = h - \hat{h}$. En effet, les mesures de la hauteur d'eau étant généralement bruitées, on ne souhaite pas les dériver, ce qui amplifierait considérablement le bruit.

Sans perdre en généralité, on peut également simplifier les équations du modèle (pas de force de Coriolis, pas de friction ou dissipation par viscosité, pas de forçage par le vent, ...). L'observateur pour le système simplifié est alors :

$$\frac{\partial \hat{h}}{\partial t} = -\nabla(\hat{h}\hat{v}) + \phi_h * (h - \hat{h}), \tag{3.57}$$

$$\frac{\partial \hat{v}}{\partial t} = -\hat{v}\nabla\hat{v} - g\nabla\hat{h} + \phi_v * \nabla(h - \hat{h}). \tag{3.58}$$

Dans le cas dégénéré où $\phi_h = K_h\delta_0$ et $\phi_v = K_v\delta_0$, $K_h$ et $K_v$ étant des scalaires positifs et $\delta_0$ représentant la mesure de Dirac en 0, les équations (3.57-3.58) deviennent la formulation classique du nudging (ou observateur de Luenberger).

On suppose par ailleurs que le système est proche d'un état d'équilibre, et on ne considèrera que des petites vitesses $\delta v = v - \bar{v} \ll \sqrt{g\bar{h}}$ et $\delta h = h - \bar{h} \ll \bar{h}$, où $\bar{h}$ et $\bar{v} = 0$ représentent respectivement la hauteur d'eau et vitesse du fluide au point d'équilibre. En notant $\tilde{h}$ et $\tilde{v}$ l'écart (sur la hauteur d'eau et la vitesse respectivement) entre l'observateur et la solution réelle du système simplifié et linéarisé, alors ces erreurs d'estimation vérifient les équations linéaires suivantes :

$$\frac{\partial \tilde{h}}{\partial t} = -\bar{h}\nabla\tilde{v} - \phi_h * \tilde{h}, \tag{3.59}$$

$$\frac{\partial \tilde{v}}{\partial t} = -g\nabla\tilde{h} - \phi_v * \nabla\tilde{h}. \tag{3.60}$$

Un choix raisonnable pour les fonctions $\phi_h$ et $\phi_v$ consiste à prendre des noyaux gaussiens de la forme :

$$\phi_h(x,y) = \beta_h\, exp(-\alpha_h(x^2+y^2)), \tag{3.61}$$

$$\phi_v(x,y) = \beta_v\, exp(-\alpha_v(x^2+y^2)), \tag{3.62}$$

puisqu'on suppose généralement que les mesures sont entachées d'un bruit blanc gaussien. Toutefois, les résultats de convergence obtenus ci-dessous peuvent s'étendre au cas de noyaux qui s'écrivent sous la forme :

$$\phi_h(x,y) = (f(x) * f(x))\,(f(y) * f(y)), \tag{3.63}$$

$$\phi_v(x,y) = (g(x) * g(x))\,(g(y) * g(y)), \tag{3.64}$$

$$\tag{3.65}$$

où $f$ et $g$ sont des fonctions lisses (par exemple $C^\infty$), et dont les cœfficients de Fourier sont réels et strictement positifs.

En éliminant la vitesse du système d'équations (3.59-3.60), la hauteur vérifie l'équation suivante :

$$\frac{\partial^2 \tilde{h}}{\partial t^2} = g\bar{h}\Delta\tilde{h} + \bar{h}\,\phi_v * \Delta\tilde{h} - \phi_h * \frac{\partial \tilde{h}}{\partial t}, \tag{3.66}$$

qui s'apparente à une équation des ondes amortie, avec un amortissement visqueux externe. Cette équation peut être réécrite sous la forme suivante :

$$\frac{\partial^2 \tilde{h}}{\partial t^2} = \phi_v * \Delta\tilde{h} - \phi_h * \frac{\partial \tilde{h}}{\partial t}, \tag{3.67}$$

en redéfinissant

$$\phi_v(x,y) = g\bar{h}\delta_0 + \bar{h}\,\beta_v\, exp(-\alpha_v(x^2+y^2)), \tag{3.68}$$

où $\delta_0$ représente la mesure de Dirac en 0.

Dans ces conditions, nous avons le résultat suivant [25] :

**Théorème 3.7**

$$\lim_{t\to+\infty} \int_\Omega \left( \|\nabla\tilde{h}\|^2 + \left|\frac{\partial\tilde{h}}{\partial t}\right|^2 \right) = 0, \tag{3.69}$$

qui montre la convergence forte de l'erreur vers 0 pour le système linéarisé, et donc de l'observateur vers la solution réelle. La convergence est alors étendue sans difficulté à la vitesse $v$.

Une analyse dimensionnelle permet également de prédire les valeurs typiques des cœfficients de gain des fonctions $\phi_h$ et $\phi_v$ (cf équations (3.61) et (3.68)) :

$$\beta_h = 2\xi_0\omega_0, \quad \bar{h}\beta_v = L_0^2\omega_0^2 - g\bar{h}, \tag{3.70}$$

où $\omega_0$ et $L_0$ représentent respectivement la pulsation et la taille caractéristiques de l'écoulement, et $\xi_0$ représente le cœfficient d'amortissement du système. De même, les cœfficients $\alpha_h$ et $\alpha_v$ peuvent être déterminés a priori, en imposant $\alpha_h^{-2} = \alpha_v^{-2}$, cette valeur étant égale à la taille caractéristique de la région d'influence des observations. Cette dernière dépend essentiellement de la quantité de bruit sur les mesures, mais aussi de la répartition spatiale des données.

### 3.5.4 Expériences numériques

Nous avons testé numériquement la classe d'observateurs définis par l'ajout d'un terme de rappel égal à $\phi_h * (h - \hat{h})$ pour la hauteur d'eau, et $\phi_v * \nabla(h - \hat{h})$ pour la vitesse, où les fonctions $\phi_h$ et $\phi_v$ sont définies par les équations (3.61) et (3.62). Différents tests ont été réalisés sur un modèle shallow water, d'abord sur la version simplifiée et linéarisée autour de l'état d'équilibre, puis sur le modèle complet et non linéaire. Nous avons également testé différentes valeurs des paramètres $\alpha_h$, $\alpha_v$, $\beta_h$ et $\beta_v$, avec des données plus ou moins bruitées. Nous avons notamment comparé cet observateur avec le nudging classique, ou observateur de Luenberger, en prenant des valeurs très grandes pour les cœfficients $\alpha$ [25].

Les conclusions de ces expériences numériques sont doubles. D'une part, grâce à la convolution avec un noyau gaussien, il est possible d'améliorer sensiblement les résultats du nudging classique, en filtrant beaucoup mieux les erreurs de mesure et en répartissant spatialement l'information contenue dans les observations. D'autre part, les simulations numériques montrent que le terme de rappel dans les équations sur la vitesse permet de corriger cette dernière assez nettement, en utilisant pourtant uniquement des mesures sur la hauteur d'eau. De plus, comme la vitesse est beaucoup plus corrigée que lorsqu'aucun terme de rappel n'est ajouté, la hauteur d'eau se trouve être elle aussi mieux corrigée, grâce au couplage entre les variables d'état.

Il faut noter que cette méthode est à peine plus coûteuse que le nudging standard, puisqu'il suffit en pratique de tronquer la convolution avec le noyau gaussien en dehors d'un disque de quelques points autour de chaque observation (avec un rayon de l'ordre de 5 à 10 dans nos expériences).

Une perspective à court terme consiste à tester cette méthode sur le système rétrograde, ce qui permettrait par la suite d'améliorer l'algorithme BFN en le rendant encore moins sensible aux bruits sur les observations, et surtout en corrigeant davantage les variables non observées grâces aux variables observées.

## 3.6 Conclusion

L'algorithme du nudging direct et rétrograde que nous avons introduit s'avère être une méthode d'assimilation de données extrêmement prometteuse. Cette méthode est en effet particulièrement facile à mettre en œuvre puisqu'elle ne nécessite aucune linéarisation d'opérateurs, pas d'état adjoint, et pas d'algorithme d'optimisation. La seule tâche qui incombe à l'utilisateur est d'ajouter le terme de relaxation dans les équations du modèle.

Le BFN a été comparé avec la méthode variationnelle sur beaucoup de modèles géophysiques plus ou moins simplifiés, et plus ou moins turbulents. Les conclusions de cette approche expérimentale sont que le BFN converge très rapidement et qu'il produit généralement de meilleurs résultats que la méthode variationnelle dans le même temps. La condition initiale n'est pas toujours très bien identifiée, mais la solution à l'instant final de la période d'assimilation est toujours bien mieux identifiée, ce qui conduit à des prévisions de très grande qualité, qui restent fiables assez longtemps après la fin de la période d'assimilation.

Nous avons également étudié la possibilité d'hybrider notre algorithme avec la méthode variationnelle, en commençant par réaliser quelques itérations du BFN avant de continuer avec la méthode variationnelle en partant de la trajectoire identifiée par le BFN. Ceci peut être vu comme un préconditionnement de la méthode variationnelle par le BFN, ce qui conduit à l'accélérer très sensiblement.

Enfin, certains résultats théoriques viennent justifier ou expliquer la méthode sur des cas simples.

Plusieurs pistes de recherche sont à l'étude, et particulièrement la possibilité d'améliorer les matrices (ou cœfficients) de nudging, par exemple en les faisant varier avec les itérations, ou avec le temps, mais toujours avec l'idée de conserver la simplicité de la méthode.

# Chapitre 4

# Assimilation de données images

Ce chapitre résume les travaux contenus dans [23].

## 4.1 Introduction

À l'interface des deux précédents thèmes que nous avons considérés (assimilation de données et traitement d'images), nous avons étudié la problématique de l'assimilation de données images. Dans le contexte actuel de l'assimilation de données en géophysique, de très nombreuses sources d'observations sont utilisées, même si la plupart des mesures sont fournies par les satellites d'observation de la Terre. Toutefois, une quantité importante de données ne sont pas (ou très peu) utilisées : les images satellitaires. Certains satellites sont entièrement dédiés à la prise de vues, comme par exemple le satellite SPOT, mais d'autres satellites d'observation plus spécifiquement conçus pour mesurer diverses quantités physiques prennent également des clichés à très haute résolution de leur zone d'observation [62].

Plusieurs idées ont récemment été développées pour assimiler les images satellites. Une première idée consiste à identifier certaines structures caractéristiques de l'image (comme par exemple les zones tourbillonnaires, ou les filaments) et les suivre dans le temps. Cette technique est actuellement utilisée en météorologie [135]. Une approche que l'on pourrait qualifier de duale, consiste à créer des images modèle (i.e. produites par le modèle mathématique), puis à les comparer avec les images réelles, en utilisant par exemple des méthodes spectrales [132].

Nous proposons ici d'utiliser une méthode extrêmement rapide et efficace pour identifier et extraire les champs de vitesse du fluide à partir de séquences d'images. Pour chaque paire d'images considérées, nous estimons un champ complet de la vitesse, avec un vecteur vitesse par pixel de l'image. Si l'on considère que la plupart des satellites fournissent des images à très haute résolution (de l'ordre de $5000 \times 5000$ pixels) toutes les 15 à 60 minutes, la quantité de données ainsi extraites devient comparable à l'ensemble des données actuellement assimilées en océanographie [103] ! Toutefois, comme nous allons le voir, les champs de vitesse extraits ne sont utilisables (et certifiés par un certain niveau de qualité) que dans certaines zones de l'image.

Nous nous appuyons sur l'hypothèse de conservation de l'intensité lumineuse. Cette hypothèse a été introduite dans [109], et linéarisée dans les approches basées sur le flot optique [128, 39, 47]. Cette hypothèse est parfois remplacée par une équation de continuité afin de prendre en compte l'étalement des souces lumineuses [93, 94, 76, 122].

Dans notre approche, nous proposons de ne pas linéariser l'hypothèse de conservation de l'intensité lumineuse, et de considérer au contraire une fonction coût non linéaire qui mesure la qualité du recalage entre deux images, et qui tient compte du fait que les images sont disponibles de façon discrète en temps. Puis nous considérons différentes régularisations du problème, ainsi qu'une approche multi-grille pour accélérer l'identification. De nombreux tests numériques ont été réalisés avec des images synthétiques puis réelles, et nous ont permis d'extraire beaucoup plus rapidement des champs de vitesse complets, contrairement à la méthode PIV (Particle Imaging Velocimetry, [27]) qui sert actuellement de référence en mécanique des fluides et océanographie, et qui ne peut fournir qu'un vecteur vitesse par zone d'environ $10 \times 10$ pixels, en un temps plus long.

## 4.2 Modélisation et résolution du problème [23]

Dans cette section, nous définissons un critère à minimiser, et nous présentons les différentes techniques permettant de le minimiser rapidement et efficacement.

### 4.2.1 Conservation de l'intensité lumineuse

Soient deux images $I_0$ et $I_1$ que l'on suppose consécutives dans la séquence (ou le film) des données, et que l'on souhaite recaler. On note $\Omega \subset \mathbb{R}^2$ le domaine de travail correspondant au support de l'image, en supposant sans restriction que les images sont bi-dimensionnelles. Le déplacement entre ces deux images est donné par le champ de vecteurs $(u, v)$ tel qu'en tout point $(x, y) \in \Omega$,

$$I_1(x + u(x, y), y + v(x, y)) = I_0(x, y). \tag{4.1}$$

Il n'y a généralement pas unicité du champ de vecteurs vérifiant (4.1). De plus, le bruit sur les données rend le problème généralement mal posé.

### 4.2.2 Fonction coût

Nous proposons donc de définir une fonction coût mesurant l'écart entre ces deux quantités au sens des moindres carrés :

$$J(u, v) = \frac{1}{2} \int_\Omega [F(I_0, I_1; u, v)(x, y)]^2 \, dxdy + \frac{1}{2} \alpha R(u, v), \tag{4.2}$$

où $R(u, v)$ est un terme de régularisation spatiale à définir, et $\alpha > 0$ est un poids sur la régularisation. Enfin, $F$ est la fonction à annuler :

$$F(I_0, I_1; u, v)(x, y) = I_1(x + u(x, y), y + v(x, y)) - I_0(x, y). \tag{4.3}$$

### 4.2.3 Régularisation

Nous avons étudié différentes régularisations, soit classiques (comme par exemple la norme $L^2$), soit liées à la physique du problème (norme de la divergence) :

$$\begin{aligned}
R_0(u,v) &= \|u\|^2 + \|v\|^2, && (4.4)\\
R_1(u,v) &= \|\nabla u\|^2 + \|\nabla v\|^2 = \|\partial_x u\|^2 + \|\partial_y u\|^2 + \|\partial_x v\|^2 + \|\partial_y v\|^2, && (4.5)\\
R_{div}(u,v) &= \|div(u,v)\|^2 = \|\partial_x u + \partial_y v\|^2, && (4.6)\\
R_{curl}(u,v) &= \|curl(u,v)\|^2 = \|\partial_y u - \partial_x v\|^2, && (4.7)\\
R_{div/curl}(u,v) &= \|div(u,v)\|^2 + \|curl(u,v)\|^2 = \|\partial_x u + \partial_y v\|^2 + \|\partial_y u - \partial_x v\|^2, && (4.8)\\
R_{\nabla div}(u,v) &= \|\nabla div(u,v)\|^2 = \|\partial^2_{xx} u + \partial^2_{xy} v\|^2 + \|\partial^2_{xy} u + \partial^2_{yy} v\|^2, && (4.9)\\
R_{\nabla div/\nabla curl}(u,v) &= \|\nabla div(u,v)\|^2 + \|\nabla curl(u,v)\|^2 && (4.10)\\
&= \|\partial^2_{xx} u + \partial^2_{xy} v\|^2 + \|\partial^2_{xy} u + \partial^2_{yy} v\|^2 + \|\partial^2_{xy} u - \partial^2_{xx} v\|^2 + \|\partial^2_{yy} u - \partial^2_{xy} v\|^2.
\end{aligned}$$

Dans la suite, on notera $R(u,v) = \|S(u,v)\|^2$ où $S$ est un opérateur linéaire. Des cœfficients peuvent être introduits pour pondérer les différents termes entre eux.

### 4.2.4 Approche multi-grille et optimisation

La minimisation de $J$ est réalisée sur une suite de sous-espaces emboités :

$$\mathcal{C}_{16} \subset \mathcal{C}_8 \subset \mathcal{C}_4 \subset \mathcal{C}_2 \subset \mathcal{C}_1, \tag{4.11}$$

où $\mathcal{C}_q$ est l'ensemble des champs de déplacement admissibles à l'échelle $q$, c'est-à-dire affines par morceaux par rapport à chaque variable d'espace sur des carrés de $q \times q$ pixels.

La différence principale avec les approches relevant du flot optique (voir par exemple [134, 146]) est que nous considérons la minimisation de la fonctionnelle non linéaire $J$, ce qui permet notamment d'identifier des champs de déplacement arbitrairement grands, contrairement aux approches où la fonction coût est linéarisée.

Une fois la minimisation réalisée, par exemple sur l'espace $\mathcal{C}_{16}$ (ce qui rend l'optimisation particulièrement rapide, compte tenu de la réduction de la dimension du problème), le champ optimal est utilisé comme point de départ pour la minimisation sur l'espace $\mathcal{C}_8$, et ainsi de suite jusqu'à obtenir un minimum sur l'espace complet $\mathcal{C}_1$, et donc une solution totalement raffinée.

Les différentes optimisations sont faites avec un algorithme de Gauss-Newton. Ainsi, en initialisant la minimisation avec un champ $(u^0, v^0)$ donné (en pratique, un champ nul), cela revient à chercher une mise à jour sous la forme

$$(u^k, v^k) := (u^{k-1}, v^{k-1}) + (du^k, dv^k), \tag{4.12}$$

où $(du^k, dv^k)$ est solution de

$$(DF^T DF + \alpha S^T S)(du, dv) = -DF^T F - \alpha S^T S(u,v), \tag{4.13}$$

où $F = F(I_0, I_1; u^{k-1}, v^{k-1})$ est l'erreur, $DF = DF(I_0, I_1; u^{k-1}, v^{k-1})$ est la matrice jacobienne de l'erreur, et $S$ est l'opérateur linéaire de régularisation.

Une autre nouveauté de notre approche réside dans l'assemblage de la matrice $DF^T DF$, nécessaire à la résolution de l'équation (4.13).

Soit $V$ un champ de vecteurs de $\mathcal{C}_q$, i.e. défini sur la grille constituée de carrés de $q \times q$ pixels. Soit $(\mathbf{e}_i)$ la base canonique orthonormée de $\mathcal{C}_q$, qui contient donc des vecteurs nuls sauf sur une maille où le champ contient des vecteurs unitaires parallèles à l'un des deux axes de $\mathbb{R}^2$. Le cœfficient $(k, l)$ de la matrice $DF^T DF$ est alors égal à

$$(DF^T DF)_{k,l} = (DF^T DF\mathbf{e}_k | \mathbf{e}_l) = (DF\mathbf{e}_k | DF\mathbf{e}_l)_{L^2(\Omega)}. \tag{4.14}$$

Comme les déplacements élémentaires $\mathbf{e}_k$ sont nuls sauf sur une maille, la matrice $DF^T DF$ est donc creuse. En utilisant la formulation suivante de la jacobienne :

$$DF(u, v).d(x, y) = \nabla I_1(x + u(x, y), y + v(x, y)).d(x, y), \tag{4.15}$$

la matrice $DF^T DF$ peut être assemblée de la façon suivante :

$$\begin{aligned}
DF^T DF &= \sum_{k,l} (DF^T DF)_{k,l}\, \mathbf{e}_k \otimes \mathbf{e}_l \\
&= \sum_{k,l} \int_\Omega (\nabla I_1(x') \,|\, \mathbf{e}_k(x))\, (\nabla I_1(x') \,|\, \mathbf{e}_l(x))\, \mathbf{e}_k \otimes \mathbf{e}_l \; dx \\
&= \sum_{k,l} \sum_{R \in \mathcal{R}_q} \int_R (\nabla I_1(x') \,|\, \mathbf{e}_k(x))\, (\nabla I_1(x') \,|\, \mathbf{e}_l(x))\, \mathbf{e}_k \otimes \mathbf{e}_l \; dx \\
&= \sum_{R \in \mathcal{R}_q} \sum_{k,l} \int_R (\nabla I_1(x') \,|\, \mathbf{e}_k(x))\, (\nabla I_1(x') \,|\, \mathbf{e}_l(x))\, \mathbf{e}_k \otimes \mathbf{e}_l \; dx, \quad (4.16)
\end{aligned}$$

avec la notation $x' = x + V(x)$, et en notant $\mathcal{R}_q$ l'ensemble des carrés $q \times q$ de la grille. Le nombre de produits scalaires de la forme $(\nabla I_1(x + V(x)) | \mathbf{e}_k(x))$ à calculer est de 8 pour chaque élément de $\mathcal{R}_q$. Il suffit donc de lire une seule fois les données contenues dans l'image $I_1$ pour remplir la matrice $DF^T DF$. De même, les termes $DF^T F$ et $S^T S$ sont assemblés rapidement avec une technique similaire.

Finalement, l'équation (4.13) est résolue avec une méthode de gradient conjugué sans préconditionnement.

### 4.2.5 Estimateur de qualité des résultats

Comme notre étude consiste à extraire des données dans le but de pouvoir les assimiler, nous proposons de fournir un estimateur de la qualité des résultats obtenus. Cela revient à fournir les statistiques d'erreurs correspondant à ces mesures, comme pour des données classiques.

Nous proposons ainsi la formule suivante pour mesurer la qualité des résultats :

$$e(I_0, I_1; u, v)(x, y) = 1 - \frac{|I_1(x + u(x, y); y + v(x, y)) - I_0(x, y)|}{|I_1(x, y) - I_0(x, y)|} \tag{4.17}$$

si le dénominateur est non nul, et $e(I_0, I_1; u, v) = 0$ sinon.

On voit clairement que si la différence entre les deux images $I_0$ et $I_1$ est grande au départ, et beaucoup moins après application du champ de vitesse, alors le champ identifié est plausible, et la formule (4.17) donne une valeur proche de 1. À l'inverse, si l'identification ne s'est pas bien déroulée, i.e. si l'écart entre les deux images n'a pas baissé, alors $e$ sera proche de 0. De même, s'il n'y a aucun signal dans une région de l'image, alors cela conduit à un estimateur égal à 0. Cela ne veut pas dire que le champ de vitesse identifié n'est pas le bon, mais sans signal, nous ne pouvons pas le certifier.

## 4.3 Simulations numériques [23]

Dans cette section, nous présentons rapidement le contexte des différentes expériences numériques réalisées et présentées dans [23].

### 4.3.1 Données synthétiques

Nous avons tout d'abord testé notre approche sur des données simulées. Nous avons considéré ici un modèle shallow water (ou équations de Saint-Venant), bidimensionnel, qui se trouve détaillé en section 3.3.3 (mais avec différentes valeurs de paramètres) ou dans [23].

Ce modèle est ensuite couplé avec une équation d'advection, indiquant que la concentration en colorant est transportée par la vitesse du fluide :

$$\partial_t c + u\partial_x c + v\partial_y c = 0, \tag{4.18}$$

où $c$ est la concentration d'un traceur passif (par exemple de la chlorophylle ou un polluant chimique tel que le pétrole dans les océans). En fixant une quantité initiale de colorant $c(t = 0)$, nous pouvons ainsi simuler le déplacement réel d'un traceur dans l'océan, et récupérer en sortie du modèle numérique des images de la concentration.

À partir de plusieurs de ces images de concentration, acquises tous les 100 pas de temps par exemple (afin de respecter le fait qu'en pratique, les images satellitaires ne sont pas disponibles à tout instant), nous pouvons extraire des champs de vitesse et les comparer avec les vitesses réelles du modèle.

Comme le montrent les résultats présentés dans [23], nous obtenons grâce à l'approche multi-échelle et au schéma performant d'optimisation (sans information a priori puisque nous l'initialisons avec un champ constant et nul) d'excellents résultats très rapidement. Le recalage entre deux images est presque parfait au bout de quelques itérations. Le champ identifié correspond également au comportement global de l'océan, à savoir une rotation du fluide dûe à la présence d'un tourbillon, avec un léger mouvement de translation puisque le tourbillon se déplace dans les exemples considérés.

On peut remarquer que la régularisation $R_{div}$ (4.6) conduit à des résultats de moins bonne qualité que la régularisation $R_1$ (4.5), alors qu'a priori, le champ de vitesse devrait être à divergence nulle. Cela vient du fait que le champ identifié correspond à une intégrale du champ de vitesse sur les 100 pas de temps qui séparent les deux images, le long des lignes de champ. C'est en quelques sortes un champ de vitesse lagrangien, et qui n'a plus aucune raison d'être à divergence nulle.

Une application intéressante que nous avons pu tester également est le suivi automatique d'une structure caractéristique. Nous avons manuellement défini une boîte autour d'un objet (ici, un tourbillon) sur une des images. Puis, en utilisant pour chaque image la moyenne sur la boîte de la vitesse identifiée par notre algorithme, nous pouvons en déduire un déplacement moyen de la zone d'intérêt entre les deux images. La boîte est ainsi automatiquement décalée, et nous avons ainsi pu suivre sans difficulté et entièrement automatiquement le tourbillon sur une centaine d'images successives.

### 4.3.2 Données expérimentales

Nous avons ensuite testé notre algorithme sur des images expérimentales, provenant de la plaque Coriolis [75]. L'expérience est la suivante : une cuve tournante de 13 mètres de diamètre contenant de l'eau est mise en rotation, afin de simuler la force de Coriolis appliquée à un fluide. En conséquence, le comportement de l'eau à l'intérieur de la cuve est proche de ce qui se passe dans un océan comme l'Atlantique Nord dans les images que nous avons testées.

Un tourbillon est alors créé, dans lequel on injecte soit un colorant, soit des particules fines. Enfin, une caméra prend des clichés de l'expérience à différents intervalles de temps.

Plusieurs cas ont été considérés en prenant des images correspondant à un délai d'acquisition soit rapide soit lent, et en travaillant sur des images de colorant ou de particules. Dans tous les cas, le champ de vitesse reconstruit correspond parfaitement au déplacement réel du fluide, à savoir un mouvement de rotation du tourbillon, plongé dans un mouvement beaucoup plus lent de translation globale. L'approche multi-échelle a été comparée avec une approche classique, où la minimisation est directement réalisée sur la grille fine. Les résultats sont alors considérablement dégradés, ainsi que le temps de calcul.

Ces résultats ont été comparés avec ceux fournis par la méthode PIV (Particle Imaging Velocimetry), qui sert de référence dans le domaine à l'interface de la géophysique et de la mécanique des fluides. Ils sont qualitativement équivalents puisque les champs identifiés se ressemblent. Toutefois, nous apportons deux avantages majeurs grâce à notre algorithme : le temps de calcul, qui nous permet de traiter des séquences de plusieurs centaines d'images à très haute résolution sans difficulté en quelques heures là où plusiers jours sont nécessaires à la méthode PIV ; et la précision des résultats, puisque nous pouvons identifier jusqu'à un vecteur vitesse par pixel de l'image originale, là où la méthode PIV fournit de l'ordre d'un vecteur vitesse par zone de $10 \times 10$ pixels. Cela permet notamment d'identifier des phénomènes tourbillonnaires de beaucoup plus petite échelle.

## 4.4 Conclusions

Nous avons présenté un algorithme extrêmement rapide et efficace dans le but d'extraire des champs de vitesse à partir de séquences d'images. La rapidité et la qualité des résultats proviennent des différents choix méthodologiques : approche multi-échelle, pas de linéarisation de la fonction coût, régularisation appropriée, assemblage rapide de la jacobienne, ...

Les résultats numériques obtenus dans des conditions simulées et expérimentales démontrent l'efficacité de la méthode. D'autres tests ont été réalisés en traitant des films complets à haute résolution provenant également de la plaque Coriolis, et confirment la qualité de la méthode.

Au cours de cette étude, nous avons réalisé que les vitesses identifiées correspondent en fait à des champs lagrangiens, à cause du délai d'acquisition entre les images. Ces champs de vitesse peuvent ensuite être vus comme des observations du système, et utilisés dans un algorithme d'assimilation de données. Toutefois, si le temps entre deux images est trop important, alors la vitesse identifiée (qui correspond à une vitesse apparente) ne peut être assimilée à la vitesse réelle du fluide, et il faut alors utiliser une méthode d'assimilation de données lagrangiennes pour traiter ces observations.

# Chapitre 5

# Conclusions générales et perspectives

Nous avons présenté ici plusieurs algorithmes pour résoudre des problèmes en traitement d'images et en assimilation de données. Tous ces algorithmes ont les points communs d'être rapides, efficaces, robustes, et faciles à mettre en œuvre. Ce travail a été particulièrement motivé par les applications, et les contraintes liées aux applications. Ces contraintes sont par exemple d'être capable de traiter des images et films en temps réel, mais également d'assimiler de grandes quantités de données météorologiques ou océaniques en un temps fixé (généralement imposé par les contraintes opérationnelles de fournir un résultat à échéances régulières).

Il nous a paru important de définir des méthodes en rupture avec l'état de l'art, que ce soit en imagerie ou en assimilation de données. En traitement d'images, l'introduction du gradient topologique a apporté un éclairage nouveau sur les problèmes, grâce notamment au caractère global du gradient topologique (contrairement au gradient classique qui est beaucoup plus local). De même en assimilation de données, la communauté est scindée en deux écoles diamétralement opposées : les méthodes variationnelles telles que le 4D-VAR, nécessitant le développement humainement coûteux de l'adjoint, et les méthodes séquentielles basées sur des filtres de Kalman (étendus, évolutifs, stochastiques, ...) reposant sur une bonne connaissance des statistiques d'erreur. Nous avons fait le choix de définir une méthode qui se trouve à l'interface des deux, et qui combine les avantages des méthodes sans en avoir les inconvénients.

Les perspectives restent nombreuses dans ces diverses thématiques, à la fois parce que certains problèmes n'ont pas encore été étudiés, mais aussi grâce aux possibilités d'amélioration intrinsèque des méthodes développées. Des perspectives à court terme consistent donc à améliorer les algorithmes définis. Par exemple, la méthode du gradient topologique peut être améliorée en cherchant des contours de l'image qui ne soient pas tous aussi importants les uns que les autres, en utilisant plus que deux valeurs différentes pour la conductivité. En assimilation de données, la méthode du nudging direct et rétrograde peut également être améliorée, en augmentant ou diminuant automatiquement la valeur des gains au fil des itérations, pour essayer de garder un équilibre entre le terme des équations du modèle et le terme de rappel aux observations.

À plus long terme, parmi les problèmes restant à étudier en traitement d'images, nous pouvons citer par exemple la compression et le défloutage d'images, et en assimilation de données, un challenge intéressant consiste à étudier la possibilité d'assimiler des données réelles avec le nudging direct et rétrograde.

# Liste des publications

## 1. Publications en rapport avec la thèse

### Thèse

[1] D. Auroux : *Étude de quelques méthodes d'assimilation de données pour l'environnement.* Thèse de doctorat, Université de Nice Sophia-Antipolis, Décembre 2003.

### Publications dans des revues internationales à comité de lecture

[2] F. Veersé, D. Auroux et M. Fisher : Limited-memory BFGS diagonal preconditioners for a data assimilation problem in meteorology. *Optim. Engineer.*, 1.3:323–339, 2000.

[3] D. Auroux : Generalization of the dual variational data assimilation algorithm to a nonlinear layered quasi-geostrophic ocean model. *Inverse Problems*, 23: 2485–2503, 2007.

### Chapitre de livre à comité de lecture

[4] D. Auroux et J. Blum : *Data assimilation methods for an oceanographic problem*, volume XVI de *Lecture Notes - Mathematics in Industry*, pages 179–194. Springer-Verlag, 2004.

### Note

[5] D. Auroux et J. Blum : A dual data assimilation method for a layered quasi-geostrophic ocean model. *Rev. R. Acad. Cien. Serie A Mat.*, 96(3):316–320, 2002.

### Rapport de recherche

[6] F. Veersé et D. Auroux : Some numerical experiments on scaling and updating L-BFGS diagonal preconditioners. Rapport technique 3858, INRIA, 2000.

### Acte de congrès national avec comité de lecture

[7] D. Auroux : Assimilation variationnelle de données océanographiques - approches primale et duale. 9(2), pages 147–159. Annales Mathématiques Blaise Pascal, 2002.

## 2. Publications postérieures à la thèse

### Publications dans des revues internationales à comité de lecture

[8] D. Auroux : Several data assimilation methods for geophysical problems. *Ind. J. Pure Appl. Math.*, 37(1):41–58, 2006.

[9] D. Auroux et M. Masmoudi : A one-shot inpainting algorithm based on the topological asymptotic analysis. *Comp. Appl. Math.*, 25(2-3):1–17, 2006.

[10] D. Auroux, L. Jaafar Belaid et M. Masmoudi : A topological asymptotic analysis for the regularized grey-level image classification problem. *Math. Model. Numer. Anal.*, 41(3):607–625, 2007.

[11] D. Auroux et J. Blum : A nudging-based data assimilation method for oceanographic problems : the Back and Forth Nudging (BFN) algorithm. *Nonlin. Proc. Geophys.*, 15:305–319, 2008.

[12] D. Auroux : From restoration by topological gradient to medical image segmentation via an asymptotic expansion. *Math. Comput. Model.*, 2008. Sous presse.

[13] D. Auroux et M. Masmoudi : Image processing by topological asymptotic expansion. *J. Math. Imaging Vision*, 2008. Sous presse.

[14] D. Auroux : The back and forth nudging algorithm applied to a shallow water model, comparison and hybridization with the 4D-VAR. *Int. J. Numer. Methods Fluids*, 2008. Sous presse.

[15] M. Masmoudi, D. Auroux et Y. Parte : The state of the art in collaborative design. *Comput. Fluid Dyn. J.*, 2008. Accepté pour publication.

### Chapitre de livre à comité de lecture

[16] D. Auroux, M. Masmoudi et L. Jaafar Belaid : *Image restoration and classification by topological asymptotic expansion.* Variational Formulations in Mechanics : Theory and Applications, E. Taroco, E.A. de Souza Neto and A.A. Novotny (Eds). CIMNE, Barcelona, Spain, 2006.

[17] D. Auroux, J. Clément, J. Hermetz, M. Masmoudi et Y. Parte : *État de l'art et nouvelles tendances en conception collaborative.* Optimisation multidisciplinaire en mécanique. Hermes Science Publishing, London, 2008.

### Note aux comptes rendus

[18] D. Auroux et J. Blum : Back and forth nudging algorithm for data assimilation problems. *C. R. Acad. Sci. Paris, Ser. I*, 340:873–878, 2005.

### Actes de congrès internationaux à comité de lecture

[19] G. Caille, Y. Cailloce, C. Guiraud, D. Auroux, T. Touya et M. Masmoudi : Large multibeam array antennas with reduced number of active chains. *In Proc. EuCAP 2007 - Antennas and Propagation*, pages 1–9, 2007.

[20] T. Touya et D. Auroux : Control and topological optimization of a large multibeam array antenna. *In* A. Daryoush, éditeur : *Proc. ARP 2008 - Antennas, Radar, and Wave Propagation*. ACTA Press, 2008.

[21] D. Auroux, P. Bansart et J. Blum : An easy-to-implement and efficient data assimilation method for the identification of the initial condition : the Back and Forth Nudging (BFN) algorithm. J. Physics Conf. Series, 2008. Sous presse.

### Preprints

[22] D. Auroux, L. Jaafar Belaid et B. Rjaibi : Application of the topological gradient method to color image restoration. Soumis, 2008.

[23] D. Auroux et J. Fehrenbach : Identification of velocity fields for geophysical fluids from a sequence of images. Soumis, 2008.

[24] D. Auroux et M. Nodet : The back and forth nudging algorithm for data assimilation problems : theoretical results on transport equations. Soumis, 2008.

[25] D. Auroux et S. Bonnabel : Symmetry-preserving nudging : theory and application to a shallow water model. Soumis, 2008.

[26] Y. Ahipo, D. Auroux, L. Cohen et M. Masmoudi : A hybrid scheme for contour detection and completion based on topological gradient and fast marching algorithms - an application to image inpainting and segmentation. Prépublication, 2008.

# Bibliographie


[27] R. Adrian : Particle imaging techniques for experimental fluid mechanics. *Ann. Rev. Fluid Mech.*, 23:261–304, 1991.

[28] N. Aghannan et P. Rouchon : On invariant asymptotic observers. *In Proc. of the 41st IEEE Conf. on Decision and Control*, volume 2, pages 1479–1484, 2002.

[29] N. Aghannan et P. Rouchon : An intrinsic observer for a class of lagrangian systems. *IEEE Trans. Autom. Contr.*, 48(6):936–945, 2003.

[30] G. Alessandrini, E. Beretta et S. Vessela : Determining linear cracks by boundary measurements : Lipschitz stability. *SIAM J. Math. Anal.*, 27(2):361–375, 1996.

[31] G. Alessandrini et A. Diaz Valenzuela : Unique determination of multiple cracks by two measurements. *SIAM J. Control Optim.*, 34(3):913–921, 1996.

[32] G. Allaire : Shape optimization by the homogenization method. *Applied Mathematical Sciences*, 146, Springer-Verlag, 2002.

[33] G. Allaire, F. Jouve et A. M. Taoder : A level set method for shape optimization. *C. R. Acad. Sci. Paris Ser. I*, 334:1125–1130, 2002.

[34] G. Allaire et R. Kohn : Optimal design for minimum weight and compliance in plane stress using extremal microstructures. *Eur. J. Mech. A Solids*, 12:839–878, 1993.

[35] L. Alvarez, C. Castaño, M. Garcia, K. Krissian, L. Mazorra, A. Salgado et J. Sanchez : *A variational approach for 3D motion estimation of incompressible PIV flows*, pages 837–847. Springer, Berlin, 2007.

[36] H. Ammari, M. S. Vogelius et D. Volkov : Asymptotic formulas for perturbations in the electromagnetic fields due to the presence of inhomogeneities of small diameter ii - the full maxwell equations. *J. Math. Pures Appl.*, 80(8):769–814, 2001.

[37] S. Amstutz, I. Horchani et M. Masmoudi : Crack detection by the topological gradient method. *Control and Cybernetics*, 34(1):119–138, 2005.

[38] S. Amstutz, M. Masmoudi et B. Samet : The topological asymptotic for the helmoltz equation. *SIAM J. Control Optim.*, 42(5):1523–1544, 2003.

[39] P. Ananden : A computational framework and an algorithm for the measurement of visual motion. *Int. J. Comput. Vis.*, 2:283–310, 1989.

[40] S. Andrieux et A. Ben Abda : Identification of planar cracks by complete overdetermined data : inversion formulae. *Inverse Problems*, 12:553–563, 1996.

[41] M. Arnold et B. N. Datta : Single-input eigenvalue assignment algorithms : A close look. *SIAM J. Matrix Anal. Appl.*, 19(2):444–467, 1998.

[42] G. Aubert et J.-F. Aujol : Optimal partitions, regularized solutions, and application to image classification. *Applicable Analysis*, 84(1):15–35, 2005.

[43] G. Aubert et P. Kornprobst : *Mathematical Problems in Image Processing*, volume 147 de *Applied Mathematical Sciences*. Springer-Verlag, 2001.

[44] G. Aubert et L. Vese : A variational method in image recovery. *SIAM J. Numer. Anal.*, 34(5):1948–1979, 1997.

[45] J.-F. Aujol, G. Aubert et L. Blanc-Féraud : Wavelet-based level set evolution for classification of textured images. *IEEE Trans. Image Proc.*, 12(12):1634–1641, 2003.

[46] J.-W. Bao et R. M. Errico : An adjoint examination of a nudging method for data assimilation. *Month. Weather Rev.*, 125:1355–1373, 1997.

[47] S. Beauchemin et J. Barron : The computation of optical flow. *ACM Computing Surveys*, 27(3):433–467, 1995.

[48] Z. Belhachmi et D. Bucur : Stability and uniqueness for the crack identification problem. *SIAM J. Control Optim.*, 46(1):253–263, 2007.

[49] A. Ben Abda, H. Ben Ameur et M. Jaoua : Identification of 2d cracks by elastic boundary measurements. *Inverse Problems*, 15:67–77, 1999.

[50] A. Ben Abda, M. Kallel, J. Leblond et J.-P. Marmorat : Line-segment cracks recovery from incomplete boundary data. *Inverse Problems*, 18:1057–1077, 2002.

[51] M. Bendsoe : Optimal topology design of continuum structure : an introduction. Rapport technique, Department of Mathematics, Technical University of Denmark, Lyngby, Denmark, 1996.

[52] M. Bendsoe et P. Kikuchi : Generating optimal topologies in structural design using an homogenisation method. *Comput. Methods Appl. Mech. Engin.*, 71:197–224, 1988.

[53] A. F. Bennett : *Inverse Modeling of the Ocean and Atmosphere*. Cambridge University Press, Cambridge, 2002.

[54] M. Berthod, Z. Kato, S. Yu et J. Zerubia : Bayesian image classification using markov random fields. *Image Vision Comput.*, 14(4):285–293, 1996.

[55] A. Blake et M. Isard : *Active Contours*. Springer-Verlag, 1998.

[56] E. Blayo, S. Durbiano, P. A. Vidard et F.-X. Le Dimet : *Reduced order strategies for variational data assimilation in oceanic models*. Springer-Verlag, 2003.

[57] E. Blayo, J. Verron et J.-M. Molines : Assimilation of TOPEX/POSEIDON altimeter data into a circulation model of the North Atlantic. *J. Geophys. Res.*, 99(C12):24691–24705, 1994.

[58] S. Bonnabel, Ph. Martin et P. Rouchon : Non-linear symmetry-preserving observers on Lie groups. *IEEE Trans. Autom. Contr.*, 2008. accepted for publication.

[59] S. Bonnabel, Ph. Martin et P. Rouchon : Symmetry-preserving observers. *IEEE Trans. Automat. Contr.*, 2008. In press.

[60] F. Bonnans et P. Rouchon : *Commande et optimisation de systèmes dynamiques*. Les Éditions de l'École Polytechnique, Palaiseau, 2005.

[61] C. A. Bouman et M. Shapiro : A multiscale random field model for bayesian image segmentation. *IEEE Trans. Image Proc.*, 3:162–177, 1994.

[62] F. Bouttier et G. Kelly : Observing system experiments with the ecmwf data assimilation system. *Quart. J. Roy. Meteor. Soc.*, 127:1469–1488, 2001.

[63] M. Bruhl, M. Hanke et M. Pidcock : Crack detection using electrostatic measurements. *Math. Model. Numer. Anal.*, 35:595–605, 2001.

[64] K. Bryan et M. S. Vogelius : A review of selected works on crack identification. *In Proc. IMA workshop on Geometric Methods in Inverse Problems and PDE Control*, August 2001.

[65] A.-P. Calderón : On an inverse boundary value problem. *In Seminar on Numerical Analysis and Its Applications to Continuum Physics*, pages 65–73, Rio de Janeiro, Brasil, 1980. Soc. Brasil. Mat.

[66] F. Catté, T. Coll, P.-L. Lions et J.-M. Morel : Image selective smoothing and edge detection by non linear diffusion. *SIAM J. Numer. Anal.*, 29:182–193, 1992.

[67] T. Chan et J. Shen : Mathematical models for local deterministic inpaintings. Rapport technique 00-11, UCLA CAM, March 2000.

[68] T. Chan et J. Shen : Non-texture inpainting by curvature-driven diffusions (ccd). Rapport technique 00-35, UCLA CAM, September 2000.

[69] T. Chan et J. Shen : *Image processing and analysis : variational, PDE, wavelet, and stochastic methods*. SIAM, 2005.

[70] P. Charbonnier : *Reconstruction d'images : régularisation avec prise en compte des discontinuités*. Thèse de doctorat, Université de Nice Sophia-Antipolis, 1994.

[71] L. D. Cohen : Multiple contour findind and perceptual grouping using minimal paths. Rapport technique, Les Cahiers du CEREMADE, Université Paris Dauphine, 2001.

[72] L. D. Cohen : *Minimal paths and fast marching methods for image analysis*. Mathematical models in computer vision : the handbook, N. Paragios, Y. Chen and O. Faugeras (Eds). Springer, 2005.

[73] L. D. Cohen et R. Kimmel : Global minimum for active contour models : a minimal path approach. *Int. J. Computer Vision*, 24(1):57–78, 1997.

[74] J. D. Cole : On a quasilinear parabolic equation occurring in aerodynamics. *Quart. Appl. Math.*, 9(3):225–236, 1951.

[75] Coriolis rotating platform, website `http://www.coriolis-legi.org/`.

[76] T. Corpetti, E. Mémin et P. Pérez : Dense estimation of fluid flows. *IEEE Trans. Pattern Anal. Mach. Intel.*, 24:365–380, 2002.

[77] R. Courant et D. Hilbert : *Methods of Mathematical Physics, Volume II.* Wiley-Interscience, 1962.

[78] P. Courtier : Dual formulation of four-dimensional variational assimilation. *Q. J. R. Meteorol. Soc.*, 123:2449–2461, 1997.

[79] A. Cuzol, P. Hellier et E. Mémin : A low dimensional fluid motion estimator. *Int. J. Comput. Vis.*, 75:329–349, 2007.

[80] B. N. Datta : An algorithm to assign eigenvalues in a hessenberg matrix : Single-input case. *IEEE Trans. Automatic Control*, 32(5):414–417, 1987.

[81] R. Dautray et J.-L. Lions : *Analyse mathématique et calcul numérique pour les sciences et les techniques.* Collection CEA. Masson, Paris, 1987.

[82] T. Deschamps et L. D. Cohen : Minimal path in 3D images and application to virdual endoscopy. Rapport technique, Les Cahiers du CEREMADE, Université Paris Dauphine, 2000.

[83] S. Di Zenzo : A note on the gradient of a multi-image. *Comput. Vision Graph. Image Proc.*, 33:116–125, 1986.

[84] J. Dicker : *Fast marching methods and level set methods : an implementation.* Thèse de doctorat, University of British Columbia, 2006.

[85] F. Dubois, N. Petit et P. Rouchon : Motion planing and nonlinear simulations for a tank containing a fluid. *In European Control Conference, Karlsruhe*, 1999.

[86] M. Elad et M. Aharon : Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Proc.*, 15(12):3736–3745, 2006.

[87] M. Elad, J.-L. Starck, P. Querre et D. L. Donoho : Simultaneous cartoon and texture image inpainting using morphological component analysis (mca). *J. Appl. Comput. Harmonic Anal.*, 19:340–358, 2005.

[88] H.W. Engl et C.W. Groetsch : *Inverse and Ill Posed Problems.* Academic Press, New York, 1987.

[89] L. C. Evans : *Partial Differential Equations.* American Mathematical Society, Providence, 1998.

[90] G. Evensen et P. J. van Leeuwen : An ensemble Kalman smoother for nonlinear dynamics. *Mon. Wea. Rev.*, 128:1852–1867, 1999.

[91] J. Fehrenbach et M. Masmoudi : A fast algorithm for image registration. *C. R. Acad. Sci. Paris, Ser. I*, 346:593–598, 2008.

[92] M. Fisher et P. Courtier : Estimating the covariance matrix of analysis and forecast error in variational data assimilation. Rapport technique 220, ECMWF, 1995.

[93] J. M. Fitzpatrick : A method for calculating velocity in time dependent images based on the continuity equation. *In Proc. Conf. Comp. Vision Pattern Rec.*, pages 78–81, San Francisco, USA, 1985.

[94] J. M. Fitzpatrick : The existence of geometrical density-image transformations corresponding to object motion. *Comput. Vis. Grap. Imag. Proc.*, 44:155–174, 1988.

[95] J. B. Flor et I. Eames : Dynamics of monopolar vortices on the beta plane. *J. Fluid Mech.*, 456:353–376, 2002.

[96] A. Friedman et M. S. Vogelius : Determining cracks by boundary measurements. *Indiana Univ. Math. J.*, 38(3):527–556, 1989.

[97] A. Friedman et M. S. Vogelius : Identification of small inhomogeneities of extreme conductivity by boundary measurements : a theorem of continuous dependance. *Arch. Rational Mech. Anal.*, 105(4):299–326, 1989.

[98] S. Garreau, P. Guillaume et M. Masmoudi : The topological asymptotic for pde systems : The elasticity case. *SIAM J. Control Optim.*, 39(6):1756–1778, 2001.

[99] M. Ghil : Meteorological data assimilation for oceanographers. part i : description and theoretical framework. *Dyn. Atmos. Oceans*, 13:171–218, 1989.

[100] M. Ghil et P. Manalotte-Rizzoli : Data assimilation in meteorology and oceanography. *Adv. Geophys.*, 23:141–265, 1991.

[101] J.-Ch. Gilbert et C. Lemaréchal : Some numerical experiments with variable-storage quasi-Newton algorithms. *Math. Prog.*, 45:407–435, 1989.

[102] J. Giroire et J.-C. Nédélec : Numerical solution of an exterior neumann problem using a double layer potentiel. *Math. Comput.*, 32(144):973–990, 1978.

[103] GOES geostationnary satellite server `http://www.goes.noaa.gov/`.

[104] P. Guillaume et K. Sididris : The topological asymptotic expansion for the dirichlet problem. *SIAM J. Control Optim.*, 41(4):1042–1072, 2002.

[105] P. Guillaume et K. Sididris : The topological sensitivity and shape optimization for the stokes equations. *SIAM J. Control Optim.*, 43(1):1–31, 2004.

[106] J. Hoke et R. A. Anthes : The initialization of numerical models by a dynamic initialization technique. *Month. Weather Rev.*, 104:1551–1556, 1976.

[107] W. R. Holland : The role of mesoscale eddies in the general circulation of the ocean. *J. Phys. Ocean.*, 8(3):363–392, 1978.

[108] E. Hopf : The partial differential equation $u_t + uu_x = \mu u_{xx}$. *Comm. Pure Appl. Math.*, 3:201–230, 1950.

[109] B. Horn et B. Schunk : Determining optical flow. *Artifical Intelligence*, 17:185–203, 1981.

[110] E. Huot, T. Isambert, I. Herlin, J.-P. Berroir et G. Korotaev : Data assimilation of satellite images within an oceanographic circulation model. *In Proc. Int. Conf. Acoustics Speech Sign. Proc.*, Toulouse, France, 2006.

[111] T. Isambert, I. Herlin et J.-P. Berroir : Fast and stable vector spline method for fluid flow estimation. *In Proc. Int. Conf. Image Proc.*, pages 505–508, San Antonio, USA, 2007.

[112] L. Jaafar-Belaid, M. Jaoua, M. Masmoudi et L. Siala : Image restoration and edge detection by topological asymptotic expansion. *C. R. Acad. Sci., Ser. I*, 342(5):313–318, 2006.

[113] S. Jiang et M. Ghil : Tracking nonlinear solutions with simulated altimetric data in a shallow-water model. *J. Phys. Oceanogr.*, 27(1):72–95, 1997.

[114] R. E. Kalman : A new approach to linear filtering and prediction problems. *Trans. ASME - J. Basic Engin.*, 82:35–45, 1960.

[115] E. Kalnay : *Atmospheric modeling, data assimilation and predictability.* Cambridge University Press, 2003.

[116] E. Kalnay, S. Ki Park, Z.-X. Pu et J. Gao : Application of the quasi-inverse method to data assimilation. *Month. Weather Rev.*, 128:864–875, 2000.

[117] A. Khludnev et V. Kovtunenko : *Analysis of cracks in solids.* WIT Press, Southampton-Boston, 2000.

[118] R. Kohn et M. Vogelius : Relaxation of a variational method for impedance computed tomography. *Comm. Pure Appl. Math.*, 40(6):745–777, 1987.

[119] V. Komornik et P. Loreti : *Fourier Series in Control Theory.* Springer Monographs in Mathematics. Springer-Verlag, New York, 2005.

[120] S. Kubo et K. Ohji : *Inverse problems and the electric potential computed tomography method as one of their application.* Mechanical Modeling of New Electromagnetic Materials. Elsevier Science Publisher, 1990.

[121] I. Larrabide, R. A. Feijóo, A. A. Novotny, E. Taroco et M. Masmoudi : An image segmentation method base on a discrete version of the topological derivative. *In Proc. 6th World Congress on Structural and Multidisciplinary Optimization*, Brazil, June 2005.

[122] R. Larsen, K. Conradsen et B. K. Ersboll : Estimation of dense image flow fields in fluids. *IEEE Trans. Geoscience Remote Sensing*, 36(1):256–264, 1998.

[123] F.-X. Le Dimet et O. Talagrand : Variational algorithms for analysis and assimilation of meteorological observations : theoretical aspects. *Tellus*, 38A:97–110, 1986.

[124] Y. Li et F. Santosa : A computational algorithm for minimizing total variation in image restoration. *IEEE Trans. Image Proc.*, 5:987–995, 1996.

[125] R. L. Lillie : *Whole Earth Geophysics : an introduction textbook for gelologists and geophysicists.* Prentice Hall, 1999.

[126] A. C. Lorenc, R. S. Bell et B. Macpherson : The meteorological office analysis correction data assimilation scheme. *Quart. J. Royal Meteor. Soc.*, 117:59–89, 1991.

[127] E. N. Lorenz : Deterministic non periodic flow. *J. Atmos. Sci*, 20:130–141, 1963.

[128] B. Lucas et T. Kanade : An iterative image registration technique with an application to stereo vision. *In Proc Seventh International Joint Conference on Artificial Intelligence*, pages 674–679, Vancouver, Canada, 1981.

[129] D. Luenberger : Observers for multivariable systems. *IEEE Trans. Autom. Contr.*, 11:190–197, 1966.

[130] B. Luong, J. Blum et J. Verron : A variational method for the resolution of a data assimilation problem in oceanography. *Inverse Problems*, 14:979–997, 1998.

[131] W. H. Lyne, R. Swinbank et N. T. Birch : A data assimilation experiment and the global circulation during the fgge special observing periods. *Quart. J. Roy. Meteor. Soc.*, 108:575–594, 1982.

[132] J. Ma, A. Antoniadis et F.-X. Le Dimet : Curvlets-based snake for multiscale detection and tracking of geophysical fluids. *IEEE Trans. Geoscience Remote Sensing*, 45(1):3626–3638, 2006.

[133] M. Masmoudi : *The topological asymptotic*, volume 16 de *Computational Methods for Control Applications, R. Glowinski, H. Karawada, and J. Périaux (Eds.)*, pages 53–72. GAKUTO Internat. Ser. Math. Sci. Appl., Tokyo, Japan, 2001.

[134] E. Mémin et P. Perez : Optical flow estimation and object-based segmentation with robust techniques. *IEEE Trans. Image Proc.*, 7(5):703–719, 1998.

[135] Y. Michel et F. Bouttier : Automatic tracking of dry intrusions on satellite water vapour imagery and model output. *Quart. J. Roy. Meteor. Soc.*, 132:2257–2276, 2006.

[136] D. Mumford et J. Shah : Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.*, 42(5):577–685, 1989.

[137] N. Nishimura et S. Kobayashi : A boundary integral equation method for an inverse problem related to crack detection. *Int. J. Num. Methods Engrg.*, 32:1371–1387, 1991.

[138] J. Nocedal : Updating quasi-Newton matrices with limited storage. *Math. Comput.*, 35:773–782, 1980.

[139] P. J. Olver : *Equivalence, Invariants, and Symmetry.* Cambridge University Press, 1995.

[140] N. Paragios et R. Deriche : Geodesic active regions and level set methods for supervised texture segmentation. *Int. J. Comput. Vision*, 46(3):223–247, 2002.

[141] T. Pavlidis et Y.-T. Liow : Integrating region growing and edge detection. *IEEE Trans. Pattern Anal. Mach. Intel.*, 12(3):225–233, 1990.

[142] P. Perona et J. Malik : Scale space and edge detection detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intel.*, 12:629–639, 1990.

[143] C. Pires, R. Vautard et O. Talagrand : On extending the limits of variational assimilation in nonlinear chaotic systems. *Tellus*, 48A:96–121, 1996.

[144] F. Rabier, H. Järvinen, E. Klinker, J.-F. Mahfouf et A. Simmons : The ECMWF operational implementation of four-dimensional assimilation. i : Experimental results with simplified physics. *Q. J. R. Meteorol. Soc.*, 126:1143–1170, 2000.

[145] N. Rawlinson et M. Sambridge : The fast marching method : an effective tool for tomographic imaging and tracking multiple phases in complex layered media. *Explor. Geophys.*, 36:341–350, 2005.

[146] P. Ruhnau, T. Kohlberger, C. Schnörr et H. Nobach : Variational optical flow estimation for particle image velocimetry. *Experiments in Fluids*, 38(1):21–32, 2005.

[147] D. L. Russell : Controllability and stabilizability theory for linear partial differential equations : recent progress and open questions. *SIAM Rev.*, 20(4): 639–739, 1978.

[148] B. Samet, S. Amstutz et M. Masmoudi : The topological asymptotic for the helmholtz equation. *SIAM J. Control Optim.*, 42(5):1523–1544, 2003.

[149] C. Samson, L. Blanc-Féraud, G. Aubert et J. Zerubia : A level set model for image classification. *Int. J. Comput. Vision*, 40(3):187–197, 2000.

[150] C. Samson, L. Blanc-Féraud, G. Aubert et J. Zerubia : A variational model for image classification and restoration. *IEEE Trans. Pattern Anal. Mach. Intel.*, 22(5):460–472, 2000.

[151] F. Santosa : A level-set approach for inverse problems involving obstacles. *ESAIM Control Optim. Calc. Var.*, 1:17–33, 1996.

[152] F. Santosa et M. Vogelius : A computational algorithm to determine cracks from electrostatic boundary measurements. *Int. J. Eng. Sci.*, 29:917–937, 1991.

[153] G. Sapiro : *Differential Equations and Image Analysis*. Cambridge University Press, Cambridge, 2001.

[154] A. Schumacher : *Topologieoptimisierung von Bauteilstrukturen unter Verwendung von Lopchpositionierungkrieterien*. Thèse de doctorat, Universitat-Gesamthochschule-Siegen, Germany, 1995.

[155] L. Schwartz : Opérateurs invariants par rotations. Fonctions métaharmoniques. *In Séminaire Schwartz*, volume 2(9), pages 1–5, 1954.

[156] J. A. Sethian : *Level Set Methods and Fast Marching Methods : Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Sciences.* Cambridge Monograph on Applied and Computational Mathematics. Cambridge University Press, Cambridge, 1999.

[157] J. Sokolowski et A. Zochowski : On the topological derivative in shape optimization. *SIAM J. Control Optim.*, 37:1241–1272, 1999.

[158] J. Sokolowski et A. Zochowski : Topological derivatives of shape functionals for elasticity systems. *Int. Ser. Numer. Math*, 139:231–244, 2002.

[159] D. R. Stauffer et J. W. Bao : Optimal determination of nudging coefficients using the adjoint equations. *Tellus*, 45A:358–369, 1993.

[160] D. R. Stauffer et N. L. Seaman : Use of four dimensional data assimilation in a limited area mesoscale model - part 1 : Experiments with synoptic-scale data. *Month. Weather Rev.*, 118:1250–1277, 1990.

[161] O. Talagrand : On the mathematics of data assimilation. *Tellus*, 33A:321–339, 1981.

[162] O. Talagrand : A study of the dynamics of four-dimensional data assimilation. *Tellus*, 33A:43–60, 1981.

[163] O. Talagrand : Assimilation of observations, an introduction. *J. Met. Soc. Japan*, 75(1B):191–209, 1997.

[164] A. Telea et J. van Wijk : An augmented fast marching method for computing skeletons and centerlines. *In Proc. Eurographics IEEE-TCVG Symposium on Visualization*, Vienna, 2002. Springer.

[165] D. Terzopoulos : Regularization of inverse visual problems involving discontinuities. *IEEE Trans. Pattern Anal. Mach. Intel.*, 8(4):413–424, 1986.

[166] A. N. Tikhonov et V. Y. Arsenin : *Solutions of Ill Posed Problems.* Winston and sons, Washington, 1997.

[167] E. Trélat : *Contrôle optimal : théorie et applications.* Vuibert, Paris, 2005.

[168] J. Verron : Altimeter data assimilation into an ocean circulation model : sensitivity to orbital parameters. *J. Geophys. Res.*, 95(C7):443–459, 1990.

[169] J. Verron, L. Gourdeau, D. T. Pham, R. Murtugudde et A. J. Busalacchi : An extended Kalman filter to assimilate satellite altimeter data into a nonlinear numerical model of the tropical pacific ocean : Method and validation. *J. Geophys. Res.*, 104:5441–5458, 1999.

[170] J. Verron et W. R. Holland : Impact de données d'altimétrie satellitaire sur les simulations numériques des circulations générales océaniques aux latitudes moyennes. *Annales Geophysicae*, 7(1):31–46, 1989.

[171] J. Verron, J.-M. Molines et E. Blayo : Assimilation of geosat data into a quasigeostrophic model of the north atlantic between $20^o$n and $50^o$n : preliminary results. *Oceanol. Acta*, 15(5):575–583, 1992.

[172] P.-A. Vidard, F.-X. Le Dimet et A. Piacentini : Determination of optimal nudging coefficients. *Tellus*, 55A:1–15, 2003.

[173] M. Y. Wang, X. M. Wang et D. M. Guo : A level set method for structural topology optimization. *Comput. Meth. Appl. Mech. Engin.*, 192:227–246, 2003.

[174] J. Weickert : Theoretical foundations of anisotropic diffusion in image processing. *In Theoretical Foundations of Computer Vision*, volume 11, pages 221–236, Dagstuhl, Germany, 1994.

[175] J. Weickert : *Anisotropic diffusion in image processing*. Thèse de doctorat, University of Kaiserslautern, Germany, 1996.

[176] J. Weickert : Efficient image segmentation using partial differential equations and morphology. *Pattern Recognition*, 34(9):1813–1824, 2001.

[177] P. Wen, X. Wu et C. Wu : An interactive image inpainting method based on RBF networks. *In Third International Symposium on Neural Networks*, pages 629–637, 2006.

[178] T. Zhou, F. Tang, J. Wang, Z. Wang et Q. Peng : Digital image inpainting with radial basis functions. *J. Image Graphics (In Chinese)*, 9(10):1190–1196, 2004.

[179] F. Zhu et J. Tian : Modified fast marching methods and level set method for medical image segmentation. *J. X-ray Sci. Tech*, 11(4):193–204, 2003.

[180] X. Zou, I. M. Navon et F.-X. Le Dimet : An optimal nudging data assimilation scheme using parameter estimation. *Q. J. R. Meteorol. Soc.*, 118:1163–1186, 1992.

**Résumé :**

Dans une première partie, nous avons étudié des problèmes de traitement d'images. Nous avons utilisé l'analyse asymptotique topologique pour la détection des contours d'une image. Cela permet de considérer alors plusieurs applications : restauration/débruitage, classification. L'inpainting est traité d'une façon un peu différente, et la double donnée Dirichlet et Neumann sur le bord du domaine caché permet de reconstruire les contours dans la partie cachée de l'image. Enfin la segmentation peut être traitée comme limite de la classification, en s'appuyant sur des résultats d'analycité de la solution quand on fait tendre un paramètre vers 0. La rapidité de cette méthode permet de traiter ces différents cas en temps réel, y compris pour des films.

Dans une seconde partie, nous avons abordé l'assimilation de données, le but étant d'identifier la condition initiale d'un système à partir d'observations partielles. Nous avons défini un nouvel algorithme, basé sur le "nudging" (méthode de relaxation consistant à rajouter un terme de rappel vers les observations directement dans l'équation afin de tirer la solution vers les observations). En considérant itérativement et alternativement des résolutions du système direct et rétrograde en temps, avec à chaque fois un terme de rappel vers les observations, on peut améliorer l'estimation de la condition initiale. Là encore, la méthode est performante et extrêmement rapide, comme de nombreux tests numériques l'ont démontré. En parallèle, plusieurs résultats théoriques de convergence ont été obtenus dans des cas simplifiés.

Enfin, une étude a été réalisée à l'interface de ces deux thématiques : l'extraction de données, et plus précisement de champs de vitesses, à partir de séquences d'images météorologiques ou océanographiques. L'idée consiste à chercher un champ de vitesse (ou déplacement) qui transporte une image sur la suivante. L'approche considérée est variationnelle, et basée sur la minimisation d'une fonctionnelle non linéaire dépendant du champ de vitesse. Une approche multi-grille permet d'obtenir très rapidement des champs de vitesse. Ces vitesses peuvent alors être assimilées directement dans un système d'assimilation.

**Summary :**

In a first chapter, we consider image processing problems. We applied the topological asymptotic analysis to the edge detection problem. Once the edges are identified, one can easily consider the restoration/enhancement and classification problems. The inpainting problem has also been considered, but from a slightly different point of view: given the Dirichlet and Neumann conditions on the boundary of the unknown part of the image, the topological gradient allows one to retrieve the missing edges of the hidden zone, and then to reconstruct an unblurred image. Finally, the segmentation problem has been considered with the same mathematical tools, using the analycity of the enhanced solution with respect to a small parameter. All these algorithms are extremely efficient and fast, and allows us to process images and even movies in real time.

The second chapter is devoted to data assimilation. We developed a new algorithm : the Back and Forth Nudging (BFN). The standard nudging technique consists in adding to the equations of the model a relaxation term that is supposed to force the observations to the model. The BFN algorithm consists in repeatedly performing forward and backward integrations of the model with relaxation (or nudging) terms, using opposite signs in the direct and inverse integrations, so as to make the backward evolution numerically stable. Extensive numerical experiments have been performed on several simplified geophysical models, showing the efficiency of this easy-to-implement and fast approach. Moreover, several theoretical results of convergence have been obtained in simple situations.

Finally, we also worked at the interface of these two topics and considered image data assimilation. The idea is to extract velocity fields from a sequence of oceanographic or meteorological images. A variational approach has been proposed, in which the minimization of a nonlinear cost function provides a displacement (or velocity) field between two images. A multi-grid approach and an appropriate minimization process, allow us to extract the information very quickly. These "pseudo"-observations can then be directly assimilated as the velocity is usually a model variable.