UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Mathématiques appliquées**

Arrêté ministériel : 7 août 2006

Présentée par

Morgane HENRY

Thèse dirigée par **Emmanuel MAITRE**
et codirigée par **Valérie PERRIER**

préparée au sein **du Laboratoire Jean Kuntzmann**
et de **l'Ecole Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique**

Transport optimal et ondelettes : nouveaux algorithmes et application à l'image

Thèse soutenue publiquement le ,
devant le jury composé de :

M. Gabriel PEYRÉ
Directeur de recherche CNRS, Université Paris-Dauphine, Rapporteur
M. Jacques LIANDRAT
Professeur, École Centrale Marseille, Rapporteur
Mme Carole LE GUYADER
Professeur, INSA de Rouen, Examinateur
M. Nicolas PAPADAKIS
Chargé de recherche CNRS, Université de Bordeaux, Examinateur
M. Emmanuel MAITRE
Professeur, Grenoble INP - Ensimag, Directeur de thèse
Mme Valérie PERRIER
Professeur, Grenoble INP - Ensimag, Co-Directeur de thèse

Résumé

Le transport optimal trouve un nombre grandissant d'applications, dont celle qui nous intéresse dans ce travail, l'interpolation d'images. Malgré cet essor, la résolution numérique de ce transport soulève des difficultés et le développement d'algorithmes efficaces reste un problème d'actualité, en particulier pour des images de grande taille, comme on en trouve dans certains domaines (météorologie,...). Nous nous intéressons dans ce travail à la formulation de Benamou et Brenier, qui ont placé le problème dans un contexte de mécanique des milieux continus en ajoutant une dimension temporelle. Leur formulation consiste en la minimisation d'une fonctionnelle sur un espace des contraintes contenant une condition de divergence nulle, et les algorithmes existants utilisent une projection sur cet espace. A l'opposé, dans cette thèse, nous définissons et mettons en oeuvre des algorithmes travaillant directement dans cet espace. En effet, nous montrons que la fonctionnelle a de meilleures propriétés de convexité sur celui-ci. Pour travailler dans cet espace, nous considérons trois représentations des champs de vecteurs à divergence nulle. La première est une base d'ondelettes à divergence nulle. Cette formulation a été implémentée numériquement dans le cas des ondelettes périodiques à l'aide d'une descente de gradient, menant à un algorithme de convergence lente mais validant la faisabilité de la méthode. La deuxième approche consiste à représenter les vecteurs à divergence nulle par leur fonction de courant munie d'un relèvement des conditions au bord et la troisième à utiliser la décomposition de Helmholtz-Hodge. Nous montrons de plus que dans le cas unidimensionnel en espace, en utilisant l'une ou l'autre de ces deux dernières représentations, nous nous ramenons à la résolution d'une équation de type courbure minimale sur chaque ligne de niveau du potentiel, munie des conditions de Dirichlet appropriées. La minimisation de la fonctionnelle est alors assurée par un algorithme primal-dual pour problèmes convexes de Chambolle-Pock, qui peut aisément être adapté à nos différentes formulations et est facilement parallèlisable, menant à une implémentation performante et simple. En outre, nous démontrons les gains significatifs de nos algorithmes par rapport à l'état de l'art et leur application sur des images de taille réelle.

Abstract

Optimal transport has an increasing number of applications, including image interpolation, which we study in this work. Yet, numerical resolution is still challenging, especially for real size images found in applications. We are interested in the Benamou and Brenier formulation, which rephrases the problem in the context of fluid mechanics by adding a time dimension. It is based on the minimization of a functional on a constraint space, containing a divergence free constraint and the existing algorithms require a projection onto the divergence-free constraint at each iteration. In this thesis, we propose to work directly in the space of constraints for the functional to minimize. Indeed, we prove that the functional we consider has better convexity properties on the set of constraints. To work in this space, we use three different divergence-free vector decompositions. The first in which we got interested is a divergence-free wavelet base. This formulation has been implemented numerically using periodic wavelets and a gradient descent, which lead to an algorithm with a slow convergence but validating the practicability of the method. First, we represented the divergence-free vector fields by their stream function, then we studied the Helmholtz-Hodge decompositions. We prove that both these representations lead to a new formulation of the problem, which in 1D + time, is equivalent to the resolution of a minimal surface equation on every level set of the potential, equipped with appropriate Dirichlet boundary conditions. We use a primal dual algorithm for convex problems developed by Chambolle and Pock, which can be easily adapted to our formulations and can be easily sped up on parallel architectures. Therefore our method will also provide a fast algorithm, simple to implement. Moreover, we show numerical experiments which demonstrate that our algorithms are faster than state of the art methods and efficient with real-sized images.

Table des matières

1 Introduction **7**

2 Propriétés du problème de Monge-Kantorovich **11**
2.1 Introduction 11
2.1.1 Problème de Monge et problème de Kantorovich 12
2.1.2 Existence et gradient de fonctions convexes 15
2.2 Formulation mécanique des fluides 15
2.2.1 Description de la distance L^2 de Wasserstein 15
2.2.2 Algorithme Benamou-Brenier 19
2.3 Étude de la convexité dans l'espace des contraintes 19
2.4 Comportements connus 22
2.4.1 Déplacement en lignes droites 22
2.4.2 Déplacements par translation et homothétie 23
2.4.3 Cas unidimensionnel 23

3 Reformulation du problème à l'aide de décompositions de vecteurs à divergence nulle **25**
3.1 Bases d'ondelettes à divergence nulle 25
3.1.1 Généralités sur les ondelettes 1D 26
3.1.2 Base d'ondelettes à divergence nulle de $L^2(\mathbb{R}^2)^2$ 28
3.1.3 Base d'ondelettes à divergence nulle de $L^2((0,1)^2)^2$ 31
3.1.4 Formulation du problème de transport optimal sur bases d'ondelettes à divergence nulle 36
3.2 Fonction de courant 38
3.2.1 Transport optimal 1D 38
3.2.2 Transport optimal 2D 43
3.3 Décomposition de Helmholtz-Hodge 44

4 Algorithmes de résolution dans l'espace des contraintes **47**
4.1 Descente de gradient dans l'espace des coefficients d'ondelettes à divergence nulle 47
4.1.1 Algorithme dans l'espace des coefficients à divergence nulle 47
4.1.2 Calcul du gradient à l'aide des ondelettes duales 49

4.1.3 Calcul du gradient à l'aide des matrices de Gram 51
4.2 Algorithme primal-dual 53
4.2.1 Introduction 53
4.2.2 Application au cas $(\rho, m) = \nabla \times \phi$ 56
4.2.3 Application au cas $(\rho, m) = \nabla \times \phi + \nabla h$ 59
4.2.4 Autres applications 63

5 Applications Numériques **65**
5.1 Description du cadre discret 65
5.1.1 Discrétisation pour des images 1D 65
5.1.2 Discrétisation pour des images 2D 68
5.2 Comparaisons et tests numériques 70
5.2.1 Algorithmes pour signaux de dimension 1. 70
5.2.2 Applications aux images de dimension 2. 95
5.3 Implémentation en C++ 114
5.4 Applications aux images réelles 115
5.4.1 Comportements observés lors du transport de deux images 117

6 Conclusion **119**

A Calcul de quelques opérateurs proximaux **121**
A.1 Pour $\mathcal{J}_{ms} = ||.||_1$ 121
A.2 Opérateur proximal de $\mathcal{J}$ 122

B Opérateurs discrets de dérivée partielle adjoints **125**

Chapitre 1

Introduction

Le transport optimal est un domaine en pleine expansion, ayant de nombreuses applications dans des domaines aussi variés que l'économie [18], l'apprentissage machine (*machine learning*) [77] ou les équations aux dérivées partielles [14, 48], mais c'est son application au traitement d'image qui nous intéressera par la suite. Cette dernière application est particulièrement pertinente car le transport optimal définit une métrique entre des densités [83]. Une des applications les plus courantes est donc de déterminer une distance entre des images prises à différents moments ou de points de vue différents pour les comparer ou obtenir des informations intéressantes, comme par exemple pour la classification d'images [78].

Le transport optimal peut être formulé de façon statique ou dynamique, la formulation statique étant la recherche de la distance minimale entre deux images, alors que le problème dynamique est la recherche du chemin optimal entre ces deux images. De plus, la nature géométrique du transport optimal le rend intéressant pour l'interpolation de formes, par exemple en assimilation de données pour la comparaison d'images météorologiques [24, 33], ou en imagerie médicale, même si les résultats sont plutôt simplistes. C'est pourquoi des travaux récents développent des formulations nouvelles du transport optimal pour rendre l'interpolation d'images rencontrées dans les applications plus en adéquation avec la physique sous-jacente. Ces différents travaux comprennent le traitement des plans d'activation de neurones [40], l'analyse d'IRM de cerveaux sains et sclérosés [80], l'analyse de plans de l'hippocampe [10], ou encore l'étude de la croissance d'un cerveau [19]. Une des pistes étudiées pour rendre le transport optimal plus pertinent est la modification de l'énergie considérée avec l'ajout de termes pour prendre en compte les propriétés physiques des images que l'on souhaite interpoler [11, 47]. Une autre piste est l'interpolation entre la distance issue du transport optimal et une autre distance [69], ce qui peut permettre par exemple de généraliser le transport pour des mesures de masses différentes et de se comporter comme une version localisée du transport [19, 57]. Cette dernière modification est intéressante car en imagerie le fait que le transport optimal nécessite des images de même masse est problématique, les phénomènes à observer et analyser comportant souvent la création ou la destruction de masse. Le transport opti-

mal entre mesures de masses différentes a été étudié dès 1942 dans [54], où une partie de la masse peut être envoyée sur un point à l'infini, et est toujours utilisée dans [40] pour des problèmes en neuroimages. Dans [34] c'est la frontière qui est un réservoir infini de masse et dans [59] on pénalise à l'aide d'une fonction dépendant de la densité, connue *a priori*, pour l'interpolation d'images de tumeurs qui grossissent.
Depuis quelques années, de nombreuses autres applications au traitement d'images sont étudiées, telles que les images en couleurs (RGB) [35, 36], le transfert de couleurs [71], la segmentation d'images [70, 68] et le calcul de barycentres [72]. Une étude a également été réalisée sur l'interpolation d'images [69, 70, 72].

Dans ce travail nous nous concentrons sur le problème d'interpolation d'images et la résolution numérique du problème de transport optimal dynamique, pour lequel il existe peu de méthodes numériques jusqu'à présent [7, 43, 69], par opposition au transport statique [8, 9, 17, 44] . Du point de vue numérique, le développement de nouveaux algorithmes efficaces est toujours un challenge notamment parce que le calcul du transport optimal entre deux densités n'est pas un problème strictement convexe. Nous utilisons le travail précurseur de Benamou-Brenier [7], plaçant le problème dans un contexte de mécanique des fluides en ajoutant une dimension temporelle. Les algorithmes existants pour cette formulation nécessitent une projection sur la contrainte de divergence nulle à chaque itération, ce qui équivaut à la résolution d'une équation de Poisson en 3D, déjà très coûteuse, pour des images 2D. Dans cette thèse, nous travaillons dans l'espace des contraintes, contenant la contrainte de divergence nulle, pour éviter cette projection.

Le second chapitre est dédié à la présentation du problème de transport optimal que l'on souhaite étudier, en commençant par une introduction générale des problèmes de Monge et Kantorovich, puis la formulation mécanique des fluides sur laquelle nous travaillons. Nous définissons ensuite l'espace des contraintes que nous utilisons et montrons qu'en se plaçant dans cet espace, la fonctionnelle que l'on minimise pour obtenir le transport optimal a de meilleures propriétés de convexité. Nous rappelons ensuite quelques propriétés du transport optimal qui seront observées dans le chapitre 5 des applications numériques.

Dans le troisième chapitre nous proposons l'utilisation de trois représentations différentes permettant la prise en compte de la divergence nulle du vecteur à étudier pour rester dans l'espace des contraintes. Nous introduisons tout d'abord des bases d'ondelettes à divergence nulle périodiques ainsi que des bases prenant en compte des conditions sur le bord de type Dirichlet. Nous nous sommes également intéressés à deux autres représentations des vecteurs à divergence nulle. Nous avons tout d'abord représenté les vecteurs par leur fonction de courant munie d'un relèvement des conditions au bord, dans le cas du transport 1D. Ensuite, nous avons représenté les vecteurs à l'aide de la décomposition de Helmholtz-Hodge pour des transports 1D et 2D. Nous montrons que la minimisation de la fonctionnelle issue du transport optimal est équivalente en 1D dans ces deux cas à la

résolution d'une équation de type courbure minimale sur les lignes de niveau du potentiel.

Dans le quatrième chapitre, pour mettre en oeuvre les différentes formulations du chapitre 3, nous détaillons les différents algorithmes que nous utilisons. Tout d'abord, nous présentons un algorithme de descente de gradient sur les coefficients d'ondelettes à divergence nulle, le calcul du gradient pouvant se faire de deux manières différentes. Pour implémenter les décompositions sur bases d'ondelettes à divergence nulle nous utilisons une descente de gradient à pas fixe pour valider notre approche. Cet algorithme est lent mais permet dans un premier temps de montrer la faisabilité de cette approche d'un point de vue numérique. Pour les formulations impliquants les fonctions de courant ainsi que les décompositions de Helmholtz-Hodge, nous implémentons les algorithmes primaux-duaux développés par Chambolle et Pock. Ces algorithmes sont rapides, simples à mettre en oeuvre numériquement et aisément parallélisables.

Le dernier chapitre est consacré aux tests numériques. Nous décrivons le cadre discret dans lequel nous travaillons et en particulier les différentes grilles centrées et décalées utilisées. Nous analysons ensuite la descente de gradient utilisée avec la décomposition sur ondelettes périodiques. Celle-ci a une convergence lente qui valide notre approche d'un point de vue numérique mais doit être améliorée avec l'utilisation des ondelettes prenant en compte des conditions au bord de type Dirichlet. Nous montrons ensuite que nos algorithmes primaux-duaux à l'aide des fonctions de courant et de la décomposition de Helmholtz-Hodge pour des images 1D et 2D sont plus performants que l'état de l'art. De plus, nous avons implémenté et parallélisé avec OpenMP l'algorithme pour images 2D en C++. Nous montrons par ailleurs des exemples avec des densités de masses différentes ainsi que pour des densités de support compact. Nous finissons par deux applications à des images réelles munies d'explications des résultats observés.

Chapitre 2

Propriétés du problème de Monge-Kantorovich

Nous introduisons dans ce chapitre la problématique du transport optimal et définissons un ensemble des contraintes approprié dans lequel nous proposons de travailler. Dans cet objectif, nous explicitons tout d'abord le problème à l'origine du transport optimal, ainsi que les deux premières formulations définies par Monge puis Kantorovich. Par la suite, la formulation de type mécanique des fluides qui nous intéresse est détaillée, et plus particulièrement l'algorithme de Benamou-Brenier. Nous prouvons enfin un résultat de convexité pour le problème que nous considérons, justifiant notre démarche.

Ce chapitre se termine par l'explication de résultats connus du transport optimal dynamique, qui seront observés dans les applications numériques (voir chapitre 5).

2.1 Introduction

Le géomètre Gaspard Monge publia en 1871 "*Mémoire sur la théorie des déblais et des remblais*" [66], où il détailla un problème de transfert de masse dans le domaine de l'ingénierie civile. Ce problème, fondateur du transport optimal, est le suivant : ayant un tas de matériau (du sable par exemple) que l'on veut transporter et utiliser à un autre endroit (remplir un trou, construire un chateau,...), et sachant que le transport de ce matériau a un coût, comment déplacer chaque élément du tas pour que le coût total soit minimal ?
Monge partit du principe, dans son exemple, que le coût du déplacement d'une unité de masse est le produit de la masse déplacée et de la distance parcourue.

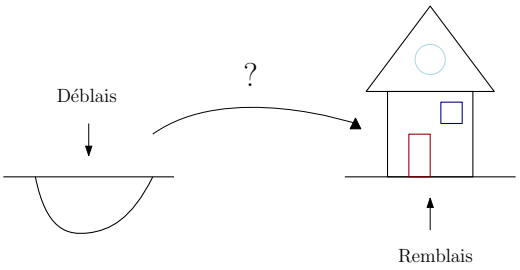
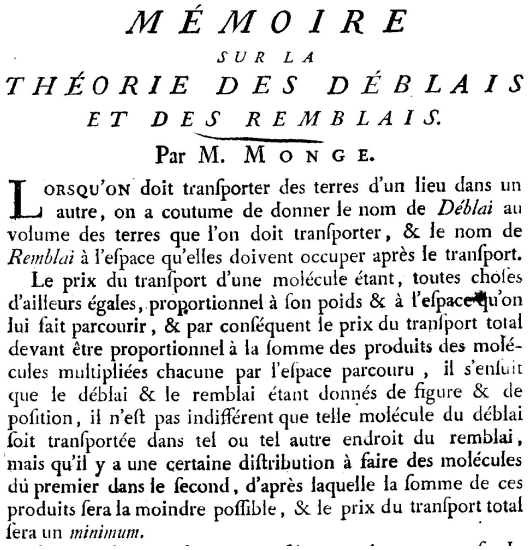


FIGURE 2.1: Déplacement de matériau d'un site à un autre en minimisant le coût de transport.



MÉMOIRE
SUR LA
THÉORIE DES DÉBLAIS
ET DES REMBLAIS.
Par M. MONGE.

LORSQU'ON doit tranſporter des terres d'un lieu dans un autre, on a coutume de donner le nom de *Déblai* au volume des terres que l'on doit tranſporter, & le nom de *Remblai* à l'eſpace qu'elles doivent occuper après le tranſport.

Le prix du tranſport d'une molécule étant, toutes choſes d'ailleurs égales, proportionnel à ſon poids & à l'eſpace qu'on lui fait parcourir, & par conſéquent le prix du tranſport total devant être proportionnel à la ſomme des produits des molécules multipliées chacune par l'eſpace parcouru, il s'enſuit que le déblai & le remblai étant donnés de figure & de poſition, il n'eſt pas indifférent que telle molécule du déblai ſoit tranſportée dans tel ou tel autre endroit du remblai, mais qu'il y a une certaine diſtribution à faire des molécules du premier dans le ſecond, d'après laquelle la ſomme de ces produits ſera la moindre poſſible, & le prix du tranſport total ſera un *minimum*.

FIGURE 2.2: Première page du mémoire de Gaspard Monge.

Monge étudia le problème en trois dimensions pour des distributions de masses continues. Il observa que le transport devait se faire en lignes droites orthogonales à des familles de surfaces. Plus tard, le mathématicien russe Leonid Vitaliyevich Kantorovich, qui reçut le prix Nobel d'économie en 1975 pour sa contribution à la théorie des allocations optimales des ressources, se pencha également sur le problème en 1942 [55]. Il ne réalisa qu'après que ses recherches étaient liées à celles de Monge [56]. Il définit une notion de distance entre des mesures de probabilité : la distance entre deux mesures devrait être le coût du transport optimal de l'une à l'autre, si la fonction coût est la fonction distance. Cette distance est à présent connue sous le nom de distance de Wasserstein.

2.1.1 Problème de Monge et problème de Kantorovich

Problème de Monge. Soient μ une mesure sur l'espace X, ν une mesure sur l'espace Y, A un ensemble μ-mesurable sur X et B ν-mesurable sur Y. Dans l'exemple décrit plus haut, $\mu(A)$ est la quantité de sable dans A et $\nu(B)$ est la quantité de sable déplaçable

dans B. Il nous faut à présent définir le coût du déplacement évoqué plus haut, appelé **coût de transport** :

$$c : \begin{array}{ccc} X \times Y & \to & \mathbb{R} \cup \{+\infty\} \\ (x, y) & \mapsto & c(x, y), \end{array}$$

il s'agit du coût du déplacement d'une unité de masse de sable de x à y. Monge partit du principe suivant : la masse présente en x ne peut être divisée pour se retrouver en plusieurs endroits. Il existe donc une application mesurable $T : X \to Y$ telle que $y = T(x)$.
On a alors pour tout $B \subset Y, \nu(B) = \mu(T^{-1}(B))$, c'est-à-dire que ν est la **mesure image**

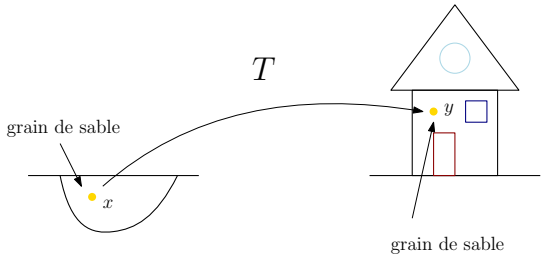


FIGURE 2.3: Déplacement du grain de sable en x vers y avec l'application T.

de μ par T. On peut également écrire que T **transporte** μ sur ν : $T\sharp\mu = \nu$. Pour une application $T \in C^1$ sur X et μ et ν admettant les densités ρ_0 et ρ_1 de même masse, le transport T de μ sur ν s'écrit comme la relation non linéaire suivante :

$$\rho_1(T(x))|\det(\nabla T(x))| = \rho_0(x). \tag{2.1}$$

En effet, dans ce cas, l'égalité $\nu(B) = \mu(T^{-1}(B))$, pour $B \subset Y$, s'écrit

$$\int_B \rho_1(x)dx = \int_{T(x)\in B} \rho_0(x)dx,$$

et on obtient l'équation (2.1) avec un simple changement de variable.
Le problème de Monge est alors le suivant :

$$\inf_{\{T:X\to Y, T\sharp\mu=\nu\}} \int_X c(x, T(x))d\mu(x), \tag{2.2}$$

dont la valeur est le **coût du transport optimal** et la solution le **plan de transport optimal**. Ce problème n'a été résolu que pour certaines fonctions coût, pour $c(x, y) = |x - y|$ et $X = Y = \mathbb{R}^n$ en 1999 par Evans et Gangbo [30], en 1996 pour les fonctions $c(x, y) = h(x - y)$ pour h strictement concave ou strictement convexe et $X = Y = \mathbb{R}^n$ par Gangbo et McCann [37].

Problème de Kantorovich. Pour le problème de Kantorovich, on relaxe le problème de Monge en considérant que l'on peut diviser une unité de masse pour la déplacer en

plusieurs endroits. Il faut donc définir la mesure de probabilité du déplacement de la masse de x à y :

$$\pi : X \times Y \to \mathbb{R}^+.$$

On appelera $\pi \in P(X \times Y)$ le **plan de transfert**. Cette mesure admet μ et ν comme

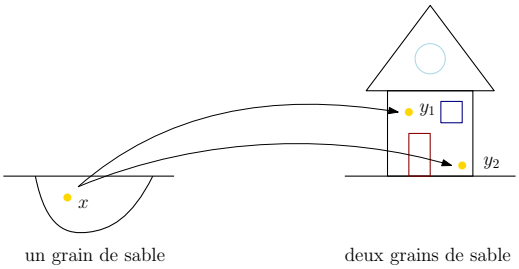


FIGURE 2.4: Division du grain de sable en x et déplacement vers y_1 et y_2.

marginales, c'est-à-dire que μ (respectivement ν) est la mesure image de π par $(x, y) \mapsto x$ (respectivement $(x, y) \mapsto y$), ce qui revient à

$$\pi(A \times Y) = \mu(A) \quad \text{et} \quad \pi(X \times B) = \nu(B). \tag{2.3}$$

En d'autres termes toute la masse déplacée de x doit être égale à $d\mu(x)$ et toute la masse déplacée vers y doit être égale à $d\nu(y)$:

$$\int_Y d\pi(x, y) = d\mu(x) \qquad \int_X d\pi(x, y) = d\nu(y) \tag{2.4}$$

On dénote par $\Pi(\mu, \nu)$ l'ensemble des mesures de probabilité sur $X \times Y$ vérifiant (2.3). Cet ensemble est non nul car $\mu \otimes \nu \in \Pi(\mu, \nu)$, ce transfert de masse consiste à prendre chaque grain de sable du tas de départ et à le répartir uniformément sur le trou d'arrivée, ce qui n'est pas optimal. Le problème de Kantorovich s'écrit alors :

$$\inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} c(x, y) d\pi(x, y). \tag{2.5}$$

La solution de ce problème est le **coût du transport optimal** entre μ et ν. Le π réalisant ce minimum sera le **plan de transfert optimal**. Il suffit que c soit semi continue inférieurement pour qu'il y ait une solution au problème de Kantorovich. Il n'y a en revanche pas unicité en général. Le problème de Kantorovich est bien une relaxation du problème de Monge car si on considère une application de transport T qui est solution du problème de Monge, alors $\pi = (Id \times T)\sharp\mu$ est le plan de transfert optimal solution du problème de Kantorovich. Inversement, un plan de transport est induit par une application T obtenue par le transport optimal, si π est concentré sur un graphe Γ π-mesurable.
Le cas qui nous intéressera dans la suite, si on dénote par $|.|$ la norme euclidienne sur $\mathbb{R}^n$, est le coût $c(x, y) = |x - y|^2$, qui comporte des propriétés intéressantes et un lien avec la mécanique des fluides.

2.1.2 Existence et gradient de fonctions convexes

McCann [62] a prouvé **l'existence et l'unicité** de la solution pour le problème de Monge, pour la fonction coût c qui nous intéresse. Brenier [12] ainsi que Knott et Smith [81] ont initié cette approche que McCann et Gangbo [37, 62] ont ensuite retravaillée. Dans ces travaux, il est prouvé que les applications **solutions, T, sont gradients de fonction convexe**, $T = \nabla\psi$. C'est ce qu'on appelle le théorème de Brenier.
Cela vient des deux propriétés suivantes :

— Comme prouvé par Rockafellar dans [74], un ensemble est inclus dans le sous-différentiel d'une fonction propre, convexe et semi-continue inférieurement si et seulement s'il est cycliquement monotone. Un ensemble E est cycliquement monotone si pour tout $m \geq 1$, et pour tous $(x_1, y_1), ..., (x_m, y_m)$ dans E,

$$\sum_{i=1}^{m} |x_i - y_i|^2 \leq \sum_{i=1}^{m} |x_i - y_{i-1}|^2$$

où $y_0 = y_m$.

— De plus, il a été prouvé dans [37] que le plan de transfert optimal entre les mesures μ et ν, solution du problème de Kantorovich, est à support cycliquement monotone.

On obtient donc pour μ et ν des mesures de probabilités sur $\mathbb{R}^n$, qu'un plan de transfert optimal pour le problème de Kantorovich est à support dans le sous-différentiel d'une fonction propre, semi continue inférieurement et convexe.

2.2 Formulation mécanique des fluides

2.2.1 Description de la distance L^2 de Wasserstein

Dans les cas détaillés ci-dessus le coût du déplacement ne dépend que des états initiaux et finaux. Pour pouvoir mieux prendre en compte les éventualités pendant le trajet, par exemple les obstacles, il faut pouvoir analyser ce qu'il se passe à chaque étape du trajet et donc introduire une variable de temps. On peut ainsi étudier les trajectoires de chacune des particules. On peut remarquer que dans la formulation de Monge était déjà contenue une notion de trajectoire même si la variable de temps a été éliminée de façon à réduire la dimension du problème. Benamou et Brenier [7] ont défini une formulation mécanique des fluides en réintroduisant la variable t pour obtenir un problème de minimisation espace-temps non quadratique mais convexe en la densité et le moment avec des contraintes affines à la place du problème de minimisation en espace, quadratique, non convexe et aux contraintes non linéaires.
On se place maintenant dans $\Omega = (0, 1)^n$, avec $n > 0$, et on considère que les mesures μ et ν sont associées aux densités ρ_0 et ρ_1 positives ou nulles bornées telles que

$$\int_\Omega \rho_0(x)dx = \int_\Omega \rho_1(x)dx = 1.$$

On peut par exemple voir ρ_0 et ρ_1 comme la densité de particules dans un compact à un instant $t = 0$ et à un instant $t = 1$. On considère l'ensemble des applications qui transportent $\rho_0(x)dx$ sur $\rho_1(x)dx$ et on va chercher à résoudre

$$\min_{T,\ \rho_1 = T\sharp\rho_0} \int c(x, T(x))\rho_0(x)dx. \tag{2.6}$$

Comme on s'intéresse ici au cas $c(x, y) = |x - y|^2$:

$$d_2(\rho_0, \rho_1) = \sqrt{\min_{T,\ \rho_1 = T\sharp\rho_0} \int |x - T(x)|^2 \rho_0(x)dx} \tag{2.7}$$

et on appelle ce minimum la **distance L^2 de Wasserstein** entre ρ_0 et ρ_1 ([83]). Au lieu de calculer l'application de transport on peut calculer le **chemin géodésique** entre les densités ρ_0 et ρ_1 selon cette distance.

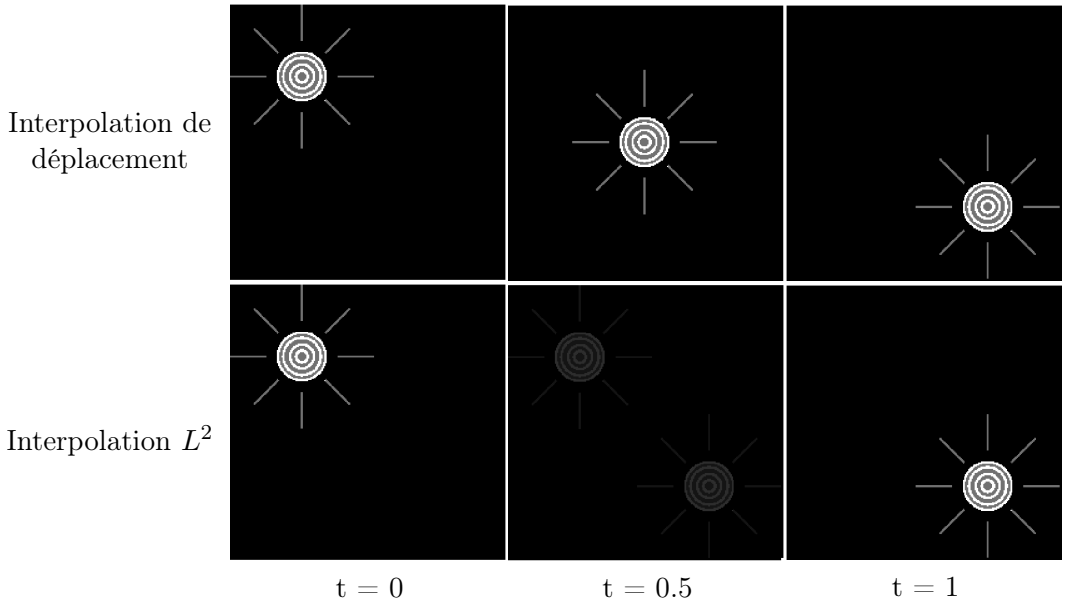


FIGURE 2.5: Différence entre l'interpolation L^2 et l'interpolation de déplacement.

On va pour cela considérer un intervalle de temps $(0, \tau)$ ($t \in (0, \tau)$ paramétrise le chemin). On prendra $\tau = 1$ dans la suite sans perte de généralité. On définit ensuite $Q = (0, 1) \times \Omega$ et l'espace de fonctions suivant :

$$V(Q) = \{f \in (L^2(Q))^{1+n},\ \mathrm{div}_{t,x}\, f = 0\}. \tag{2.8}$$

Etant donné T^* réalisant (2.6), McCann [63] définit sous le nom d'interpolation de déplacement la géodésique reliant ρ_0 à ρ_1. C'est la fonction $(t, x) \to \rho(t, x) \in \mathbb{R}^+$, densité du chemin géodésique entre ρ_0 et ρ_1 donnée par :

$$\rho(t, x) = \rho_0(T_t(x))|\det(\nabla_x T_t(x))|,\ \text{où } T_t = (1 - t)Id + tT^*. \tag{2.9}$$

Il faut à présent considérer le champ de vecteurs, que l'on peut voir comme le champ de vitesse qui déplace les particules, $v(t,x) \in \mathbb{R}^n$. Le couple (ρ, v) vérifie l'équation de continuité

$$\mathrm{div}_{t,x}(\rho(t,x), \rho v(t,x)) = \partial_t \rho(t,x) + \nabla_x \cdot \rho v(t,x) = 0 \tag{2.10}$$

pour $t \in (0,1)$ et $x \in \Omega$, munie des conditions initiales et finales suivantes

$$\rho(0,x) = \rho_0(x),\ \rho(1,x) = \rho_1(x),\ \forall x \in \Omega. \tag{2.11}$$

Benamou et Brenier ont démontré dans [7] la proposition suivante :

Proposition 1 (Benamou-Brenier)**.**

$$d_2(\rho_0,\rho_1)^2 = \inf_{T,\ \rho_1 = T\sharp\rho_0} \int_{\mathbb{R}^n} |T(x)-x|^2 \rho_0(x) dx = \inf_{(\rho,\rho v)\in C_{BB}} \int_0^1 \int_{\mathbb{R}^n} \rho(t,x)|v(t,x)|^2 dxdt \tag{2.12}$$

où $C_{BB} = \{(\rho, v); \partial_t \rho + \nabla.(\rho v) = 0,\ \rho(0,.) = \rho_0 \ \textit{et}\ \rho(1,.) = \rho_1\}$

Remarque. Minimiser cette nouvelle fonctionnelle est un problème non convexe dont les contraintes sont non linéaires.

Démonstration. Cette preuve se fera en deux étapes. On montrera formellement que l'un des deux infima est plus petit que l'autre puis on montre que pour un couple $(\rho, \rho v)$ particulier ils sont égaux. Pour une démonstration détaillée voir [83, 42]. On considère ρ_0 et ρ_1 bornées et à support compact dans $\mathbb{R}^n$ et ρ, v suffisament régulières dans C_{BB}. Soit $X(t,x)$ la caractéristique définie par :

$$X(0,x) = x,\ \partial_t X(t,x) = v(t, X(t,x)) \text{ et } \rho(t,.) = X(t,.)\sharp\rho_0 \text{ pour tout } t. \tag{2.13}$$

Pour toute fonction test f, on a

$$\begin{aligned} \iint f(t,x)\rho(t,x)dxdt &= \iint f(t,X(t,x))\rho_0(x)dxdt, \\ \iint f(t,x)\rho(t,x)v(t,x)dxdt &= \iint \partial_t X(t,x) f(t,X(t,x))\rho_0(x)dxdt. \end{aligned} \tag{2.14}$$

Les conditions au bord de ρ (2.11) et l'équation (2.14) impliquent que $X(1,x)$ satisfait, pour tout $A \subset \mathbb{R}^n$ l'équation

$$\int_{x\in A} \rho_1(x)dx = \int_{X(1,x)\in A} \rho_0(x)dx,$$

tout comme l'application de transfert optimale $T = \nabla\psi(x)$. Ensuite on a

$$\begin{aligned}
\int_{\mathbb{R}^n}\int_0^1 \rho(t,x)|v(t,x)|^2 dxdt &= \int_{\mathbb{R}^n}\int_0^1 \rho_0(x)|v(t,X(t,x))|^2 dtdx \\
&= \int_{\mathbb{R}^n}\int_0^1 \rho_0(x)|\partial_t X(t,x)|^2 dtdx \\
&= \int_{\mathbb{R}^n} \rho_0(x)\int_0^1 |\partial_t X(t,x)|^2 dtdx \\
&\geq \int_{\mathbb{R}^n} \rho_0(x)|X(1,x) - X(0,x)|^2 dx \text{ (inégalité de Jensen)} \\
&= \int_{\mathbb{R}^n} \rho_0(x)|X(1,x) - x|^2 dx \\
&\geq \int_{\mathbb{R}^n} \rho_0(x)|\nabla\psi(x) - x|^2 dx
\end{aligned}$$

car $X(1,x)$ et $\nabla\psi(x)$ transportent ρ_0 sur ρ_1 et $\nabla\psi$ est le plan optimal.
Si on trouve maintenant (ρ, v) tel qu'on ait égalité, la preuve sera complète. Choisissons :

$$X(t,x) = x + t(\nabla\psi(x) - x)$$

où $\nabla\psi$ est l'application optimale pour le problème de Monge et ρ est le transport de ρ_0 par X (cf. (2.13)). En reprenant le raisonnement de la première étape, le couple (ρ, v) défini par (2.13) vérifie l'équation (2.10) et on a

$$\begin{aligned}
\int \rho(t,x)|v(t,x)|^2 dx &= \int \rho_0(x)|v(t,X(t,x))|^2 dx \\
&= \int \rho_0(x)|\partial_t X(t,x)|^2 dx \\
&= \int \rho_0(x)|\nabla\psi(x) - x|^2 dx,
\end{aligned}$$

ce qui prouve la proposition. □

L'existence de minimiseurs pour cette fonctionnelle a été prouvée pour le problème dans l'espace des mesures de Radon [83]. C'est un sous-espace de l'espace des distributions donc on peut considérer les contraintes au sens des distributions. C'est une version variationelle relaxée du problème étudié dans cette section, qui est défini dans un cadre hilbertien. L'étude du problème dans un espace de Hilbert a fait l'objet de travaux réalisés par Guittet [42, 41] et Hug [46].

Remarque. On peut observer dans la preuve de la proposition qu'il est possible de retrouver l'application de transport optimal à partir de la vitesse optimale v. Pour $X(t,x)$ solution de (2.13), l'application de transport optimal est $X(1,x)$.

2.2.2 Algorithme Benamou-Brenier

Pour obtenir un problème convexe avec des contraintes linéaires, Benamou et Brenier ont introduit le moment $m = \rho v$ et la formulation suivante

$$\min_{(\rho,m)\in C_{BB}} \mathcal{J}(\rho, m) \text{ avec } \mathcal{J}(\rho, m) = \int_0^1 \int_\Omega J(\rho(t,x), m(t,x))dxdt, \tag{2.15}$$

et

$$\forall (\rho, m) \in \mathbb{R} \times \mathbb{R}^n,\ J(\rho, m) = \begin{cases} \frac{|m|^2}{2\rho}, & \text{si } \rho > 0, \\ 0, & \text{si } (\rho, m) = (0, 0), \\ +\infty, & \text{sinon .} \end{cases} \tag{2.16}$$

Pour résoudre ce problème, Benamou et Brenier ont considéré le lagrangien suivant :

$$L(\phi, \rho, m) = \int_0^1 \int_\Omega (J(\rho, m) - \partial_t \phi \rho - \nabla_x \phi \cdot m)\, dxdt - \int_\Omega (\phi(0, x)\rho_0(x) - \phi(1, x)\rho_1(x))\, dx. \tag{2.17}$$

Le problème est donc équivalent au problème de point-selle :

$$\inf_{\rho,m} \sup_{\phi} L(\phi, \rho, m).$$

Une méthode dite de lagrangien augmenté est ensuite utilisée pour résoudre ce problème en mettant des conditions périodiques au bord pour le moment m.

Remarque. Cet algorithme bien que résolvant exactement le problème du transport optimal nécessite la résolution d'une équation de Poisson en 3D pour des images 2D à chaque itération, ce qui est très coûteux.

Dans la suite de notre travail, nous avons choisi des conditions sur le bord de type Neuman pour la vitesse alors que celles considérées par Benamou et Brenier sont périodiques. En effet, pour l'application à des images ce sont des conditions plus indiquées. Notre espace des contraintes devient dans ce cadre

$$C := \{(\rho, m);\ \mathrm{div}_{t,x}(\rho, m) = 0,\ m(., x).\nu_\Omega = 0,\ \forall x \in \partial\Omega,\ \rho(0, .) = \rho_0,\ \rho(1, .) = \rho_1\}. \tag{2.18}$$

Le but de cette thèse est de travailler directement dans cet espace, puisque comme nous allons le prouver dans la section suivante, la fonctionnelle $\mathcal{J}$ a de meilleures propriétés de convexité dans celui-ci moyennant une hypothèse de régularité.

2.3 Étude de la convexité dans l'espace des contraintes

Nous étudions dans cette section un résultat de convexité pour $\mathcal{J}$ sur l'espace des contraintes (2.18). Ce résultat impose de la régularité sur (ρ, m), régularité étudiée dans [46].

Proposition 2 (Hug)**.** *Le champ de vitesse dérivé du transport optimal, $v = \frac{m}{\rho}$ appartient à $W^{1,1}(Q)$. De plus, soit $M > 0$ et $\rho_0, \rho_1 \in L^\infty(\Omega)$ de même masse telles que $0 \leq \rho_0, \rho_1 \leq M$. Alors, la densité ρ est dans $L^\infty(Q)$ et $0 \leq \rho \leq M$.*

Considérons l'espace des contraintes :

$$C_\infty := \{(\rho, m) \in C;\ \rho \in L^\infty(Q),\ 0 \leq \rho \text{ et } m = \rho v,\ \text{avec } v \in L^1(Q)\}$$

Proposition 3. *L'espace C_∞ est non vide et convexe.*

Démonstration. Tout d'abord il faut noter que C_∞ n'est pas vide. En effet, d'après la proposition 2, la solution du problème de transport optimal, (ρ, m), pour $\rho_0, \rho_1 \in L^\infty(\Omega)$ et positives, est dans C_∞.
Soient maintenant (ρ, m) et (ρ', m') dans C_∞ et $\alpha \in]0,1[$, alors le couple $(\rho_\alpha, m_\alpha) = (\alpha\rho + (1-\alpha)\rho', \alpha m + (1-\alpha)m')$ est bien dans C qui est convexe et $\rho_\alpha \in L^\infty(Q)$ est positive. Vérifions à présent que $m_\alpha = \rho_\alpha v_\alpha$ avec $v_\alpha \in L^1(Q)$:

$$m_\alpha = \alpha m + (1-\alpha)m' = \big(\alpha\rho + (1-\alpha)\rho'\big)\,\frac{\alpha m + (1-\alpha)m'}{\alpha\rho + (1-\alpha)\rho'} = \rho_\alpha \frac{\alpha m + (1-\alpha)m'}{\alpha\rho + (1-\alpha)\rho'}.$$

Puisque ρ et ρ' sont positives, on a

$$\begin{aligned} v_\alpha = \left|\frac{m_\alpha}{\rho_\alpha}\right| = \left|\frac{\alpha m + (1-\alpha)m'}{\alpha\rho + (1-\alpha)\rho'}\right| &\leq \left|\frac{\alpha m}{\alpha\rho + (1-\alpha)\rho'}\right| + \left|\frac{(1-\alpha)m'}{\alpha\rho + (1-\alpha)\rho'}\right| \\ &\leq \left|\frac{\alpha m}{\alpha\rho}\right| + \left|\frac{(1-\alpha)m'}{(1-\alpha)\rho'}\right| = |v| + |v'| \in L^1(Q). \end{aligned}$$

Donc v_α est dans $L^1(Q)$ et C_∞ est convexe. □

Grâce aux propriétés du transport optimal citées plus haut on peut minimiser $\mathcal{J}$ dans C_∞ sans changement du minimiseur qui se trouve dans cet espace, et démontrer le résultat suivant :

Proposition 4. *La fonctionnelle $\mathcal{J}$ définie par (2.16) vérifie :*

1. *$\mathcal{J}$ est une fonction propre, semi continue inférieurement et convexe sur C_∞.*
2. *Si (ρ, m) et (ρ', m') dans C_∞ vérifient pour $\alpha \in]0,1[$*

$$\mathcal{J}(\alpha(\rho, m) + (1-\alpha)(\rho', m')) = \alpha\mathcal{J}(\rho, m) + (1-\alpha)\mathcal{J}(\rho', m'), \tag{2.19}$$

alors $\delta\rho = \rho - \rho'$ vérifie

$$\begin{cases} \partial_t(\delta\rho) + \nabla_x(w\delta\rho) = 0 \\ \delta\rho|_{\partial Q} = 0, \end{cases} \qquad \text{où} \qquad w = \begin{cases} v \text{ si } \rho > 0 \\ v' \text{ si } \rho' > 0 \\ 0 \text{ sinon} \end{cases}$$

où $v = v'$ si $\rho\rho' > 0$.

Démonstration. La première partie de ce théorème est démontrée dans [6].
Nous allons maintenant montrer le second point. On commence par prendre $\alpha \in]0,1[$, $(\rho, m) \in \operatorname{dom} \mathcal{J}$ et $(\rho', m') \in \operatorname{dom} \mathcal{J}$ tels que

$$\int_Q J(\alpha(\rho, m) + (1-\alpha)(\rho', m'))dxdt = \int_Q \left(\alpha J(\rho, m) + (1-\alpha)J(\rho', m')\right) dxdt. \quad (2.20)$$

Comme $m = \rho v$ avec $v \in L^1(Q)$, $m' = \rho' v'$ avec $v' \in L^1(Q)$ et $m_\alpha = \rho_\alpha v_\alpha$ avec $v_\alpha \in L^1(Q)$, on a pour presque tout $(t,x) \in Q$, $J(\rho, m) = \frac{1}{2}\rho v^2$ et donc on obtient :

$$\begin{aligned} 0 &= \int_Q \left(\rho_\alpha v_\alpha^2 - \alpha\rho v^2 - (1-\alpha)\rho' v'^2\right) dxdt \\ &= \int_Q \left(\left(\alpha\rho + (1-\alpha)\rho'\right) \left(\frac{\alpha m + (1-\alpha)m'}{\alpha\rho + (1-\alpha)\rho'}\right)^2 - \alpha\rho v^2 - (1-\alpha)\rho' v'^2 \right) dxdt \end{aligned}$$

Ce qui peut être réécrit

$$\int_Q \frac{|\alpha\rho v + (1-\alpha)\rho' v'|^2 \rho\rho' - (\alpha\rho'|\rho v|^2 + \rho(1-\alpha)|\rho' v'|^2)(\alpha\rho + (1-\alpha)\rho')}{(\alpha\rho + (1-\alpha)\rho')\rho\rho'} dxdt = 0.$$

En développant puis assemblant les termes, l'équation précédente devient

$$\int_Q \frac{(2(1-\alpha)\alpha\langle \rho v, \rho' v'\rangle)\rho\rho' - \alpha(1-\alpha)\rho'^2|\rho v|^2 - \alpha(1-\alpha)\rho^2|\rho' v'|^2}{(\alpha\rho + (1-\alpha)\rho')\rho\rho'} dxdt = 0,$$

en divisant finalement par $\alpha(1-\alpha) \neq 0$ on obtient :

$$\int_Q \frac{(\rho'\rho v - \rho\rho' v')^2}{(\alpha\rho + (1-\alpha)\rho')\rho\rho'} dxdt = 0.$$

Cela est équivalent à :

$$\rho'\rho v - \rho\rho' v' = 0 \text{ pour presque tout } (t,x) \in Q, \quad (2.21)$$

donc $v = v'$ si $\rho\rho' > 0$ et on peut définir

$$w = \begin{cases} v \text{ si } \rho > 0 \\ v' \text{ si } \rho' > 0 \\ 0 \text{ sinon,} \end{cases}$$

appartenant à $L^1(Q)$ car $|w|_{L^1} \leq |v|_{L^1} + |v'|_{L^1}$. En se rappelant que (ρ, m) et $(\rho', m') \in C_\infty$,

$$\begin{aligned} \partial_t\rho + \nabla_x \cdot m = \partial_t\rho + \nabla_x \cdot \rho v = \partial_t\rho + \nabla_x \cdot w\rho \quad (2.22) \\ \partial_t\rho' + \nabla_x \cdot m' = \partial_t\rho' + \nabla_x \cdot \rho' v' = \partial_t\rho' + \nabla_x \cdot w\rho' \end{aligned}$$

et on obtient bien le résultat souhaité

$$\partial_t(\delta\rho) + \nabla_x(w\delta\rho) = 0$$

avec $w \in L^1(Q)$. □

Remarques.

— La proposition 4 montre que $\delta\rho = \rho - \rho'$ est solution d'une équation de continuité, avec une vitesse $w \in L^1(Q)$ et une densité $\delta\rho \in L^\infty(Q)$ avec des conditions sur le bord nulles. Comme prouvé dans [28, 1], pour une vitesse dans $W^{1,1}(Q)$ il y a un unique couple solution et donc $\delta\rho = 0$. Cependant pour $w \in L^1(Q)$ on ne peut garantir que l'unique solution soit zéro.

— Si on ajoute la condition $v \in W^{1,1}(Q)$ dans l'ensemble C_∞, on ne peut pas prouver qu'il soit convexe.

Par la suite, nous considérerons ρ_0 et ρ_1 dans $L^\infty(\Omega)$ et positives.

2.4 Comportements connus

Plusieurs comportements sont particulièrement intéressants à détailler ici. En effet, dans certains cas, on peut calculer exactement le transport optimal et donc utiliser ces résultats pour valider nos applications numériques dans le chapitre 5. Les explications plus complètes de ces cas peuvent être trouvées dans [83, 7].

2.4.1 Déplacement en lignes droites

Dans ce paragraphe nous présentons maintenant une explication développée dans [83] (pages 144-145) des trajectoires que prennent les points x déplacés. Ce phénomène pourra être particulièrement bien observé dans les applications en une dimension de la partie 5. En se rappelant la formulation de Monge du problème de transport optimal :

$$\inf_{T,\ T\sharp\mu=\nu} \int_X c(x, T(x)) d\mu(x)$$

on veut maintenant pouvoir calculer les trajectoires de chaque particule située en x à $t = 0$, que l'on notera $(T_t(x))_{0\leq t\leq 1}$. Si on note le coût de ce déplacement $C[T_t(x)]$ et qu'on suppose que la trajectoire $t \to T_t(x)$ est continue et C^1 par morceaux $d\mu$-presque partout, on étudie à présent le problème suivant :

$$\inf_{T_t,\ T_0=Id,\ T_1\sharp\mu=\nu} \int_X C[(T_t(x))_{0\leq t\leq 1}] d\mu(x).$$

Pour que les deux problèmes soient compatibles, il faut qu'ils définissent tous les deux la même application optimale $T = T_1$ et le même coût. En effet, la somme du coût de déplacement de chaque x doit être la même que la somme du coût de chaque trajectoire qui va de x à $T(x)$.
Une condition suffisante pour cela est

$$c(x, y) = \inf_{T_t,\ T_0(x)=x,\ T_1(x)=y} C[(T_t(x))_{0\leq t\leq 1}]. \tag{2.23}$$

Pour $c(x,y) = |y-x|^2$ et $C[T_t] = \int_0^1 |\partial_t T_t|^2 dt$, l'égalité (2.23) est vérifiée. Comme $|.|^2$ est strictement convexe, l'inégalité de Jensen nous donne :

$$c(x,y) = c(T_0(x), T_1(x)) = \left| \int_0^1 \partial_t T_t(x) dt \right|^2 \leq \int_0^1 |\partial_t T_t(x)|^2 dt,$$

où il y a égalité si et seulement si $\partial_t T_t$ est constant. L'infimum de (2.23) est donc atteint en $T_t = x + t(y-x)$ et les seules **trajectoires optimales** sont les **lignes droites** parcourues à **vitesse constante**.

2.4.2 Déplacements par translation et homothétie

Supposons que ρ_1 soit obtenue à partir de ρ_0 par translation et homothétie, c'est-à-dire :

$$\rho_1(x) = \lambda^n \rho_0(\lambda(x-p))$$

pour $p \in \mathbb{R}^n$ et $\lambda > 0$, ce que l'on peut réecrire

$$\frac{1}{\lambda^n} \rho_1\left(\frac{x+\lambda p}{\lambda}\right) = \rho_0(x).$$

D'après l'équation (2.1), l'application

$$T(x) = \frac{1}{\lambda} x + p$$

transporte ρ_0 sur ρ_1. De plus c'est le gradient d'une fonction convexe $T(x) = \nabla\left(\frac{1}{2\lambda}|x|^2 + p \cdot x\right)$. On en déduit que T est l'application qui réalise le transport optimal.

— Si $\lambda = 1$, la distance L^2 de Wasserstein est

$$\boxed{d_2(\rho_0, \rho_1)^2 = |p|^2.}$$

— Si $\lambda \neq 1$, et qu'on suppose en plus que

$$\int x \rho_0(x) dx = 0 \text{ et } \int x^2 \rho_0(x) = 1 \tag{2.24}$$

la distance L^2 de Wasserstein est

$$\boxed{d_2(\rho_0, \rho_1)^2 = (1 - \tfrac{1}{\lambda})^2 + |p|^2.}$$

2.4.3 Cas unidimensionnel

Sur $\mathbb{R}$, les gradients de fonctions convexes sont croissants et les sous-gradients sont des graphes croissants complets, c'est-à-dire le graphe d'une fonction croissante que l'on

rend continue en les x où les limites à droite et à gauche ne sont pas les mêmes. Donc le transport de ρ_0 sur ρ_1 par T s'écrit pour $A =]0, T(x)[$

$$\int_0^{T(x)} \rho_1(y)dy = \int_{T(y)\in A} \rho_0(y)dy.$$

Sur]0,1[, T(0) = 0 et donc

$$\int_0^{T(x)} \rho_1(y)dy = \int_0^{x} \rho_0(y)dy.$$

Soit $F_0(x) = \int_{-\infty}^{x} d\rho_0$ la fonction de répartition de ρ_0 et F_1 celle de ρ_1, nous avons

$$F_1(T(x)) = F_0(x).$$

On peut définir l'inverse généralisée de F sur $[0,1]$ par $F^{-1}(t) = \inf\{x \in \mathbb{R};\ F(x) > t\}$, alors l'application qui réalise le transport optimal s'écrit :

$$T(x) = F_1^{-1}(F_0(x))$$

et la distance L^2 de Wasserstein est

$$\boxed{d_2(\rho_0, \rho_1)^2 = \int_0^1 |F_1^{-1}(t) - F_0^{-1}(t)|^2 dt.}$$

Chapitre 3

Reformulation du problème à l'aide de décompositions de vecteurs à divergence nulle

Pour travailler dans l'espace des contraintes C défini par (2.18), nous utilisons des décompositions permettant de préserver la contrainte de divergence nulle. En effet, les algorithmes actuels utilisés pour résoudre le problème du transport optimal dynamique nécessitent la projection sur cette contrainte de divergence nulle, qui est effectuée en résolvant une équation de Poisson à chaque itération. Cette résolution étant coûteuse, nous proposons l'utilisation d'outils pour préserver la divergence nulle à tout instant, et ainsi augmenter la performance des algorithmes existants.
La première idée décrite dans ce chapitre est de décomposer les vecteurs sur des bases d'ondelettes à divergence nulle, soit périodiques, soit prenant en compte des conditions de type Dirichlet homogènes sur le bord. Ces ondelettes ont été utilisées pour la simulation des équations de Navier-Stokes incompressibles [50]. L'étude de ces bases a amené l'idée d'utiliser d'autres décompositions classiques en mécanique des fluides. Les deux que nous avons étudiées et qui sont ensuite développées ici sont la représentation d'un vecteur à divergence nulle par sa fonction de courant munie d'un relèvement des conditions sur le bord, et la décomposition de Helmholtz-Hodge d'un vecteur général. Nous démontrons ensuite que lorsqu'on considère des densités 1D, la minimisation de la fonctionnelle issue du transport optimal à l'aide de ces décompositions est équivalente à la résolution d'une équation de type courbure minimale sur les lignes de niveau du potentiel.

3.1 Bases d'ondelettes à divergence nulle

Nous détaillons dans cette partie la construction de bases d'ondelettes à divergence nulle anisotropes développées par Deriaz [27, 26], puis leur extension au cas non périodique avec des bases à divergence nulle sur $[0,1]^n$, prenant en compte des conditions aux limites développées par Kadri-Harouna [50]. L'utilisation de ces bases pour l'analyse de

la turbulence ou pour la résolution des équations de Navier-Stokes donne des résultats convaincants, justifiant leur application au problème de transport optimal. La recherche de telles bases remonte aux travaux de G. Battle et P. Federbush [5] et Lemarié [58]. Elles permettent pour les algorithmes résolvant les problèmes de transport optimal dynamique de préserver la contrainte de divergence nulle. De plus, un gain en complexité est envisageable dans la mesure où on n'utilise qu'un petit nombre d'ondelettes qui se concentrent dans les zones de singularité.

3.1.1 Généralités sur les ondelettes 1D

Les **analyses multirésolutions** (AMR) sont des espaces d'approximation générés par des bases de fonctions, introduites par Mallat [60].

AMR de $L^2(\mathbb{R})$

On définit tout d'abord une analyse multirésolution, que l'on dénotera AMR par la suite, sur la droite réelle.

Définition 1. *Une AMR de $L^2(\mathbb{R})$ est une suite croissante de sous espaces fermés $(V_j)_{j\in\mathbb{Z}}$ vérifiant :*

i) $\forall j \in \mathbb{Z},\ V_j \subset V_{j+1}$,
ii) $\forall j \in \mathbb{Z},\ f(t) \in V_j \Leftrightarrow f(2t) \in V_{j+1}$,
iii) $\overline{\bigcap_{j=-\infty}^{+\infty} V_j} = \{0\}$,
iv) $\overline{(\bigcup_{j=-\infty}^{+\infty} V_j)} = L^2(\mathbb{R})$.
v) Il existe $\varphi \in V_0$ tel que $\{\varphi(t-n)\}_{n\in\mathbb{Z}}$ est une base de Riesz de V_0.

Quelques remarques sur cette définition :

— La fonction φ est appelée **fonction d'échelle**. D'après (ii) il existe $\{h_k\}_{k\in\mathbb{Z}}$ une suite réelle appelée filtre d'échelle telle que

$$\varphi(x) = \sqrt{2}\sum_{k\in\mathbb{Z}} h_k \varphi(2x-k). \tag{3.1}$$

φ est à support compact ($[n_{min}, n_{max}]$) si et seulement si un nombre fini de termes de $\{h_k\}_k$ sont non nuls [53]. Chaque espace V_j est généré par la famille de fonctions

$$\{\varphi_{j,k}\}_{k\in\mathbb{Z}};\ \varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k).$$

— Le paramètre j est la résolution. D'après (i), on peut écrire

$$V_{j+1} = V_j \oplus W_j. \tag{3.2}$$

W_j est un supplémentaire direct de V_j dans V_{j+1} qui n'est pas unique. Les **ondelettes** se définissent comme bases des espaces W_j, engendrées par une fonction

$\psi \in W_0$. ψ vérifie

$$\psi(x) = \sqrt{2} \sum_k g_k \varphi(2x - k), \tag{3.3}$$

où $\{g_k\}_{k\in\mathbb{Z}}$ est une suite réelle appelée filtre de détails. La famille $\{\psi(. - k)\}_{k\in\mathbb{Z}}$ est une base de Riesz de W_0 et $\{\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)\}_{k\in\mathbb{Z}}$ est une base de Riesz de W_j appelée **base d'ondelettes**.

— Les relations (3.2) et les relations (iii) et (iv) impliquent :

$$\boxed{L^2(\mathbb{R}) = \bigoplus_{j\in\mathbb{Z}} W_j} \tag{3.4}$$

et donc $\{\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)\}_{j,k\in\mathbb{Z}}$ est une base de Riesz de $L^2(\mathbb{R})$.

En pratique W_j, est choisi soit comme le complémentaire orthogonal de V_j dans V_{j+1} (voir [64] pour plus de détails) soit comme un supplémentaire biorthogonal de V_j dans V_{j+1}. C'est cette dernière construction qui va nous intéresser ici.

Analyse multirésolution biorthogonale

Une **analyse multirésolution biorthogonale** (AMRB) de $L^2(\mathbb{R})$ est constituée de deux AMR de $L^2(\mathbb{R})$ notées $\{V_j\}_{j\in\mathbb{Z}}$ et $\{\tilde{V}_j\}_{j\in\mathbb{Z}}$ qui vérifient :

$$\begin{gathered} V_{j+1} = V_j \oplus W_j \text{ et } \tilde{V}_{j+1} = \tilde{V}_j \oplus \tilde{W}_j \\ V_j \perp \tilde{W}_j,\ \tilde{V}_j \perp W_j,\ \tilde{W}_j \perp W_{j'},\ j \neq j' \\ W_j = V_{j+1} \cap (\tilde{V}_j)^\perp \iff L^2(\mathbb{R}) = V_j \oplus (\tilde{V}_j)^\perp. \end{gathered}$$

Plus de précisions sur ces bases peuvent être trouvées dans [49, 60, 20].

Décomposition et reconstruction

L'analyse multirésolution d'une fonction $f \in L^2(\mathbb{R})$ consiste à décomposer f en une **approximation grossière** $f_0 \in V_0$ et tous les **détails** se trouvant dans les espaces W_j, pour $j \geq 0$.

On note la projection de f dans l'espace V_j parallèlement à l'espace $\left(\tilde{V}_j\right)^\perp$:

$$\mathcal{P}_j(f) = \sum_k \langle f, \tilde{\varphi}_{j,k}\rangle \varphi_{j,k}$$

et la projection sur les espaces de détails W_j parallèlement à $\tilde{W}_j^\perp$ par :

$$\mathcal{Q}_j(f) = \sum_k \langle f, \tilde{\psi}_{j,k}\rangle \psi_{j,k}.$$

On notera dans la suite $c_{j,k} = \langle f, \tilde{\varphi}_{j,k}\rangle$ et $d_{j,k} = \langle f, \tilde{\psi}_{j,k}\rangle$.

On a alors pour toute fonction $f \in L^2(\mathbb{R})$

$$f = \sum_k c_{0,k}\varphi_{0,k} + \sum_{j\geq 0}\sum_k d_{j,k}\psi_{j,k}.$$

La transformée en ondelettes est l'application qui associe à f les coefficients $\{c_{0,k}\}_{k\in\mathbb{Z}} \cup \{d_{j,k}\}_{j\geq 0,k\in\mathbb{Z}}$. De plus, la relation (3.4) implique

$$\boxed{f = \sum_{j,k\in\mathbb{Z}} d_{j,k}\psi_{j,k}.}$$

Pour la reconstruction, il faut calculer f à l'aide de son approximation grossière et de ses détails grâce aux relations satisfaites par les fonctions d'échelle :

$$\varphi_{j+1,m} = \sum_k \tilde{h}_{m-2k}\varphi_{j,k} + \sum_k \tilde{g}_{m-2k}\psi_{j,k}$$
$$\tilde{\varphi}_{j+1,m} = \sum_k h_{m-2k}\tilde{\varphi}_{j,k} + \sum_k g_{m-2k}\tilde{\psi}_{j,k},$$

où $\{h_k\}_{k\in\mathbb{Z}}$ est le filtre d'échelle défini par (3.1), $\{g_k\}_{k\in\mathbb{Z}}$ le filtre de détails défini par (3.3) et $\{\tilde{h}_k\}_{k\in\mathbb{Z}}$ et $\{\tilde{g}_k\}_{k\in\mathbb{Z}}$ les filtres biorthogonaux associés.

Ces opérations seront effectuées en pratique grâce à la **transformée en ondelettes rapide** introduite par Mallat [60] en utilisant les relations à deux échelles :
- Décomposition : $c_{j,k} = \sum_l \tilde{h}_{l-2k}c_{j+1,l}$ et $d_{j,k} = \sum_l \tilde{g}_{l-2k}c_{j+1,l}$
- Reconstruction : $c_{j+1,k} = \sum_l h_{k-2l}c_{j,l} + \sum_l g_{k-2l}d_{j,l}$.

3.1.2 Base d'ondelettes à divergence nulle de $L^2(\mathbb{R}^2)^2$

Nous développons à présent des AMR en dimensions supérieures pour considérer des espaces à divergence nulle. La construction d'AMR peut se généraliser en dimension quelconque de façon simple, les ondelettes sont alors des produits tensoriels entre ondelette et fonction d'échelle ou ondelettes et ondelettes d'un même niveau, ce qui engendre des **AMR isotropes** ; si on considère des produits tensoriels d'ondelettes de différents niveaux, on constitue alors des bases d'ondelettes **anisotropes**. Nous allons ici décrire une AMR anisotrope en deux dimensions, les dimensions supérieures peuvent être déduites.

On considère deux AMRB de $L^2(\mathbb{R})$, V_j^0 et V_j^1, ainsi que leurs fonctions d'échelle et ondelettes respectives (φ^0, ψ^0) et (φ^1, ψ^1). Alors, une AMR de $L^2(\mathbb{R}^2)$ est donnée par

$$V_j = V_j^0 \otimes V_j^1$$

où $\{\varphi^0_{j,k_1}(t)\varphi^1_{j,k_2}(x);\ (k_1,k_2) \in \mathbb{Z}^2\}$ est une base de V_j. Les ondelettes anisotropes associées à cette AMR sont $\psi_{\mathbf{j},\mathbf{k}}(t,x) = \psi^0_{j_1,k_1}(t)\psi^1_{j_2,k_2}(x)$.

AMR de $L^2(\mathbb{R}^2)^2$

Une analyse multirésolution de $(L^2(\mathbb{R}^2))^2$ est formée par les espaces $\mathbf{V}_j = (V_j^1 \otimes V_j^0) \times (V_j^0 \otimes V_j^1)$ dont une base d'ondelettes anisotropes est formée par les ondelettes vecteurs

$$\Psi^1_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} \psi^1_{j_1,k_1}(t)\psi^0_{j_2,k_2}(x) \\ 0 \end{pmatrix} \tag{3.5}$$

et

$$\Psi^2_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} 0 \\ \psi^0_{j_1,k_1}(t)\psi^1_{j_2,k_2}(x) \end{pmatrix}, \tag{3.6}$$

Alors, si on considère le vecteur $\mathbf{u} = (u_1, u_2)$ appartenant à $(L^2(\mathbb{R}^2))^2$, il s'écrit

$$\mathbf{u}(t,x) = \sum_{\mathbf{j},\mathbf{k}} d^1_{\mathbf{j},\mathbf{k}}\Psi^1_{\mathbf{j},\mathbf{k}}(t,x) + d^2_{\mathbf{j},\mathbf{k}}\Psi^2_{\mathbf{j},\mathbf{k}}(t,x) \tag{3.7}$$

avec

$$u_1(t,x) = \sum_{\mathbf{j},\mathbf{k}} d^1_{\mathbf{j},\mathbf{k}}\psi^1(2^{j_1}t - k_1)\psi^0(2^{j_2}x - k_2)$$

$$u_2(t,x) = \sum_{\mathbf{j},\mathbf{k}} d^2_{\mathbf{j},\mathbf{k}}\psi^0(2^{j_1}t - k_1)\psi^1(2^{j_2}x - k_2)$$

Dérivation et ondelettes à divergence nulle

A partir d'une base d'ondelettes en deux dimensions, nous pouvons construire une base d'ondelettes à divergence nulle en choisissant judicieusement V_j^0 et V_j^1. Pour cela, il faut tout d'abord énoncer une proposition de Lemarié [58] qui permet de dériver des ondelettes et des fonctions d'échelle :

Proposition 5. *Soient $V_j^1, \tilde{V}_j^1$ une AMR biorthogonale de $L^2(\mathbb{R})$, et (φ^1, ψ^1) et $(\tilde{\varphi}^1, \tilde{\psi}^1)$ leurs fonctions d'échelle et ondelettes associées telles que φ^1, ψ^1 soient $C^{1+\varepsilon}$ pour un $\varepsilon > 0$. Alors il existe une AMR biorthogonale associée $(V_j^0, \tilde{V}_j^0)$, avec des fonctions d'échelle et ondelettes biorthogonales (φ^0, ψ^0) et $(\tilde{\varphi^0}, \tilde{\psi}^0)$ telles que*

$$(\varphi^1)'(x) = \varphi^0(x) - \varphi^0(x-1) \text{ et } \psi^0(x) = \frac{1}{4}(\psi^1)'(x).$$

De plus, les fonctions duales vérifient

$$\int_x^{x+1} \tilde{\varphi}^1(t)dt = \tilde{\varphi}^0(x) \text{ et } (\tilde{\psi^0})'(x) = -4\tilde{\psi}^1(x)$$

Ondelettes périodiques anisotropes à divergence nulle

Dans le cas des ondelettes à divergence nulle anisotropes bidimensionnelles, il n'y a qu'une seule ondelette génératrice sur laquelle agissent 2 facteurs de dilatation (différents en t et x). Nous allons considérer

$$\Psi^{div}_{\mathbf{j},\mathbf{k}}(t,x) := \frac{1}{\sqrt{4^{j_1}+4^{j_2}}} \left| \begin{array}{c} 2^{j_2}\psi^1_{j_1,k_1}(t)\psi^0_{j_2,k_2}(x) \\ -2^{j_1}\psi^0_{j_1,k_1}(t)\psi^1_{j_2,k_2}(x) \end{array} \right. \tag{3.8}$$

qui est à divergence nulle d'après la proposition 5. Pour compléter la base de $L^2(\mathbb{R}^2)$, le choix de l'ondelette complémentaire le plus judicieux est :

$$\Psi^{an}_{\mathbf{j},\mathbf{k}}(t,x) = \left| \begin{array}{c} 2^{j_1}\psi^1_{j_1,k_1}(t)\psi^0_{j_2,k_2}(x) \\ 2^{j_2}\psi^0_{j_1,k_1}(t)\psi^1_{j_2,k_2}(x) \end{array} \right. ,$$

car pour $\mathbf{j}$ et $\mathbf{k}$ fixés, la fonction $\Psi^{an}_{\mathbf{j},\mathbf{k}}$ est orthogonale à $\Psi^{div}_{\mathbf{j},\mathbf{k}}$.

On obtient donc d'après (3.7) une seconde écriture de $\mathbf{u} \in L^2(\mathbb{R}^2)$:

$$\mathbf{u}(t,x) = \sum_{\mathbf{j},\mathbf{k}\in\mathbb{Z}^2} d^{div}_{\mathbf{j},\mathbf{k}}\Psi^{div}_{\mathbf{j},\mathbf{k}}(t,x) + d^{an}_{\mathbf{j},\mathbf{k}}\Psi^{an}_{\mathbf{j},\mathbf{k}}(t,x). \tag{3.9}$$

Dans le cas particulier où le vecteur $\mathbf{u}$ est à divergence nulle, il s'écrit de manière unique sous la forme :

$$\mathbf{u} = \sum_{\mathbf{j},\mathbf{k}\in\mathbb{Z}^2} \langle \mathbf{u}/\tilde{\Psi}^{div}_{\mathbf{j},\mathbf{k}}\rangle \Psi^{div}_{\mathbf{j},\mathbf{k}} = \sum_{\mathbf{j},\mathbf{k}\in\mathbb{Z}^2} d^{div}_{\mathbf{j},\mathbf{k}}\Psi^{div}_{\mathbf{j},\mathbf{k}}. \tag{3.10}$$

où l'ondelette définie par :

$$\tilde{\Psi}^{div}_{\mathbf{j},\mathbf{k}}(t,x) := \frac{1}{\sqrt{4^{j_1}+4^{j_2}}} \left| \begin{array}{c} 2^{j_2}\tilde{\psi}^1_{j_1,k_1}(t)\tilde{\psi}^0_{j_1,k_1}(x) \\ -2^{j_1}\tilde{\psi}^0_{j_1,k_1}(t)\tilde{\psi}^1_{j_1,k_1}(x) \end{array} \right.$$

engendre une famille biorthogonale pour le produit scalaire de $(L^2(\mathbb{R}^2))^2$ aux ondelettes $\Psi^{div}_{\mathbf{j},\mathbf{k}}$.

Remarque. Les ondelettes de l'analyse multirésolution biorthogonale $(\tilde{V}^1_j \otimes \tilde{V}^0_j) \times (\tilde{V}^0_j \otimes \tilde{V}^1_j)$ sont à rotationnel nul. En effet,

$$\tilde{\Psi}(t,x) := \left| \begin{array}{c} \tilde{\psi}^1(t)\tilde{\psi}^0(x) \\ \tilde{\psi}^0(t)\tilde{\psi}^1(x) \end{array} \right.$$

et comme

$$\frac{d}{dt}\tilde{\psi}^0(t) = -4\tilde{\psi}^1(t),$$

on a

$$\nabla \times \tilde{\Psi}(t,x) = \frac{\partial}{\partial t}[\tilde{\psi}^0(t)\tilde{\psi}^1(x)] - \frac{\partial}{\partial x}[\tilde{\psi}^1(t)\tilde{\psi}^0(x)] = 0$$

Algorithmes de décomposition et reconstruction

Ayant les deux décompositions (3.7) et (3.9) du vecteur $\mathbf{u}$, les coefficients d'ondelettes d^{div} et d^{an} sont reliés aux coefficients d^1 et d^2 par :

— Décomposition :

$$d^{div}_{\mathbf{j},\mathbf{k}} = \frac{2^{j_2}}{\sqrt{4^{j_1}+4^{j_2}}} d^1_{\mathbf{j},\mathbf{k}} - \frac{2^{j_1}}{\sqrt{4^{j_1}+4^{j_2}}} d^2_{\mathbf{j},\mathbf{k}}$$

$$d^{an}_{\mathbf{j},\mathbf{k}} = \frac{2^{j_1}}{\sqrt{4^{j_1}+4^{j_2}}} d^1_{\mathbf{j},\mathbf{k}} + \frac{2^{j_2}}{\sqrt{4^{j_1}+4^{j_2}}} d^2_{\mathbf{j},\mathbf{k}}$$

— Recomposition :

$$d^1_{\mathbf{j},\mathbf{k}} = \frac{2^{j_2}}{\sqrt{4^{j_1}+4^{j_2}}} d^{div}_{\mathbf{j},\mathbf{k}} + \frac{2^{j_1}}{\sqrt{4^{j_1}+4^{j_2}}} d^{an}_{\mathbf{j},\mathbf{k}}$$

$$d^2_{\mathbf{j},\mathbf{k}} = -\frac{2^{j_1}}{\sqrt{4^{j_1}+4^{j_2}}} d^{div}_{\mathbf{j},\mathbf{k}} + \frac{2^{j_2}}{\sqrt{4^{j_1}+4^{j_2}}} d^{an}_{\mathbf{j},\mathbf{k}}$$

3.1.3 Base d'ondelettes à divergence nulle de $L^2((0,1)^2)^2$

Ondelettes périodiques

On construit à présent des ondelettes périodiques sur $(0,1)$ à partir des ondelettes définies sur $\mathbb{R}$ à l'aide de l'équation suivante :

$$\text{soit } \psi \in L^2(\mathbb{R}),\ \psi^{per}(t) = \sum_{n=-\infty}^{+\infty} \psi(t+n).$$

Les bases d'ondelettes périodiques sont obtenues par périodisation de la base d'ondelettes sur $\mathbb{R}$

$$\psi^{per}_{j,k}(t) = 2^{j/2} \sum_{n\in\mathbb{Z}} \psi\left(2^j(t+n)-k\right).$$

Dans ce cas la seule fonction d'échelle est $\varphi^{per} = 1$ (car $\sum_n \varphi(x+n) = 1$ pour toute fonction d'échelle φ) et $(\varphi^{per}, (\psi_{j,k})_{j,k})$ est une base d'ondelettes de $L^2(0,1)$. Le coefficient $\langle f, \varphi^{per}\rangle$ est alors la moyenne de f sur $(0,1)$. Cette périodisation entraîne une modification des ondelettes $\psi_{j,k}$ uniquement si leur support contient 0 ou 1. Nous noterons dans la suite les ondelettes périodiques $\psi_{j,k}$ et non $\psi^{per}_{j,k}$.

On obtient de la même manière que précédemment les ondelettes à divergence nulle. On considère l'AMR de $(L^2(0,1))^2$ formée par les espaces $(V^1_j \otimes V^0_j) \times (V^0_j \otimes V^1_j)$ dont une base vectorielle est formée par les ondelettes vecteurs

$$\Psi^{1,0}_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} 1(t)1(x) \\ 0 \end{pmatrix} \quad \Psi^{1,1}_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} 1(t)\psi^0_{j_2,k_2}(x) \\ 0 \end{pmatrix} \tag{3.11}$$

$$\Psi^{1,2}_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} \psi^1_{j_1,k_1}(t)1(x) \\ 0 \end{pmatrix} \quad \Psi^{1,3}_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} \psi^1_{j_1,k_1}(t)\psi^0_{j_2,k_2}(x) \\ 0 \end{pmatrix}$$

et

$$\Psi^{2,0}_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} 0 \\ 1(t)1(x) \end{pmatrix} \quad \Psi^{2,1}_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} 0 \\ 1(t)\psi^1_{j_2,k_2}(x) \end{pmatrix} \tag{3.12}$$

$$\Psi^{2,2}_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} 0 \\ \psi^0_{j_1,k_1}(t)1(x) \end{pmatrix} \quad \Psi^{2,3}_{\mathbf{j},\mathbf{k}}(t,x) = \begin{pmatrix} 0 \\ \psi^0_{j_1,k_1}(t)\psi^1_{j_2,k_2}(x) \end{pmatrix}.$$

On peut alors écrire le vecteur $\mathbf{u} = (u_1, u_2) \in (L^2(0,1))^2$

$$\mathbf{u}(t,x) = \sum_{\epsilon=0}^{3} \sum_{\mathbf{j},\mathbf{k}} \left(d^{1,\epsilon}_{\mathbf{j},\mathbf{k}} \Psi^{1,\epsilon}_{\mathbf{j},\mathbf{k}}(t,x) + d^{2,\epsilon}_{\mathbf{j},\mathbf{k}} \Psi^{2,\epsilon}_{\mathbf{j},\mathbf{k}}(t,x) \right)$$

et dans le cas $\mathbf{u} \in (L^2(0,1))^2$ à divergence nulle et périodique on peut de plus écrire

$$\mathbf{u} = \sum_{\epsilon=1}^{3} \sum_{\mathbf{j},\mathbf{k}} d^{div,\epsilon}_{\mathbf{j},\mathbf{k}} \Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}} = \sum_{\epsilon=1}^{3} \sum_{\mathbf{j},\mathbf{k}} d^{div,\epsilon}_{\mathbf{j},\mathbf{k}} \begin{pmatrix} (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 \\ (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 \end{pmatrix},$$

où

$$\begin{aligned}
\Psi^{div,1}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_2}+1}} \nabla \times (1 \otimes \psi^1_{j_2,k_2}) = \frac{1}{\sqrt{4^{j_2}+1}} \begin{pmatrix} 1(t)(\psi^1_{j_2,k_2})'(x) \\ 0 \end{pmatrix} \\
\Psi^{div,2}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_1}+1}} \nabla \times (\psi^1_{j_1,k_1} \otimes 1) = \frac{1}{\sqrt{4^{j_1}+1}} \begin{pmatrix} 0 \\ -(\psi^1_{j_1,k_1})'(t)1(x) \end{pmatrix} \\
\Psi^{div,3}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_1}+4^{j_2}}} \nabla \times (\psi^1_{j_1,k_1} \otimes \psi^1_{j_2,k_2}) = \frac{1}{\sqrt{4^{j_1}+4^{j_2}}} \begin{pmatrix} \psi^1_{j_1,k_1}(t)(\psi^1_{j_2,k_2})'(x) \\ -(\psi^1_{j_1,k_1})'(t)\psi^1_{j_2,k_2}(x) \end{pmatrix},
\end{aligned} \tag{3.13}$$

En associant aux ondelettes $(\Psi^{div,1}_{\mathbf{j},\mathbf{k}}, \Psi^{div,2}_{\mathbf{j},\mathbf{k}}, \Psi^{div,3}_{\mathbf{j},\mathbf{k}})_{\mathbf{j},\mathbf{k}}$ les fonctions d'échelle et ondelettes complémentaires

$$\Phi^1_{\mathbf{j},\mathbf{k}}(t,x) = \frac{1}{\sqrt{1+1}} \begin{pmatrix} 1(t)1(x) \\ 0 \end{pmatrix} \quad \Phi^2_{\mathbf{j},\mathbf{k}}(t,x) = \frac{1}{\sqrt{1+1}} \begin{pmatrix} 0 \\ 1(t)1(x) \end{pmatrix}$$

$$\Psi^{an,1}_{\mathbf{j},\mathbf{k}}(t,x) = \frac{1}{\sqrt{4^{j_1}+1}} \begin{pmatrix} \psi^1_{j_1,k_1}(t)1(x) \\ 0 \end{pmatrix} \quad \Psi^{an,2}_{\mathbf{j},\mathbf{k}}(t,x) = \frac{1}{\sqrt{1+4^{j_2}}} \begin{pmatrix} 0 \\ 1(t)\psi^1_{j_2,k_2}(x) \end{pmatrix}$$

et

$$\Psi^{an,3}_{\mathbf{j},\mathbf{k}}(t,x) = \frac{1}{\sqrt{4^{j_1}+4^{j_2}}} \begin{pmatrix} 2^{j_1}\psi^1_{j_1,k_1}(t)\psi^0_{j_2,k_2}(x) \\ 2^{j_2}\psi^0_{j_1,k_1}(t)\psi^1_{j_2,k_2}(x) \end{pmatrix}$$

on obtient une autre base de $(L^2(0,1))^2$.

Algorithmes de décomposition et reconstruction

Pour un vecteur $\mathbf{u}$ à divergence nulle, les coefficients d'ondelettes $\mathbf{d}^{div,\varepsilon}$ sont reliés aux coefficients $\mathbf{d}^{1,\varepsilon}$ et $\mathbf{d}^{2,\varepsilon}$ par :

— Décomposition :

$$d^{div,1}_{\mathbf{j},\mathbf{k}} = \frac{2^{j_2}}{\sqrt{1+4^{j_2}}} d^{1,1}_{\mathbf{j},\mathbf{k}}$$

$$d^{div,2}_{\mathbf{j},\mathbf{k}} = -\frac{2^{j_1}}{\sqrt{4^{j_1}+1}} d^{2,2}$$

$$d^{div,3}_{\mathbf{j},\mathbf{k}} = \frac{2^{j_2}}{\sqrt{4^{j_1}+4^{j_2}}} d^{1,3}_{\mathbf{j},\mathbf{k}} - \frac{2^{j_1}}{\sqrt{4^{j_1}+4^{j_2}}} d^{2,3}_{\mathbf{j},\mathbf{k}}$$

— Recomposition :

$$d^{1,1}_{\mathbf{j},\mathbf{k}} = \frac{2^{j_2}}{\sqrt{1+4^{j_2}}} d^{div,1}_{\mathbf{j},\mathbf{k}} \qquad d^{2,2}_{\mathbf{j},\mathbf{k}} = -\frac{2^{j_1}}{\sqrt{4^{j_1}+1}} d^{div,2}_{\mathbf{j},\mathbf{k}}$$

$$d^{1,3}_{\mathbf{j},\mathbf{k}} = \frac{2^{j_2}}{\sqrt{4^{j_1}+4^{j_2}}} d^{div,3}_{\mathbf{j},\mathbf{k}} \qquad d^{2,3}_{\mathbf{j},\mathbf{k}} = -\frac{2^{j_1}}{\sqrt{4^{j_1}+4^{j_2}}} d^{div,3}_{\mathbf{j},\mathbf{k}}$$

Ondelettes prenant en compte les conditions sur le bord

Nous détaillons à présent les bases d'ondelettes permettant la prise en compte de conditions de Dirichlet homogènes sur le bord. Cette section est un résumé de la construction des ondelettes à divergence nulle sur le carré $[0,1]$ développée dans [51, 50].

AMR de $L^2(0,1)$

Jouini et Lemarié [49] ont prouvé l'existence de deux AMRB de $L^2(0,1)$, V_j^1 et V_j^0, reliées par dérivation, dérivées de celles définies dans la proposition 5, telles que

$$\frac{d}{dx} V_j^1 = V_j^0 \text{ et } \tilde{V}_j^0 = H_0^1(0,1) \cap \int_0^x \tilde{V}_j^1 = \{f : f' \in \tilde{V}_j^1 \text{ et } f(0) = f(1) = 0\}. \tag{3.14}$$

L'AMRB V_j^1 reproduit les polynômes jusqu'au degré $r-1$ et est construite à partir d'une fonction d'échelle φ^1 de support $[n_{min}, n_{max}]$. Sa structure est la suivante :

$$V_j^1 = V_j^{1,b} \oplus V_j^{1,int} \oplus V_j^{1,\sharp},$$

où $V_j^{1,b}$ est de dimension r et contient les fonctions d'échelle permettant la reproduction des polynômes sur le bord 0 et $V_j^{1,\sharp}$ sur le bord 1. L'espace $V_j^{1,int}$ contient les $\varphi_{j,k} = 2^{j/2}\varphi(x-k)$ dont le support est inclus dans $[0,1]$. Il faut alors choisir $j \geq j_{min}$ pour que les supports des fonctions d'échelle de bord en 0 et 1 ne s'intersectent pas. L'espace $\tilde{V}_j$ a la même structure et reproduit les polynômes jusqu'au degré $\tilde{r}-1$ et on pose :

$$\Delta_j = \dim(V_j^1) = \dim(\tilde{V}_j^1).$$

On écrira par simplicité

$$V_j^1 = \text{span}\{\varphi_{j,k}^1;\ 0 \le k \le \Delta_j - 1\} \text{ et } \tilde{V}_j^1 = \text{span}\{\tilde{\varphi}_{j,k}^1;\ 0 \le k \le \Delta_j - 1\}.$$

Si on veut à présent imposer des conditions Dirichlet homogènes sur $(V_j^1, \tilde{V}_j^1)$, il suffit d'enlever de leurs bases les deux fonctions d'échelle qui ne s'annulent pas aux bords 0 et 1. Après biorthogonalisation on obtient une AMRB de $H_0^1(0,1)$, [61] :

$$V_j^d = V_j^1 \cap H_0^1(0,1) = span\{\varphi_{j,k}^1;\ 1 \le k \le \Delta_j - 2\},$$
$$\tilde{V_j^d} = \tilde{V}_j^1 \cap H_0^1(0,1) = span\{\tilde{\varphi}_{j,k}^1;\ 1 \le k \le \Delta_j - 2\}.$$

Les espaces $(V_j^0, \tilde{V}_j^0)$ sont reliés aux $(V_j^1, \tilde{V}_j^1)$ par les relations (3.14). Leur construction est similaire à celle de $(V_j^1, \tilde{V}_j^1)$; cependant, en raison des relations explicitées dans la proposition 5, V_j^0 reproduit les polynômes jusqu'au degré $r-2$ et φ^0 a pour support $[n_{min}, n_{max} - 1]$, alors que $\tilde{V}_j^0$ reproduit les polynômes jusqu'au degré $\tilde{r}$ et $\tilde{\varphi}^0$ a pour support $[\tilde{n}_{min} - 1, \tilde{n}_{max}]$. On a de plus $\dim(V_j^0) = \dim(\tilde{V}_j^0) = \Delta_j - 1$ et

$$V_j^0 = \text{span}\{\varphi_{j,k}^0;\ 0 \le k \le \Delta_j - 2\} \text{ et } \tilde{V}_j^0 = \text{span}\{\tilde{\varphi}_{j,k}^0;\ 0 \le k \le \Delta_j - 2\}.$$

Les **espaces d'ondelettes biorthogonaux** $(W_j^1, \tilde{W}_j^1)$ sont définis comme sur $\mathbb{R}$ par $W_j^1 = V_{j+1}^1 \cap (\tilde{V}_j^1)^\perp$ et $\tilde{W}_j^1 = \tilde{V}_{j+1}^1 \cap (V_j^1)^\perp$ et on obtient finalement les bases

$$(\psi_{j,k}^1)_{k=1,2^j} \text{ et } (\tilde{\psi}_{j,k}^1)_{k=1,2^j}$$

Les espaces $(W_j^0, \tilde{W}_j^0)$ sont obtenus à partir des espaces $(W_j^1, \tilde{W}_j^1)$ par dérivation.

Ondelettes anisotropes à divergence nulle sur l'intervalle

On considère à nouveau l'espace $Q = (0,1)^2$ introduit dans le premier chapitre, ainsi que l'espace $V(Q) = \{u \in (L^2(Q))^2,\ \text{div}(u) = 0\}$. Comme on le verra plus en détail dans la prochaine section, on peut écrire

$$\begin{aligned} H^{\text{div}} &= V(Q) \cap \{u,\ u \cdot \nu_Q = 0\} \\ &= \{u \in (L^2(Q))^2;\ \text{div}(u) = 0,\ u \cdot \nu_Q = 0\} \\ &= \{u = \nabla \times \varphi,\ \varphi \in H_0^1(Q)\}. \end{aligned}$$

Nous allons donc considérer l'AMR de $H_0^1(Q)$, $(V_j^d \otimes V_j^d)$, et prendre le rotationnel des fonctions d'échelle qui l'engendrent :

$$\Phi_{j,\mathbf{k}}^{div} = \frac{1}{\sqrt{2}} \nabla \times (\varphi_{j,k_1}^d \otimes \varphi_{j,k_2}^d) = \frac{1}{\sqrt{2}} \begin{pmatrix} \varphi_{j,k_1}^d \otimes (\varphi_{j,k_2}^d)' \\ -(\varphi_{j,k_1}^d)' \otimes \varphi_{j,k_2}^d \end{pmatrix},\ 1 \le k_1, k_2 \le \Delta_j - 2.$$

Ce que nous dénoterons par $\mathbf{V}_j^{div} = \nabla \times (V_j^d \otimes V_j^d)$ est l'espace des fonctions d'échelle à divergence nulle de dimension $(\Delta_j - 2)^2$ et il a été démontré dans [51] que $(\mathbf{V}_j^{div})_{j \geq j_{min}}$ est une AMR de H^{div}. Les ondelettes associées à ces AMR sont également définies de façon similaire à ce qui a été fait dans le chapitre précédent.

Définition 2. *Pour* $\mathbf{j} = (j_1, j_2) \in \{j_{min}, ...\}^2$*, les ondelettes à divergence nulle anisotropes sont définies par :*

$$\Psi_{\boldsymbol{j},\boldsymbol{k}}^{div,1} := \frac{1}{\sqrt{4^{j_2}+1}} \nabla \times (\varphi_{j_{min},k_1}^d \otimes \psi_{j_2,k_2}^d)$$
$$\Psi_{\boldsymbol{j},\boldsymbol{k}}^{div,2} := \frac{1}{\sqrt{4^{j_1}+1}} \nabla \times (\psi_{j_1,k_1}^d \otimes \varphi_{j_{min},k_2}^d)$$
$$\Psi_{\boldsymbol{j},\boldsymbol{k}}^{div,3} := \frac{1}{\sqrt{4^{j_1}+4^{j_2}}} \nabla \times (\psi_{j_1,k_1}^d \otimes \psi_{j_2,k_2}^d)$$

Il est alors possible de décomposer toute fonction $\mathbf{u} \in H^{\text{div}}$ de façon unique par :

$$\mathbf{u} = \sum_{\mathbf{k}} c_{j_{min},\mathbf{k}}^{div} \Phi_{j_{min},\mathbf{k}}^{div} + \sum_{\epsilon} \sum_{\mathbf{j},\mathbf{k}} d_{\mathbf{j},\mathbf{k}}^{div,\epsilon} \Psi_{\mathbf{j},\mathbf{k}}^{div,\epsilon},$$

ou, en notant l'ensemble des ondelettes et fonction d'échelle de façon simplifiée : $\Psi_{\mathbf{j},\mathbf{k}}^{div,0} = \Phi_{j,\mathbf{k}}^{div}$, on obtient $H^{\text{div}} = span\{\Psi_{\mathbf{j},\mathbf{k}}^{div,0}, \Psi_{\mathbf{j},\mathbf{k}}^{div,1}, \Psi_{\mathbf{j},\mathbf{k}}^{div,2}, \Psi_{\mathbf{j},\mathbf{k}}^{div,3}\} := span\{\Psi_{\mathbf{j},\mathbf{k}}^{div}\}$

$$\boxed{\mathbf{u} = \sum_{\epsilon} \sum_{\mathbf{j},\mathbf{k}} d_{\mathbf{j},\mathbf{k}}^{div,\epsilon} \Psi_{\mathbf{j},\mathbf{k}}^{div,\epsilon} := \sum_{\mathbf{j},\mathbf{k}} d_{\mathbf{j},\mathbf{k}}^{div} \Psi_{\mathbf{j},\mathbf{k}}^{div}.} \tag{3.15}$$

Calcul des coefficients $d_{\mathbf{j},\mathbf{k}}^{div}$

En multipliant (3.15) par $\Psi_{\mathbf{j},\mathbf{k}}^{div}$, on obtient :

$$\langle \mathbf{u}, \Psi_{\mathbf{j},\mathbf{k}}^{div} \rangle = M_{div}(d_{\mathbf{j},\mathbf{k}}^{div})$$

où M_{div} est la matrice de Gram (parcimonieuse) de la base $\{\Psi_{\mathbf{j},\mathbf{k}}^{div}\}$:

$$[M_{div}]_{k,k'} = \langle \Psi_{\mathbf{j},\mathbf{k}}^{div}, \Psi_{\mathbf{j},\mathbf{k}'}^{div} \rangle.$$

Afin de calculer la matrice de Gram de la base Ψ^{div} à partir de celle de la base ψ^d, on calcule :

$$\begin{aligned}\langle \Psi_{\mathbf{j},\mathbf{k}}^{div}, \Psi_{\mathbf{j}',\mathbf{k}'}^{div} \rangle &= \langle \psi_{j_1,k_1}^d \otimes (\psi_{j_2,k_2}^d)', \psi_{j_1',k_1'}^d \otimes (\psi_{j_2',k_2'}^d)' \rangle + \langle -(\psi_{j_1,k_1}^d)' \otimes \psi_{j_2,k_2}^d, -(\psi_{j_1',k_1'}^d)' \otimes \psi_{j_2',k_2'}^d \rangle \\ &= [M]_{k_1,k_1'}^{j_1,j_1'} \cdot [R]_{k_2,k_2'}^{j_2,j_2'} + [R]_{k_1,k_1'}^{j_1,j_1'} \cdot [M]_{k_2,k_2'}^{j_2,j_2'},\end{aligned}$$

où

$$[R]_{k,k'}^{j,j'} = \langle (\psi_{j,k}^d)', (\psi_{j',k'}^d)' \rangle$$
$$[M]_{k,k'}^{j,j'} = \langle \psi_{j,k}^d, \psi_{j',k'}^d \rangle.$$

On peut ainsi écrire

$$[M_{div}]_{\mathbf{k},\mathbf{k}'} = [M]_{k_1,k_1'} \otimes [R]_{k_2,k_2'} + [R]_{k_1,k_1'} \otimes [M]_{k_2,k_2'},$$

$$M_{div} d^{div}_{\mathbf{j},\mathbf{k}} = [M]^{j_1,j_1'}_{k_1,k_1'} [d^{div}_{\mathbf{j},\mathbf{k}}] [R]^{j_2,j_2'}_{k_2,k_2'} + [R]^{j_1,j_1'}_{k_1,k_1'} [d^{div}_{\mathbf{j},\mathbf{k}}] [M]^{j_2,j_2'}_{k_2,k_2'}.$$

Comme dans le cas des ondelettes périodiques (équation (3.7)), pour calculer les coefficients d^{div}, il faut tout d'abord décomposer u dans la base d'ondelettes classique de $(V_j^d \otimes V_j^0) \times (V_j^0 \otimes V_j^d)$

$$u_1 = \sum_{\mathbf{j},\mathbf{k}} d^1_{\mathbf{j},\mathbf{k}} \psi^d_{j_1,k_1} \otimes \psi^0_{j_2,k_2} \text{ et } u_2 = \sum_{\mathbf{j},\mathbf{k}} d^2_{\mathbf{j},\mathbf{k}} \psi^0_{j_1,k_1} \otimes \psi^d_{j_2,k_2}.$$

En multipliant par l'ondelette à divergence nulle on obtient :

$$\langle \mathbf{u}, \Psi^{div}_{\mathbf{j}',\mathbf{k}'} \rangle = \sum_{\mathbf{j},\mathbf{k}} d^1_{\mathbf{j},\mathbf{k}} \langle \psi^d_{j_1,k_1} \otimes \psi^0_{j_2,k_2}, \psi^d_{j_1',k_1'} \otimes (\psi^d_{j_2',k_2'})' \rangle + \sum_{\mathbf{j},\mathbf{k}} d^2_{\mathbf{j},\mathbf{k}} \langle \psi^0_{j_1,k_1} \otimes \psi^d_{j_2,k_2}, -(\psi^d_{j_1',k_1'})' \otimes \psi^d_{j_2',k_2'} \rangle$$

On obtient donc comme relation entre les coefficients d^1, d^2 et d^{div} :

$$M_{div} \mathbf{d}^{div} = [\langle \mathbf{u}, \Psi^{div}_{\mathbf{j}',\mathbf{k}'} \rangle] = M[d^1_{\mathbf{j},\mathbf{k}}] C^0 - (C^0)^t [d^2_{\mathbf{j},\mathbf{k}}] M,$$

où C^0 est la matrice de Gram des éléments $\langle \psi^0_{j,k}, (\psi^d_{j',k'})' \rangle$.

3.1.4 Formulation du problème de transport optimal sur bases d'ondelettes à divergence nulle

Nous ne pouvons pas utiliser cette base et ces algorithmes de décomposition et recomposition sur le couple (ρ, m) car il ne vérifie pas $(\rho, m) \cdot \nu_Q = 0$. Nous allons donc définir un nouveau couple nul sur le bord et à divergence nulle contenant les informations de (ρ, m).

Définition des variables en domaine confiné

Pour utiliser la décomposition en ondelettes il faut définir un nouveau couple $(\tilde{\rho}, \tilde{m}) \in H^{\text{div}}$ qui vérifie la condition de paroi, classique en mécanique des fluides :

$$\begin{cases} \text{div } (\tilde{\rho}, \tilde{m}) = 0 \\ (\tilde{\rho}, \tilde{m}) \cdot \nu_Q = 0 \end{cases}.$$

On choisit donc

$$\begin{cases} \tilde{\rho}(t,x) &= \rho(t,x) - (t\rho_1(x) + (1-t)\rho_0(x)) \\ \tilde{m}(t,x) &= m(t,x) + \int_0^x (\rho_1(y) - \rho_0(y)) dy \end{cases}.$$

En effet,

$$\begin{aligned} div_{t,x}(\tilde{\rho},\tilde{m}) &= \partial_t\rho - \rho_1(x) + \rho_0(x) + \partial_x m + \partial_x\left(\int_0^x \rho_1(y) - \rho_0(y)dy\right) \\ &= \partial_t\rho + \partial_x m = 0 \end{aligned}$$

et

$\tilde{\rho}(0,x) = \rho(0,x) - \rho_0(x) = 0$ et $\tilde{\rho}(1,x) = \rho(1,x) - \rho_1(x) = 0$

$\tilde{m}(t,0) = m(t,0) + \int_0^0 \rho_1(y) - \rho_0(y) = m(t,0)$ et $\tilde{m}(t,1) = m(t,1) + \int_0^1 \rho_1(y) - \rho_0(y) = m(t,1)$.

De plus, en intégrant l'équation vérifiée par la divergence, on observe que la moyenne spaciale de $\tilde{\rho}$ et la moyenne temporelle de $\tilde{m}$ sont nulles :

$$\begin{aligned} 0 = \int_0^s \left(\int_0^1 div_{t,x}(\tilde{\rho},\tilde{m})dx\right)dt &= \int_0^s \left(\int_0^1 (\partial_t\tilde{\rho} + \partial_x\tilde{m})\,dx\right)dt \\ &= \int_0^1 \left(\int_0^s \partial_t\tilde{\rho}\;dt\right)dx + \int_0^s \left(\int_0^1 \partial_x\tilde{m}\;dx\right)dt \\ &= \int_0^1 \tilde{\rho}dx + \int_0^1 \tilde{\rho}(0,x)dx + \int_0^s (\tilde{m}(t,1) - \tilde{m}(t,0))\,dt = \int_0^1 \tilde{\rho}dx. \end{aligned}$$

car $\forall t\ \tilde{m}(t,0) = \tilde{m}(t,1) = 0$, et

$$\begin{aligned} 0 = \int_0^z \left(\int_0^1 div_{t,x}(\tilde{\rho},\tilde{m})dt\right)dx &= \int_0^z \left(\int_0^1 (\partial_t\tilde{\rho} + \partial_x\tilde{m})\,dt\right)dx \\ &= \int_0^z \left(\int_0^1 \partial_t\tilde{\rho}\;dt\right)dx + \int_0^1 \left(\int_0^z \partial_x\tilde{m}\;dx\right)dt \\ &= \int_0^z \tilde{\rho}(1,x) - \tilde{\rho}(0,x)dx + \int_0^1 (\tilde{m} - \tilde{m}(t,0))\,dt = \int_0^1 \tilde{m}dt. \end{aligned}$$

Alors, à J donné, le couple $(\tilde{\rho},\tilde{m})$ est calculé sur la base des ondelettes vecteurs $\left(\psi_{j,k}^{div,\epsilon}\right)_{j\leq J-1}$ sous la forme :

$$\begin{pmatrix}\tilde{\rho}(t,x)\\ \tilde{m}(t,x)\end{pmatrix} = \sum_{j_{min},\mathbf{k}} c_{j_{min},\mathbf{k}}^{div}\Phi_{j_{min},\mathbf{k}}^{div}(t,x) + \sum_{\epsilon}\sum_{\mathbf{j},\mathbf{k}} d_{\mathbf{j},\mathbf{k}}^{div,\epsilon}\Psi_{\mathbf{j},\mathbf{k}}^{div,\epsilon}(t,x).$$

Dans ce cas la nouvelle énergie à minimiser est

$$\boxed{\mathcal{J}_w(\mathbf{d}^{div}) = \int_0^1\int_0^1 \frac{(\tilde{m}(\mathbf{d}^{div}) - \int_0^x(\rho_1(x) - \rho_0(x))dy)^2}{\tilde{\rho}(\mathbf{d}^{div}) + t\rho_1(x) + (1-t)\rho_0(x)}dxdt.} \tag{3.16}$$

$\mathcal{J}_w$ est **convexe** comme composée d'une fonction convexe et d'une application linéaire et est différentiable car $\mathcal{J}$ est différentiable et la décomposition sur les ondelettes est linéaire.

3.2 Fonction de courant

Pour travailler dans
$C = \{(\rho, m);\ \mathrm{div}_{t,x}(\rho, m) = 0,\ m(.,x).\nu_\Omega = 0,\ \forall x \in \partial\Omega,\ \rho(0,.) = \rho_0,\ \rho(1,.) = \rho_1\}$, nous utilisons à présent la représentation des champs de vecteurs à divergence nulle par leur fonction de courant. En effet, pour $n = 1$ ou $n = 2$, nous travaillons dans $V(Q) = \{f \in (L^2(Q))^{1+n},\ \mathrm{div}_{t,x} f = 0\}$, où $Q = (0,1) \times \Omega$. Comme prouvé dans [38], si $v \in V(Q)$ vérifie $(v \cdot \nu_Q)|_{\partial Q} = 0$ où ν_Q est la normale sortante de Q, alors il existe une unique fonction $\phi \in (H^1(Q))^{n'}$ (n' étant égal à 1 si $n = 1$ ou égal à 3 si $n = 2$) telle que

$$\boxed{\begin{cases} v &= \nabla \times \phi \\ \phi &= 0 \text{ sur } \partial Q, \end{cases}}$$

où ϕ est appelée **la fonction de courant** et on note $\nabla = \nabla_{t,x}$ dans la suite. De plus, si $n = 2$, ϕ est à divergence nulle.

3.2.1 Transport optimal 1D

Proposition 6. *Dans le cas $Q = (0,1) \times (0,1)$, c'est-à-dire en dimension un en espace, pour $(\rho, m) \in V(Q)$ tel que $m(.,x).\nu_\Omega = 0,\ \forall x \in \partial\Omega,$ et $\rho(0,.) = \rho_0,\ \rho(1,.) = \rho_1$, il existe $\phi \in H^1(Q)$ tel que :*

$$(\rho, m) = \nabla \times \phi = \begin{pmatrix} \partial_x \phi \\ -\partial_t \phi \end{pmatrix}.$$

En notant $R_0(x) = \int_0^x \rho_0(s)ds$ et $R_1(x) = \int_0^x \rho_1(s)ds$, on obtient les conditions sur le bord :

$$\begin{cases} \phi(0,x) = R_0(x) \\ \phi(1,x) = R_1(x) \\ \phi(t,0) = R_1(0) = R_0(0) = 0 \\ \phi(t,1) = R_1(1) = R_0(1) = 1 \end{cases}. \tag{3.17}$$

Démonstration. On ne peut pas appliquer directement la représentation par fonction de courant à notre couple (ρ, m) car il ne vérifie pas $((\rho, m) \cdot \nu_Q)|_{\partial Q} = 0$. Nous allons donc comme pour l'usage des ondelettes, définir le couple $(\bar\rho, \bar m)$ (section 3.1.4) :

$$\begin{cases} \bar\rho(t,x) &= \rho(t,x) - (t\rho_1(x) + (1-t)\rho_0(x)) \\ \bar m(t,x) &= m(t,x) + \int_0^x (\rho_1(y) - \rho_0(y))dy = m(t,x) + R_1(x) - R_0(x) \end{cases}$$

qui vérifie,

$$\begin{cases} (\bar\rho, \bar m) \cdot \nu_Q &= 0 \\ \mathrm{div}_{t,x}(\bar\rho, \bar m) &= 0 \end{cases}$$

Il existe donc $\bar\phi \in H_0^1(Q)$ tel que

$$\begin{pmatrix} \bar\rho \\ \bar m \end{pmatrix} = \begin{pmatrix} \partial_x \bar\phi \\ -\partial_t \bar\phi \end{pmatrix} = \begin{pmatrix} \rho - (t\rho_1 + (1-t)\rho_0) \\ m + \int_0^x \rho_1(y) - \rho_0(y)dy \end{pmatrix}$$

Soient $\check{\phi}$ tel que $\check{\phi}(t,x) = tR_1(x) + (1-t)R_0(x)$ et $\phi = \bar{\phi} + \check{\phi}$. Alors

$$\begin{pmatrix} \partial_x \phi \\ -\partial_t \phi \end{pmatrix} = \begin{pmatrix} \partial_x \bar{\phi} + \partial_x \check{\phi} \\ -\partial_t \bar{\phi} - \partial_t \check{\phi} \end{pmatrix} = \begin{pmatrix} \bar{\rho} + (t\rho_1 + (1-t)\rho_0) \\ \bar{m} - \int_0^x \rho_1(y) - \rho_0(y) dy \end{pmatrix} = \begin{pmatrix} \rho \\ m \end{pmatrix}$$

avec

$$\begin{cases} \phi(0,x) = 0 + R_0(x) \\ \phi(1,x) = 0 + R_1(x) \\ \phi(t,0) = tR_1(0) + (1-t)R_0(0) = 0 \\ \phi(t,1) = tR_1(1) + (1-t)R_0(1) = 1 \end{cases}.$$

□

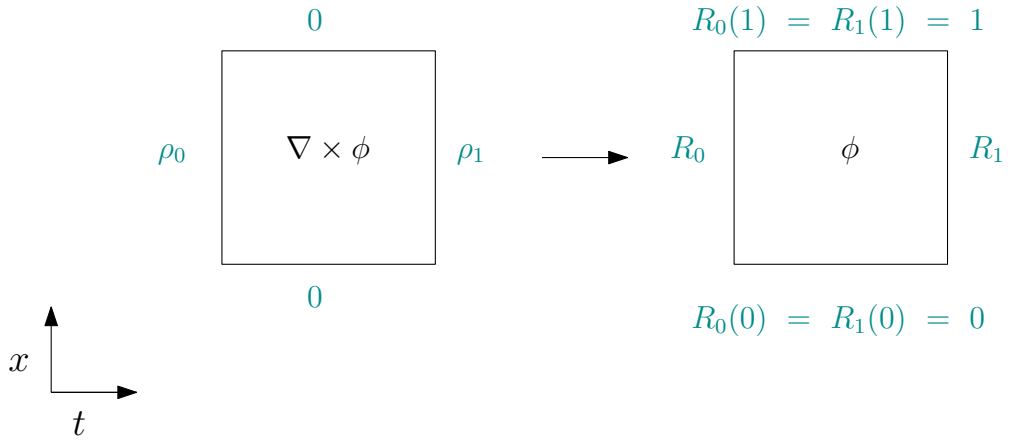


FIGURE 3.1: Conditions au bord satisfaites par ϕ d'après celles satisfaites par $\nabla \times \phi$.

Dans ce cas, l'énergie que l'on considère est pour $\partial_x \phi > 0$,

$$\boxed{E(\phi) = \mathcal{J}(\nabla \times \phi) = \int_0^1 \int_\Omega J(\nabla \times \phi(t,x)) dx dt = \int\int_Q \frac{(\partial_t \phi)^2}{\partial_x \phi} dt dx,} \tag{3.18}$$

$\phi \in H^1(Q)$, $\partial_x \phi \in L^\infty(Q)$. Cette fonctionnelle est **convexe** comme composée d'une fonctionnelle convexe et d'un opérateur linéaire.

Proposition 7. *En dimension 1 en espace, chercher le minimum de (3.18) où $\rho = \partial_x \phi > 0$ est équivalent à résoudre l'équation :*

$$\operatorname{div}_{t,x} \frac{\nabla \phi}{|\nabla \phi|} = 0. \tag{3.19}$$

Démonstration. La fonctionnelle étant convexe, formellement, étudier pour quels $\phi \in H_0^1(Q)$ dE s'annule est équivalent à résoudre :

$$dE(\phi)(\psi) = 0, \ \forall \psi \in H_0^1(Q).$$

Sachant que $E(\phi) = \int_Q J(\nabla \times \phi) dxdt$, où $J(X,Y) = \frac{Y^2}{X}$, on a

$$\partial_X J(X,Y) = -\frac{Y^2}{2X^2} \text{ et } \partial_Y J(X,Y) = \frac{Y}{X}$$

et donc, utilisant $\nabla \times \phi = (\partial_x \phi, -\partial_t \phi)$, la différentielle de E est donnée par

$$\begin{aligned} dE(\phi)(\psi) &= \int_Q \left[\partial_X J(\partial_x\phi, -\partial_t\phi)\partial_x\psi - \partial_Y J(\partial_x\phi, -\partial_t\phi)\partial_t\psi\right] dxdt \\ &= \int_Q \left(-\frac{(\partial_t\phi)^2}{2(\partial_x\phi)^2}\partial_x\psi(t,x) - \frac{-\partial_t\phi}{\partial_x\phi}\partial_t\psi(t,x)\right) dxdt = 0. \end{aligned}$$

Comme cette équation est vraie pour tout $\psi \in H_0^1(Q)$, on obtient

$$\partial_t\left(\frac{-\partial_t\phi}{\partial_x\phi}\right) + \frac{1}{2}\partial_x\left(\frac{(\partial_t\phi)^2}{(\partial_x\phi)^2}\right) = 0. \tag{3.20}$$

Si on définit maintenant $u = -\partial_t\phi$ et $v = \partial_x\phi$, l'équation (3.20) se réécrit :

$$\begin{aligned} \partial_t\left(\frac{u}{v}\right) + \frac{1}{2}\partial_x\left(\frac{u^2}{v^2}\right) &= \frac{v\partial_t u - u\partial_t v}{v^2} + \frac{1}{2}\frac{2uv^2\partial_x u - u^2 2v\partial_x v}{v^4} \\ &= \frac{v^2\partial_t u - uv\partial_t v + uv\partial_x u - u^2\partial_x v}{v^3}. \end{aligned}$$

En utilisant les mêmes notations que dans (3.19), on obtient :

$$\begin{aligned} \mathrm{div}_{t,x}\left(\frac{(-u,v)}{(u^2+v^2)^{1/2}}\right) &= \partial_t\left(\frac{-u}{(u^2+v^2)^{1/2}}\right) + \partial_x\left(\frac{v}{(u^2+v^2)^{1/2}}\right) \\ &= -\frac{\partial_t u(u^2+v^2) - u(u\partial_t u + v\partial_t v)}{(u^2+v^2)^{3/2}} + \frac{\partial_x v(u^2+v^2) - v(u\partial_x u + v\partial_x v)}{(u^2+v^2)^{3/2}} \\ &= \frac{-v^2\partial_t u + uv\partial_t v + u^2\partial_x v - vu\partial_x u}{(u^2+v^2)^{3/2}}. \end{aligned} \tag{3.21}$$

Comme on suppose $v = \partial_x\phi > 0$, (3.20) implique

$$\mathrm{div}_{t,x}\left(\frac{(-(-\partial_t\phi), \partial_x\phi)}{((-\partial_t\phi)^2 + (\partial_x\phi)^2)^{1/2}}\right) = 0.$$

et on obtient l'équation voulue (3.19). □

Remarques.

- La proposition précédente est intéressante car la minimisation de la nouvelle fonctionnelle est bien plus simple que celle de la fonctionnelle originelle, comme nous le verrons dans la partie 4.2.

- En dimension 2, les hessiennes des deux fonctions $J(X,Y) = \frac{Y^2}{2X}$ et $\overline{J}(X,Y) = \sqrt{X^2+Y^2}$ sont proportionnelles :

$$\begin{aligned} H_J = & \quad \tfrac{1}{X^3}\begin{pmatrix} Y^2 & -XY \\ -XY & X^2 \end{pmatrix} \\ = & \quad \tfrac{X^2+Y^2}{X^3}[I_2 - \tfrac{(X,Y)}{\sqrt{X^2+Y^2}} \otimes \tfrac{(X,Y)}{\sqrt{X^2+Y^2}}] \end{aligned}$$

et

$$\begin{aligned} H_{\overline{J}} = & \quad \tfrac{1}{\sqrt{X^2+Y^2}^3}\begin{pmatrix} Y^2 & -XY \\ -XY & X^2 \end{pmatrix} \\ = & \quad \tfrac{X^2+Y^2}{\sqrt{X^2+Y^2}^3}[I_2 - \tfrac{(X,Y)}{\sqrt{X^2+Y^2}} \otimes \tfrac{(X,Y)}{\sqrt{X^2+Y^2}}] = \tfrac{X^3}{\sqrt{X^2+Y^2}} H_J \end{aligned}$$

ce qui donne, si $''\!:\!''$ est la multiplication terme à terme de deux matrices

$$\begin{aligned} dE(\phi)(\psi) &= \int_Q [\partial_X J(\nabla\times\phi)\partial_x\psi - \partial_Y J(\nabla\times\phi)\partial_t\psi]\, dxdt \\ &= -\int_Q H_J(\nabla\times\phi) : \begin{pmatrix} \partial_{xx}\phi & -\partial_{xt}\phi \\ -\partial_{xt}\phi & \partial_{tt}\phi \end{pmatrix} \psi dxdt \\ &= -\int_Q \frac{|\nabla\times\phi|}{(\partial_x\phi)^3} H_{\overline{J}}(\nabla\times\phi) : \begin{pmatrix} \partial_{xx}\phi & -\partial_{xt}\phi \\ -\partial_{xt}\phi & \partial_{tt}\phi \end{pmatrix} \psi dxdt. \end{aligned}$$

Comme on a supposé que $\rho = \partial_x\phi > 0$, on obtient

$$dE(\phi) = 0 \iff d\overline{E}(\phi) = 0,$$

où on a noté $\bar{E}(\phi) = \iint_Q \bar{J}(\nabla\times\phi)$. De plus, les normes euclidiennes $|\nabla\phi|$ et $|\nabla\times\phi|$ étant égales, minimiser

$$\int_0^1 \int_\Omega |\nabla\times\phi(t,x)| dxdt$$

est équivalent à minimiser

$$\int_0^1 \int_\Omega |\nabla\phi(t,x)| dxdt,$$

dont l'équation d'Euler-Lagrange est (3.19.
On a donc obtenu une autre façon de démontrer le résultat de la proposition précédente.
- On peut remarquer l'orthogonalité des vecteurs $\nabla\phi$ et $\nabla\times\phi$, qui est à l'origine de simplifications dans l'équation (3.21).
- Le tenseur $I_2 - \frac{(X,Y)}{\sqrt{X^2+Y^2}} \otimes \frac{(X,Y)}{\sqrt{X^2+Y^2}}$ est une projection sur le plan orthogonal au vecteur $\frac{(X,Y)}{\sqrt{X^2+Y^2}}$.

- Comme (ρ, m) est à divergence nulle dans $(L^2(Q))^2$, il admet une trace normale et donc $\nabla \times \phi \cdot \nu_Q$ est défini sur ∂Q. De plus, $\nabla \times \phi \cdot \nu_Q = -\nabla\phi \cdot \tau_Q$, où τ_Q est la tangente de Q, donc $\nabla\phi$ admet une trace tangentielle sur ∂Q.

Ainsi, dans le cas unidimensionel en espace nous nous ramenons à la résolution d'une équation de type **courbure minimale sur chaque ligne de niveau du potentiel** munie des conditions de Dirichlet appropriées (3.17).
L'équation (3.19) est l'équation d'Euler-Lagrange de

$$\boxed{\mathcal{J}_{ms}(\nabla\phi) := \int_Q |\nabla\phi(t,x)| dt dx} \tag{3.22}$$

et c'est cette fonctionnelle que nous minimiserons dans la suite. Elle a été étudiée tout d'abord par Rudin, Osher et Fatemi dans [79] pour la reconstruction d'image. Elle est définie pour des fonctions $\phi \in BV(Q)$, où

$$BV(Q) = \{f \in L^1(Q),\ \int_Q |\nabla f(t,x)| dt dx < +\infty\}.$$

Il a été démontré [39] que cette fonctionnelle est semi continue inférieurement et convexe. De plus, on peut prouver qu'elle est strictement convexe sur l'espace des contraintes pour ϕ :

$$C_\phi = \{\phi \in BV(Q);\ |\phi| \leq 1,\ \phi(0,x) = R_0(x),\ \phi(1,x) = R_1(x),\ \phi(t,0) = 0,\ \phi(t,1) = 1\},$$

Proposition 8. *La fonctionnelle $\mathcal{J}_{ms}$ est strictement convexe sur l'espace des contraintes C_ϕ.*

Démonstration. Soit $\alpha \in]0,1[$ et $\phi_1, \phi_2 \in C_\phi$

$$\begin{aligned}
\int_Q J_{ms}(\nabla(\alpha\phi_1 + (1-\alpha)\phi_2)) = \alpha \int_Q J_{ms}(\nabla\phi_1) + (1-\alpha)\int_Q J_{ms}(\nabla\phi_2) \\
\iff \int_Q |\nabla(\alpha\phi_1 + (1-\alpha)\phi_2)| = \alpha \int_Q |\nabla\phi_1| + (1-\alpha)\int_Q |\nabla\phi_2| \\
\iff \int_Q (|\nabla(\alpha\phi_1 + (1-\alpha)\phi_2)| - (\alpha|\nabla\phi_1| + (1-\alpha)|\nabla\phi_2|)) = 0
\end{aligned}$$

Comme l'intégrande est négative ou nulle, cela équivaut, en notant $a = \alpha\nabla\phi_1$ et $b = (1-\alpha)\nabla\phi_2$, à

$$(|a+b| - (|a| + |b|)) = 0 \text{ ou } |a+b| = |a| + |b| \text{ pour presque tout } (t,x) \in Q.$$

Cette égalité n'est possible que si il existe $\lambda(t,x) \geq 0$ tel que $\alpha\nabla\phi_1(t,x) = \lambda(t,x)(1-\alpha)\nabla\phi_2(t,x)$. On peut donc reécrire

$$\nabla\phi_1 = \lambda(t,x)\nabla\phi_2. \tag{3.23}$$

En prenant le rotationnel de cette équation on obtient $0 = \nabla \times (\nabla\phi_1) = \nabla \times \lambda \cdot \nabla\phi_2$. Donc $\nabla \times \lambda$ et $\nabla\phi_2$ sont perpendiculaires. De plus, comme $\nabla \times \lambda$ est perpendiculaire à $\nabla\lambda$, $\nabla\phi_2$ est parallèle à $\nabla\lambda$ et donc λ est une fonction de ϕ_2. De l'égalité (3.23) on déduit que ϕ_1 s'écrit en fonction de ϕ_2, $\phi_1 = F(\phi_2)$. Par ailleurs, la condition au bord en $t = 0$ ou $t = 1$ nous donne que pour tout $x \in [0, 1]$, $\phi_1(0, x) = \phi_2(0, x)$. Or, $\phi_1(0, x)$ prend toutes les valeurs entre 0 et 1 comme intégrale de 0 à x (où $x \in [0, 1]$) d'une fonction L^∞ de masse 1. On a donc $x = F(x)$ pour $x \in [0, 1]$ et F est l'identité sur cet intervalle. □

La formule de la "co-aire" de Federer et Fleming, démontrée dans [31, 32, 2], dit que pour $\phi \in BV(Q)$ et si $Per(A, Q)$ désigne le périmètre de A dans Q

$$\int_Q |\nabla\phi| = \int_{-\infty}^{+\infty} Per(\{\phi > s\}, Q)ds$$

ce qui implique que la variation totale de ϕ est aussi la somme de toutes les longueurs des lignes de niveau.

3.2.2 Transport optimal 2D

On voudrait à présent généraliser la proposition 6 en dimension deux en espace. Cependant, l'équation donnant les conditions au bord n'est pas triviale à résoudre dans ce cas, en effet, le rotationnel devient :

$$(\rho, m) = \nabla \times \begin{pmatrix}\phi_1\\ \phi_2\\ \phi_3\end{pmatrix} = \begin{pmatrix}\partial_x\phi_3 - \partial_y\phi_2\\ \partial_y\phi_1 - \partial_t\phi_3\\ \partial_t\phi_2 - \partial_x\phi_1\end{pmatrix},\ \phi = \begin{pmatrix}\phi_1\\ \phi_2\\ \phi_3\end{pmatrix} \in (H^1(Q))^3. \tag{3.24}$$

Par conséquent, pour trouver les conditions au bord à respecter il faut résoudre un système du type

$$\begin{pmatrix}\partial_x\phi_3 - \partial_y\phi_2\\ \partial_y\phi_1 - \partial_t\phi_3\\ \partial_t\phi_2 - \partial_x\phi_1\end{pmatrix} = \begin{pmatrix}\rho\\ m_1\\ m_2\end{pmatrix} \text{ sur } \partial Q. \tag{3.25}$$

Une autre solution, sachant que le couple (ρ, m) est à divergence nulle, est d'écrire, par exemple, $\rho = -\int_0^t (\partial_x m_1 + \partial_y m_2)$ et donc d'annuler une des trois composantes de ϕ :

$$\phi = \begin{pmatrix}\phi_1\\ \phi_2\\ \phi_3\end{pmatrix} = \begin{pmatrix}0\\ \int_0^t m_2 ds\\ -\int_0^t m_1 ds\end{pmatrix}.$$

ce qui donne bien

$$\nabla \times \phi = \begin{pmatrix}-\partial_x \int_0^t m_1 ds - \partial_y \int_0^t m_2 ds\\ \partial_y 0 - \partial_t(-\int_0^t m_1 ds)\\ \partial_t \int_0^t m_2 - \partial_x 0\end{pmatrix} = \begin{pmatrix}\rho\\ m_1\\ m_2\end{pmatrix}$$

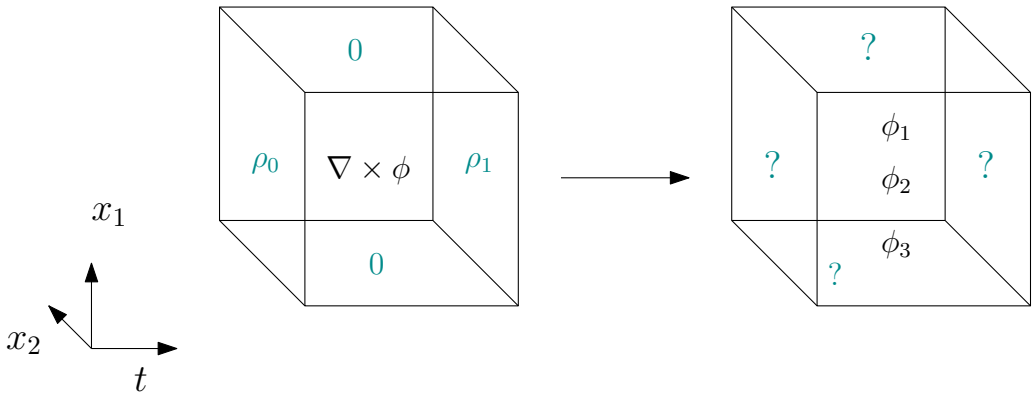


FIGURE 3.2: Conditions au bord satisfaites par ϕ d'après celles satisfaites par $\nabla \times \phi$.

Cependant, comme ϕ est également à divergence nulle, (ρ, m) vérifie maintenant pour tout t

$$\int_0^t \partial_x m_2 - \partial_y m_1 ds = 0.$$

On a imposé en annulant ϕ_1 que (ϕ_2, ϕ_3) soit à divergence nulle et que l'intégrale en temps du rotationnel scalaire en espace de m soit nulle, ce qui n'est pas vrai en général. Cela privilégie une direction ce qui n'est pas désiré dans notre formulation et pose problème numériquement. Un exemple sera donné dans la section 5.2.2.

Nous allons ainsi utiliser la décomposition de Helmholtz-Hodge explicitée dans le paragraphe suivant pour ne pas à avoir à résoudre le système (3.25).

3.3 Décomposition de Helmholtz-Hodge

Pour travailler en 1D et 2D, on peut utiliser la décomposition orthogonale de $L^2(Q)^{1+n}$, pour $n = 1$ ou $n = 2$, détaillée dans [38]. Elle consiste à décomposer en somme directe un vecteur de $L^2(Q)^{1+n}$ en une composante à divergence nulle et une composante à rotationnel nul. Tout champ de vecteurs $v = (\rho, m) \in V(Q)$ défini par (2.8) a la **décomposition de Helmholtz-Hodge** suivante :

$$\boxed{(\rho, m) = \nabla \times \phi + \nabla h}$$

avec $h \in H^1(Q)/\mathbb{R}$ et $\phi \in H_0^1(Q)$, pour $n+1 = 2$ et $\phi \in (H_0^1(Q))^3$, pour $n+1 = 3$. Par ailleurs, dans le cas $n+1 = 3$, div$\phi = 0$. Comme (ρ, m) est à divergence nulle, h vérifie :

$$\begin{cases} \Delta h = 0 \text{ dans } Q, \\ \frac{\partial h}{\partial \nu_Q} = (\rho, m) \cdot \nu_Q \text{ sur } \partial Q, \end{cases} \tag{3.26}$$

où ν_Q est la normale sortante de Q. Nous devrons donc résoudre tout d'abord le système (3.26) en h, qui est une **équation de Poisson** avec des conditions de Neumann connues ; en effet,

$$\frac{\partial h}{\partial \nu_Q} = (\rho, m)\cdot\nu_Q \text{ sur } \partial Q \iff \left\{ \begin{array}{l} \frac{\partial h}{\partial \nu_Q}(.,x) = m(.,x).\nu_\Omega = 0,\ \forall x \in \partial\Omega, \\ \frac{\partial h}{\partial \nu_Q}(0,.) = \rho(0,.) = \rho_0,\ \frac{\partial h}{\partial \nu_Q}(1,.) = \rho(1,.) = \rho_1. \end{array} \right.$$

Ensuite, connaissant h, il faut minimiser la nouvelle énergie

$$\boxed{E(\phi) = \mathcal{J}_h(\nabla\times\phi) = \int_0^1\int_\Omega J(\nabla\times\phi(t,x) + \nabla h(t,x))dxdt,} \tag{3.27}$$

où J est définie par (2.16). **La fonctionnelle $\mathcal{J}_h$ est convexe** comme composée de la fonctionnelle $\mathcal{J}$ qui est convexe et d'un opérateur affine.

Proposition 9. *En dimension 1 en espace, chercher le minimum de (3.27) où $\rho = \partial_x\phi + \partial_t h > 0$, avec h connu, est formellement équivalent à résoudre l'équation :*

$$\mathrm{div}_{t,x}\frac{\nabla\phi - \nabla\times h}{|\nabla\phi - \nabla\times h|} = 0. \tag{3.28}$$

Démonstration. Pour démontrer cette proposition, nous adoptons la même méthode que pour la proposition 7.
La fonctionnelle étant convexe, étudier pour quels $\phi \in H^1_0(Q)$ on annule dE est équivalent à résoudre :

$$dE(\phi)(\psi) = 0,\ \forall\psi \in H^1_0(Q).$$

On sait que $E(\phi) = \int_Q J(\nabla\times\phi + \nabla h)dxdt$ et que la différentielle de E est donnée par

$$\begin{aligned} dE(\phi)(\psi) &= \int_Q \left[\partial_X J(\nabla\times\phi + \nabla h)\partial_x\psi - \partial_Y J(\nabla\times\phi + \nabla h)\partial_t\psi\right] dxdt \\ &= \int_Q \left(-\frac{(-\partial_t\phi + \partial_x h)^2}{2(\partial_x\phi + \partial_t h)^2}\partial_x\psi(t,x) - \frac{-\partial_t\phi + \partial_x h}{\partial_x\phi + \partial_t h}\partial_t\psi(t,x)\right) dxdt = 0. \end{aligned}$$

Comme cette équation est vraie pour tout $\psi \in H^1_0(Q)$, on obtient

$$\partial_t\left(\frac{-\partial_t\phi + \partial_x h}{\partial_x\phi + \partial_t h}\right) + \frac{1}{2}\partial_x\left(\frac{(-\partial_t\phi + \partial_x h)^2}{(\partial_x\phi + \partial_t h)^2}\right) = 0. \tag{3.29}$$

Si on définit maintenant $u = -\partial_t\phi + \partial_x h$ et $v = \partial_x\phi + \partial_t h$, l'équation (3.29) se réécrit comme dans la preuve de la proposition 7. On retrouve les mêmes résultats, c'est-à-dire pour $v = \partial_x\phi + \partial_t h > 0$, (3.29) implique

$$\mathrm{div}_{t,x}\left(\frac{(-(-\partial_t\phi + \partial_x h), \partial_x\phi + \partial_t h)}{((-\partial_t\phi + \partial_x h)^2 + (\partial_x\phi + \partial_t h)^2)^{1/2}}\right) = 0.$$

et on obtient l'équation voulue (3.28). □

Ainsi, en 1D chercher le minimum de E dans (3.27) revient à résoudre une équation de **type courbure minimale sur chaque ligne de niveau du potentiel** ϕ. On obtient donc le système suivant :

$$\begin{cases} \Delta h = 0 \text{ dans } Q, \\ \frac{\partial h}{\partial \nu_Q} = (\rho, m) \cdot \nu_Q \text{ sur } \partial Q \end{cases} \tag{3.30}$$

et

$$\begin{cases} \mathrm{div}_{t,x} \frac{\nabla\phi - \nabla\times h}{|\nabla\phi - \nabla\times h|} = 0, \\ \phi = 0 \text{ sur } \partial Q. \end{cases} \tag{3.31}$$

L'équation (3.31) est l'équation d'Euler-Lagrange de la **fonctionnelle convexe** (comme composée d'une fonctionnelle convexe et d'un opérateur linéaire)

$$\boxed{\mathcal{J}_{ms,h}(\nabla\phi) := \int_Q |\nabla\phi - \nabla \times h| dxdt,} \tag{3.32}$$

dont nous détaillerons la minimisation dans la section 4.2.3.

Remarques.

- Comme précédemment, la minimisation de la nouvelle fonctionnelle est bien plus simple que celle de la fonctionnelle originelle (voir section 4.2).
- Les remarques de la section 3.2.1 sont toujours valides ici car les normes $|\nabla\phi - \nabla \times h|$ et $|\nabla \times \phi + \nabla h|$ sont égales.
  Malheureusement ce résultat n'est pas généralisable à la dimension supérieure. En dimension 3, les hessiennes des deux fonctions $F(X,Y,Z) = \frac{Y^2+Z^2}{2X}$ et $\overline{F}(X,Y,Z) = \sqrt{X^2+Y^2+Z^2}$ ne sont pas proportionnelles.

$$H_F = \frac{1}{X^3} \begin{pmatrix} Y^2+Z^2 & -XY & -XZ \\ -XY & X^2 & 0 \\ -XZ & 0 & X^2 \end{pmatrix}$$
$$= (X^2+Y^2+Z^2)I_3 - (X,Y,Z)\otimes(X,Y,Z) - (0,Z,-Y)\otimes(0,Z,-Y)$$

  et

$$H_{\overline{F}} = \frac{1}{\sqrt{X^2+Y^2+Z^2}^3} \begin{pmatrix} Y^2+Z^2 & -XY & -XZ \\ -XY & X^2+Z^2 & -YZ \\ -XZ & -YZ & X^2+Y^2 \end{pmatrix}$$
$$= (X^2+Y^2+Z^2)I_3 - (X,Y,Z)\otimes(X,Y,Z)$$

- On peut remarquer la L^2-orthogonalité de $\nabla\phi$ et $\nabla \times h$ grâce à la décomposition utilisée et l'orthogonalité de $\nabla\phi - \nabla \times h$ et $\nabla \times \phi + \nabla h$ qui nous permettent ici de simplifier les calculs comme c'était le cas pour (3.21).

Chapitre 4

Algorithmes de résolution dans l'espace des contraintes

Dans le chapitre précédent nous avons développé plusieurs formulations nouvelles pour le problème de transport optimal dynamique dans l'espace des contraintes. Nous décrivons à présent les algorithmes utilisés pour implémenter ces nouvelles formulations. Pour cela, nous commençons par détailler l'algorithme de gradient à pas fixe constant utilisé pour la décomposition à l'aide des ondelettes à divergence nulle, périodiques ou vérifiant des conditions de Dirichlet homogènes. Cela aboutit à deux algorithmes différents pour chacune de ces bases. Le premier utilise les ondelettes duales des ondelettes considérées et le second leurs matrices de Gram. Par la suite, nous explicitons un algorithme primal-dual pour problèmes convexes dans le cas de la décomposition de Helmholtz-Hodge et de la représentation par fonction de courant. Nous détaillons le cas 1D pour la minimisation de la fonctionnelle $\mathcal{J}_h$ avec ces deux représentations ainsi que l'implémentation de la formulation de type courbure minimale, également avec ces deux décompositions. Dans le cas 2D nous développons la minimisation de la fonctionnelle $\mathcal{J}_h$ à l'aide de la décomposition de Helmholtz-Hodge.

4.1 Descente de gradient dans l'espace des coefficients d'ondelettes à divergence nulle

4.1.1 Algorithme dans l'espace des coefficients à divergence nulle

Pour tester notre nouvelle formulation à l'aide des ondelettes à divergence nulle (périodiques ou sur l'intervalle, détaillées dans la section 3.1.3), nous utilisons un algorithme de gradient à pas fixe. Il permet de résoudre le problème

$$\min_{\mathbf{d}^{div}} \mathcal{J}_w(\mathbf{d}^{div}),$$

avec

$$\mathcal{J}_w(\mathbf{d}^{div}) = \int_0^1 \int_0^1 \frac{(\tilde{m}(\mathbf{d}^{div}) - \int_0^x (\rho_1(y) - \rho_0(y))dy)^2}{\tilde{\rho}(\mathbf{d}^{div}) + t\rho_1(x) + (1-t)\rho_0(x)} dxdt,$$

$\mathcal{J}_w$ est différentiable et convexe, comme énoncé dans la section 3.1.4.

Algorithme 1. *(Descente de gradient)*
Initialisation : $\tau > 0, (\mathbf{d}^{div})^0$.
Itérations :

$$\begin{aligned} &\textit{calcul de } \nabla J_w((\mathbf{d}^{div})^i) \\ &(\mathbf{d}^{div})^{i+1} = (\mathbf{d}^{div})^i - \tau \nabla J_w((\mathbf{d}^{div})^i) \end{aligned}$$

Comme détaillé dans la section 3.1.3, nous pouvons dénoter, de la même façon pour des ondelettes périodiques ou sur l'intervalle,

$$\begin{aligned} \begin{pmatrix} \tilde{\rho} \\ \tilde{m} \end{pmatrix} &= \sum_{j_{min},\mathbf{k}} c^{div}_{j_{min},\mathbf{k}} \Phi^{div}_{j_{min},\mathbf{k}} + \sum_{\epsilon} \sum_{\mathbf{j},\mathbf{k}} d^{div,\epsilon}_{\mathbf{j},\mathbf{k}} \Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}} \\ &= \sum_{j_{min},\mathbf{k}} c^{div}_{j_{min},\mathbf{k}} \begin{pmatrix} (\Phi^{div}_{j_{min},\mathbf{k}})^1 \\ (\Phi^{div}_{j_{min},\mathbf{k}})^2 \end{pmatrix} + \sum_{\epsilon} \sum_{\mathbf{j},\mathbf{k}} d^{div,\epsilon}_{\mathbf{j},\mathbf{k}} \begin{pmatrix} (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 \\ (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 \end{pmatrix}, \end{aligned}$$

où

$$\begin{aligned} \Phi^{div}_{j_{min},\mathbf{k}} &:= \frac{1}{\sqrt{1+1}} \nabla \times (\varphi_{j_{min},k_1} \otimes \varphi_{j_{min},k_2}) = \frac{1}{\sqrt{1+1}} \begin{pmatrix} \varphi_{j_{min},k_1}(t)(\varphi_{j_{min},k_2})'(x) \\ -(\varphi_{j_{min},k_1})'(t)\varphi_{j_{min},k_2}(x) \end{pmatrix} \\ \Psi^{div,1}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_2}+1}} \nabla \times (\varphi_{j_{min},k_1} \otimes \psi_{j_2,k_2}) = \frac{1}{\sqrt{4^{j_2}+1}} \begin{pmatrix} \varphi_{j_{min},k_1}(t)(\psi_{j_2,k_2})'(x) \\ -(\varphi_{j_{min},k_1})'(t)\psi_{j_2,k_2}(x) \end{pmatrix} \\ \Psi^{div,2}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_1}+1}} \nabla \times (\psi_{j_1,k_1} \otimes \varphi_{j_{min},k_2}) = \frac{1}{\sqrt{4^{j_1}+1}} \begin{pmatrix} \psi_{j_1,k_1}(t)(\varphi_{j_{min},k_2})'(x) \\ -(\psi_{j_1,k_1})'(t)\varphi_{j_{min},k_2}(x) \end{pmatrix} \\ \Psi^{div,3}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_1}+4^{j_2}}} \nabla \times (\psi_{j_1,k_1} \otimes \psi_{j_2,k_2}) = \frac{1}{\sqrt{4^{j_1}+4^{j_2}}} \begin{pmatrix} \psi_{j_1,k_1}(t)(\psi_{j_2,k_2})'(x) \\ -(\psi_{j_1,k_1})'(t)\psi_{j_2,k_2}(x) \end{pmatrix}. \end{aligned} \tag{4.1}$$

Nous noterons par simplicité dans la suite $\Psi^{div,0}_{\mathbf{j},\mathbf{k}} := \Phi^{div}_{j_{min},\mathbf{k}}$.

Soit $\mathbf{d}^{\text{div}} = (d^{div,\epsilon}_{j,\mathbf{k}})$, on souhaite minimiser

$$\begin{aligned} \mathcal{J}_w(\mathbf{d}^{\text{div}}) &= \int_0^1 \int_0^1 \frac{(\tilde{m}(\mathbf{d}^{\text{div}}) - \int_0^x (\rho_1(y) - \rho_0(y))dy)^2}{\tilde{\rho}(\mathbf{d}^{\text{div}}) + t\rho_1(x) + (1-t)\rho_0(x)} dxdt \\ &= \int_0^1 \int_0^1 \frac{(\sum_\epsilon \sum_{\mathbf{j},\mathbf{k}} d^{\text{div},\epsilon}_{\mathbf{j},\mathbf{k}} (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 - \int_0^x (\rho_1(y) - \rho_0(y))dy)^2}{\sum_\epsilon \sum_{\mathbf{j},\mathbf{k}} d^{\text{div},\epsilon}_{\mathbf{j},\mathbf{k}} (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 + t\rho_1(x) + (1-t)\rho_0(x)} dxdt. \end{aligned}$$

La dérivation par rapport aux coefficients $d^{\text{div},\epsilon}_{\mathbf{j},\mathbf{k}}$ est

$$\begin{aligned} \frac{\partial \mathcal{J}_w}{\partial d^{div,\epsilon}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{\text{div}}) = \int_0^1 \int_0^1 &2\frac{\tilde{m}(\mathbf{d}^{\text{div}}) - \int_0^x (\rho_1(y) - \rho_0(y))dy}{\tilde{\rho}(\mathbf{d}^{\text{div}}) + t\rho_1(x) + (1-t)\rho_0(x)} (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 \\ &- \frac{(\tilde{m}(\mathbf{d}^{\text{div}}) - \int_0^x (\rho_1(y) - \rho_0(y))dy)^2}{(\tilde{\rho}(\mathbf{d}^{\text{div}}) + t\rho_1(x) + (1-t)\rho_0(x))^2} (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 dxdt \end{aligned}$$

Il y a deux possibilités pour calculer le gradient de $\mathcal{J}_w$:

— Voir $\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\text{div}})}{\rho(\mathbf{d}^{\text{div}})}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 dxdt$ et $\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\text{div}})^2}{\tilde{\rho}(\mathbf{d}^{\text{div}})^2}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 dxdt$ comme les coefficients d'ondelettes de $\frac{m}{\rho}$ et $\frac{m^2}{\rho^2}$ sur les bases biorthogonales de $\left(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}}\right)^2$ et $\left(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}}\right)^1$ respectivement.

— Décomposer $\frac{(\tilde{m}(\mathbf{d}^{\text{div}})-\int_0^x(\rho_1(y)-\rho_0(y))dy)}{(\tilde{\rho}(\mathbf{d}^{\text{div}})+t\rho_1(x)+(1-t)\rho_0(x))}$ et $\frac{(\tilde{m}(\mathbf{d}^{\text{div}})-\int_0^x(\rho_1(y)-\rho_0(y))dy)^2}{(\tilde{\rho}(\mathbf{d}^{\text{div}})+t\rho_1(x)+(1-t)\rho_0(x))^2}$ sur les ondelettes $\Psi^{1,\epsilon}_{\mathbf{j}',\mathbf{k}'}$ et $\Psi^{2,\epsilon}_{\mathbf{j}',\mathbf{k}'}$ définies dans les équations (3.11) et (3.12), pour faire apparaître leurs matrices de Gram.

Nous détaillons dans la suite ces deux méthodes.

4.1.2 Calcul du gradient à l'aide des ondelettes duales

La première méthode pour calculer le gradient consiste à voir $\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\text{div}})}{\rho(\mathbf{d}^{\text{div}})}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 dxdt$ et $\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\text{div}})^2}{\tilde{\rho}(\mathbf{d}^{\text{div}})^2}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 dxdt$ comme les coefficients d'ondelettes de $\frac{m}{\rho}$ et $\frac{m^2}{\rho^2}$ sur les bases biorthogonales de $\left(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}}\right)^2$ et $\left(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}}\right)^1$ respectivement.

Utilisation des ondelettes périodiques

Dans le cas des ondelettes périodiques $j_{min} = 0$, $\varphi_{j_{min}} = 1_{[0,1]}$ et nous considérons $\psi = \psi^1$ (voir proposition 5). Nous avons donc comme base d'ondelettes à divergence nulle comme explicité dans la section 3.1.3 :

$$\begin{aligned}
\Psi^{div,0}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{1+1}}\nabla\times(1\otimes 1) = \frac{1}{\sqrt{1+1}}\begin{pmatrix}0\\0\end{pmatrix}\\
\Psi^{div,1}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_2}+1}}\nabla\times(1\otimes\psi^1_{j_2,k_2}) = \frac{1}{\sqrt{4^{j_2}+1}}\begin{pmatrix}1(t)(\psi^1_{j_2,k_2})'(x)\\0\end{pmatrix}\\
\Psi^{div,2}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_1}+1}}\nabla\times(\psi^1_{j_1,k_1}\otimes 1) = \frac{1}{\sqrt{4^{j_1}+1}}\begin{pmatrix}0\\-(\psi^1_{j_1,k_1})'(t)1(x)\end{pmatrix}\\
\Psi^{div,3}_{\mathbf{j},\mathbf{k}} &:= \frac{1}{\sqrt{4^{j_1}+4^{j_2}}}\nabla\times(\psi^1_{j_1,k_1}\otimes\psi^1_{j_2,k_2}) = \frac{1}{\sqrt{4^{j_1}+4^{j_2}}}\begin{pmatrix}\psi^1_{j_1,k_1}(t)(\psi^1_{j_2,k_2})'(x)\\-(\psi^1_{j_1,k_1})'(t)\psi^1_{j_2,k_2}(x)\end{pmatrix},
\end{aligned} \tag{4.2}$$

En se rappelant que pour tout ϵ,

$$\begin{aligned}
\frac{\partial\mathcal{J}_w}{\partial d^{\text{div},\epsilon}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{\text{div}}) = \int_0^1\int_0^1 & 2\frac{\tilde{m}(\mathbf{d}^{\text{div}})-\int_0^x(\rho_1(y)-\rho_0(y))dy}{\tilde{\rho}(\mathbf{d}^{\text{div}})+t\rho_1(x)+(1-t)\rho_0(x)}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2\\
&-\frac{(\tilde{m}(\mathbf{d}^{\text{div}})-\int_0^x(\rho_1(y)-\rho_0(y))dy)^2}{(\tilde{\rho}(\mathbf{d}^{\text{div}})+t\rho_1(x)+(1-t)\rho_0(x))^2}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 dxdt\\
=&2\int_0^1\int_0^1\frac{m(\mathbf{d}^{\text{div}})}{\rho(\mathbf{d}^{\text{div}})}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 dxdt - \int_0^1\int_0^1\frac{m(\mathbf{d}^{\text{div}})^2}{\tilde{\rho}(\mathbf{d}^{\text{div}})^2}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 dxdt,
\end{aligned}$$

on obtient pour les différentes valeurs de $\epsilon = 1, 2$ ou 3

$$\begin{aligned}\frac{\partial \mathcal{J}_w}{\partial d^{\mathrm{div},1}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{\mathrm{div}}) &= 2\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})}{\rho(\mathbf{d}^{\mathrm{div}})}(\Psi^{div,1}_{\mathbf{j},\mathbf{k}})^2 dxdt - \int_0^1\int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})^2}{\tilde{\rho}(\mathbf{d}^{\mathrm{div}})^2}(\Psi^{div,1}_{\mathbf{j},\mathbf{k}})^1 dxdt \\ &= -\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})^2}{\tilde{\rho}(\mathbf{d}^{\mathrm{div}})^2} 1(t)(\psi^1_{j_2,k_2})'(x) dxdt,\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathcal{J}_w}{\partial d^{\mathrm{div},2}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{\mathrm{div}}) &= 2\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})}{\rho(\mathbf{d}^{\mathrm{div}})}(\Psi^{div,2}_{\mathbf{j},\mathbf{k}})^2 dxdt - \int_0^1\int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})^2}{\tilde{\rho}(\mathbf{d}^{\mathrm{div}})^2}(\Psi^{div,2}_{\mathbf{j},\mathbf{k}})^1 dxdt \\ &= 2\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})}{\rho(\mathbf{d}^{\mathrm{div}})}\left(-(\psi^1_{j_1,k_1})'(t)1(x)\right) dxdt,\end{aligned}$$

$$\frac{\partial \mathcal{J}_w}{\partial d^{\mathrm{div},3}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{\mathrm{div}}) = 2\int_0^1\int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})}{\rho(\mathbf{d}^{\mathrm{div}})}(\Psi^{div,3}_{\mathbf{j},\mathbf{k}})^2 dxdt - \int_0^1\int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})^2}{\tilde{\rho}(\mathbf{d}^{\mathrm{div}})^2}(\Psi^{div,3}_{\mathbf{j},\mathbf{k}})^1 dxdt.$$

On peut alors calculer le gradient de $\mathcal{J}_w$ en projetant $\frac{m}{\rho}$ sur les ondelettes duales $(\tilde{\Psi}^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2$ et $\frac{m^2}{\rho^2}$ sur $(\tilde{\Psi}^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1$, en utilisant la formule

$$\begin{pmatrix} -\left(\frac{m}{\rho}\right)^2 \\ 2\frac{m}{\rho}\end{pmatrix} = \sum_\epsilon \sum_{\mathbf{j},\mathbf{k}} d^\epsilon_{\mathbf{j},\mathbf{k}} \tilde{\Psi}^{div,\epsilon}_{\mathbf{j},\mathbf{k}}.$$

On écrit dans ce cas l'algorithme de descente suivant

Algorithme 2 (GDDWP)**.**

Initialisation : $\tau > 0, (\mathbf{d}^{div})^0$

Itérations :

$\tilde{m}^i = \sum_\epsilon \sum_{\boldsymbol{j},\boldsymbol{k}} (d^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^i (\psi^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^2$ *et* $\tilde{\rho}^i = \sum_\epsilon \sum_{\boldsymbol{j},\boldsymbol{k}} (d^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^i (\Psi^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^1$

$m^i = \tilde{m}^i - \int_0^x (\rho_1 - \rho_0) dy$ *et* $\rho^i = \tilde{\rho}^i + t\rho_1 + (1-t)\rho_0$

Calcul des coefficients d'ondelettes $d^1_{\boldsymbol{j},\boldsymbol{k}}, d^2_{\boldsymbol{j},\boldsymbol{k}}$ *et* $d^3_{\boldsymbol{j},\boldsymbol{k}}$ *du vecteur* $\left(-\left(\frac{m^i}{\rho^i}\right)^2, 2\frac{m^i}{\rho^i}\right)$

$(\nabla J_w)^i = \mathbf{d}$

$(\mathbf{d}^{div})^{i+1} = (\mathbf{d}^{div})^i - \tau(\nabla J_w)^i$

Utilisation des ondelettes respectant des conditions de type Dirichlet

Dans le cas des ondelettes sur $[0,1]^2$ $j_{min} \neq 0$ et nous considérons $\psi = \psi^d$ (voir définition 2). Il n'y a pas de simplifications comme dans le cas périodique et nous considérons la base d'ondelettes (4.1). En se rappelant que pour tout ϵ,

$$\begin{aligned}\frac{\partial \mathcal{J}_w}{\partial d^{\mathrm{div},\epsilon}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{\mathrm{div}}) = \int_0^1 \int_0^1 & 2\frac{\tilde{m}(\mathbf{d}^{\mathrm{div}}) - \int_0^x (\rho_1(y)-\rho_0(y))dy}{\tilde{\rho}(\mathbf{d}^{\mathrm{div}}) + t\rho_1(x) + (1-t)\rho_0(x)}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 \\ & - \frac{(\tilde{m}(\mathbf{d}^{\mathrm{div}}) - \int_0^x (\rho_1(y)-\rho_0(y))dy)^2}{(\tilde{\rho}(\mathbf{d}^{\mathrm{div}}) + t\rho_1(x) + (1-t)\rho_0(x))^2}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 dxdt \\ = & 2\int_0^1 \int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})}{\rho(\mathbf{d}^{\mathrm{div}})}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 dxdt - \int_0^1 \int_0^1 \frac{m(\mathbf{d}^{\mathrm{div}})^2}{\tilde{\rho}(\mathbf{d}^{\mathrm{div}})^2}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 dxdt,\end{aligned}$$

on peut calculer le gradient de $\mathcal{J}_w$ en projetant $\frac{m}{\rho}$ sur les ondelettes duales $(\tilde{\Psi}^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2$ et $\frac{m^2}{\rho^2}$ sur $(\tilde{\Psi}^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1$.

En utilisant la formule

$$\begin{pmatrix} -\left(\frac{m}{\rho}\right)^2 \\ 2\frac{m}{\rho} \end{pmatrix} = \sum_{j_{min},\mathbf{k}} c_{j_{min},\mathbf{k}} \Phi^{div}_{j_{min},\mathbf{k}} + \sum_{\epsilon}\sum_{\mathbf{j},\mathbf{k}} d^{\epsilon}_{\mathbf{j},\mathbf{k}} \Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}}$$

on aboutit dans ce cas à l'algorithme de descente suivant

Algorithme 3 (GDDWD)**.**

Initialisation : $\tau > 0, (\mathbf{d}^{div})^0$

Itérations :

$\tilde{m}^i = \sum_{\epsilon}\sum_{\boldsymbol{j},\boldsymbol{k}} (d^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^i (\psi^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^2$ *et* $\tilde{\rho}^i = \sum_{\epsilon}\sum_{\boldsymbol{j},\boldsymbol{k}} (d^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^i (\Psi^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^1$

$m^i = \tilde{m}^i - \int_0^x (\rho_1 - \rho_0)dy$ *et* $\rho^i = \tilde{\rho}^i + t\rho_1 + (1-t)\rho_0$

Calcul des coefficients d'ondelettes $c_{j_{min},\mathbf{k}}, d^1_{\boldsymbol{j},\boldsymbol{k}}, d^2_{\boldsymbol{j},\boldsymbol{k}}$ *et* $d^3_{\boldsymbol{j},\boldsymbol{k}}$ *du vecteur* $\left(-\left(\frac{m^i}{\rho^i}\right)^2, 2\frac{m^i}{\rho^i}\right)$

$(\nabla J_w)^i = (\mathbf{c}, \mathbf{d})$

$(\mathbf{d}^{div})^{i+1} = (\mathbf{d}^{div})^i - \tau(\nabla J_w)^i$

4.1.3 Calcul du gradient à l'aide des matrices de Gram

La deuxième méthode pour calculer ce gradient consiste en la projection des coefficients $\frac{(\tilde{m}(\mathbf{d}^{\mathrm{div}}) - \int_0^x (\rho_1(y)-\rho_0(y))dy)}{(\tilde{\rho}(\mathbf{d}^{\mathrm{div}}) + t\rho_1(x) + (1-t)\rho_0(x))}$ et $\frac{(\tilde{m}(\mathbf{d}^{\mathrm{div}}) - \int_0^x (\rho_1(y)-\rho_0(y))dy)^2}{(\tilde{\rho}(\mathbf{d}^{\mathrm{div}}) + t\rho_1(x) + (1-t)\rho_0(x))^2}$ sur les ondelettes $\Psi^{1,\epsilon}_{\mathbf{j}',\mathbf{k}'}$ et $\Psi^{2,\epsilon}_{\mathbf{j}',\mathbf{k}'}$ définies dans les équations (3.11) et (3.12).

Utilisation des ondelettes périodiques

Comme précédemment, dans le cas des ondelettes périodiques $j_{min} = 0$ et nous considérons $\psi = \psi^1$ (voir proposition 5). Nous avons donc comme base d'ondelettes celle détaillée dans l'équation (4.2) et il faut faire la transformée en ondelettes sur les espaces appropriés pour faire apparaître des matrices de Gram. En se rappelant que le gradient s'écrit

$$\frac{\partial \mathcal{J}_w}{\partial d^{div,\epsilon}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{\text{div}}) = \int_0^1 \int_0^1 2\frac{\tilde{m}(\mathbf{d}^{\text{div}}) - \int_0^x (\rho_1(y) - \rho_0(y))dy}{\tilde{\rho}(\mathbf{d}^{\text{div}}) + t\rho_1(x) + (1-t)\rho_0(x)}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 - \frac{(\tilde{m}(\mathbf{d}^{\text{div}}) - \int_0^x (\rho_1(y) - \rho_0(y))dy)^2}{(\tilde{\rho}(\mathbf{d}^{\text{div}}) + t\rho_1(x) + (1-t)\rho_0(x))^2}(\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 dxdt$$

avec les ondelettes $\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}}$ définies dans l'équation (4.2), on souhaite décomposer $\frac{m}{\rho}$ et $\left(\frac{m}{\rho}\right)^2$ sur

$$\frac{\tilde{m}(\mathbf{d}^{\text{div}}) - \int_0^x (\rho_1(y) - \rho_0(y))dy}{\tilde{\rho}(\mathbf{d}^{\text{div}}) + t\rho_1(x) + (1-t)\rho_0(x)} = \sum_\epsilon \sum_{\mathbf{j}',\mathbf{k}'} d^{2,\epsilon}_{\mathbf{j}',\mathbf{k}'} \left(\Psi^{2,\epsilon}_{\mathbf{j}',\mathbf{k}'}\right)$$

et

$$\frac{(\tilde{m}(\mathbf{d}^{\text{div}}) - \int_0^x (\rho_1(y) - \rho_0(y))dy)^2}{(\tilde{\rho}(\mathbf{d}^{\text{div}}) + t\rho_1(x) + (1-t)\rho_0(x))^2} = \sum_\epsilon \sum_{\mathbf{j}',\mathbf{k}'} d^{1,\epsilon}_{\mathbf{j}',\mathbf{k}'} \left(\Psi^{1,\epsilon}_{\mathbf{j}',\mathbf{k}'}\right)$$

où $\Psi^{1,\epsilon}_{\mathbf{j}',\mathbf{k}'}$ et $\Psi^{2,\epsilon}_{\mathbf{j}',\mathbf{k}'}$ sont définies dans les équations (3.11) et (3.12). On a donc pour le gradient de $\mathcal{J}$

$$\frac{\partial \mathcal{J}_w}{\partial d^{div,\epsilon}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{div}) = 2\sum_{\mathbf{j}',\mathbf{k}'} d^{2,\epsilon}_{\mathbf{j}',\mathbf{k}'} < \left(\Psi^{2,\epsilon}_{\mathbf{j}',\mathbf{k}'}\right), (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^2 > - \sum_{\mathbf{j}',\mathbf{k}'} d^{1,\epsilon}_{\mathbf{j}',\mathbf{k}'} < \left(\Psi^{1,\epsilon}_{\mathbf{j}',\mathbf{k}'}\right), (\Psi^{div,\epsilon}_{\mathbf{j},\mathbf{k}})^1 >,$$

et en particulier pour les différentes valeurs de $\epsilon = 1, 2$ ou 3

$$\frac{\partial \mathcal{J}_w}{\partial d^{div,1}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{div}) = -\sum_{\mathbf{j}',\mathbf{k}'} d^{1,1}_{\mathbf{j}',\mathbf{k}'} < 1 \otimes \psi^0_{j'_2,k'_2}, 1 \otimes (\psi^1_{j_2,k_2})' > .$$

$$\frac{\partial \mathcal{J}_w}{\partial d^{div,2}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{div}) = 2\sum_{\mathbf{j}',\mathbf{k}'} d^{2,2}_{\mathbf{j}',\mathbf{k}'} < \psi^0_{j'_1,k'_1} \otimes 1, (\psi^1_{j_1,k_1})' \otimes 1 > .$$

$$\frac{\partial \mathcal{J}_w}{\partial d^{div,3}_{\mathbf{j},\mathbf{k}}}(\mathbf{d}^{div}) = 2\sum_{\mathbf{j}',\mathbf{k}'} d^{2,3}_{\mathbf{j}',\mathbf{k}'} < \psi^0_{j'_1,k'_1} \otimes \psi^1_{j'_2,k'_2}, (\psi^1_{j_1,k_1})' \otimes \psi^1_{j'_2,k'_2} > - \sum_{\mathbf{j}',\mathbf{k}'} d^{1,3}_{\mathbf{j}',\mathbf{k}'} < \psi^1_{j'_1,k'_1} \otimes \psi^0_{j'_2,k'_2}, \psi^1_{j_1,k_1} \otimes (\psi^1_{j_2,k_2})' > .$$

On dénote par $\mathbf{C}^0$ la matrice qui a pour éléments $\langle \psi^0_{j'_1,k'_1}, (\psi^1_{j_1,k_1})' \rangle$ et $\mathbf{M}$ la matrice de Gram de la base ψ^1. Le calcul de ces deux matrices peut être trouvé dans [50].
On obtient

$$\begin{aligned}
&\text{pour } \epsilon = 1,\ \nabla^1 J_w(\mathbf{d}^{div}) = -\mathbf{d}^{1,1}\mathbf{C}^0,\\
&\text{pour } \epsilon = 2,\ \nabla^2 J_w(\mathbf{d}^{div}) = 2\mathbf{C}^0\mathbf{d}^{2,2},\\
&\text{pour } \epsilon = 3,\ \nabla^3 J_w(\mathbf{d}^{div}) = 2\mathbf{C}^0\mathbf{d}^{2,3}\mathbf{M} - \mathbf{M}\mathbf{d}^{1,3}\mathbf{C}^0,
\end{aligned}$$

et l'algorithme de descente suivant

Algorithme 4 (GDGMP)**.**
Initialisation : $\tau > 0, (\boldsymbol{d}^{div})^0$
Itérations :

$$\begin{aligned}
&\tilde{m}^i = \sum_{\epsilon}\sum_{\boldsymbol{j},\boldsymbol{k}} (d^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^i (\psi^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^2 \ \textit{et}\ \tilde{\rho}^i = \sum_{\epsilon}\sum_{\boldsymbol{j},\boldsymbol{k}} (d^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^i (\Psi^{div,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^1\\
&m^i = \tilde{m}^i - \int_0^x (\rho_1 - \rho_0)dy \ \textit{et}\ \rho^i = \tilde{\rho}^i + t\rho_1 + (1-t)\rho_0\\
&\textit{Calcul des coefficients d'ondelettes } (d^{2,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^i \textit{ de } \frac{m^i}{\rho^i}\\
&\textit{Calcul des coefficients d'ondelettes } (d^{1,\epsilon}_{\boldsymbol{j},\boldsymbol{k}})^i \textit{ de } \left(\frac{m^i}{\rho^i}\right)^2\\
&(\nabla^1 J_w)^i = -(\boldsymbol{d}^{1,1})^i \boldsymbol{C}^0 \ \textit{et}\ (\nabla^2 J_w)^i = 2\boldsymbol{C}^0(\boldsymbol{d}^{2,2})^i\\
&(\nabla^3 J_w)^i = 2\boldsymbol{C}^0(\boldsymbol{d}^{2,3})^i \boldsymbol{M} - \boldsymbol{M}(\boldsymbol{d}^{1,3})^i \boldsymbol{C}^0\\
&(\boldsymbol{d}^{div})^{i+1} = (\boldsymbol{d}^{div})^i - \tau(\nabla J_w)^i.
\end{aligned}$$

Utilisation des ondelettes respectant des conditions de type Dirichlet

Comme dans la section 4.1.2, dans le cas des ondelettes sur $[0,1]^2$, $j_{min} \neq 0$ et nous considérons $\psi = \psi^d$ (voir définition 2). Il faut procéder comme explicité dans la section précédente avec les ondelettes périodiques mais avec la base d'ondelettes définie par l'équation (4.1). Il n'y a pas de simplifications avec cette base, c'est-à-dire qu'aucune des composantes des différentes ondelettes à divergence nulle ne s'annule.

4.2 Algorithme primal-dual

4.2.1 Introduction

Avant de détailler l'application de l'algorithme aux différents cas étudiés dans le chapitre 3, commençons par introduire l'algorithme primal-dual. Pour ce faire, nous avons besoin d'éléments d'analyse convexe qui seront définis ici. Pour plus de détails ou pour aller plus loin on peut lire par exemple [76, 75] ou [6].

Soient X et Y deux espaces vectoriels réels, munis d'un produit scalaire $< .,. >$ et de la norme $||.|| =< .,. >^{1/2}$. On considère un opérateur linéaire continu $K : X \to Y$ et la norme

$$||K|| = \max\{||Kx|| : x \in X, ||x|| \leq 1\}.$$

On dénote par $\Gamma_0(X)$ l'ensemble des fonctions propres, convexes et semi-continues inférieurement sur X. L'algorithme primal-dual permet de résoudre le **problème primal** non linéaire

$$\min_{x \in X} F(Kx) + G(x), \tag{4.3}$$

où F et G appartiennent à $\Gamma_0(X)$. **La transformée de Legendre** de la fonction $F \in \Gamma_0(X)$, est définie par

$$F^*(y) = \max_x \langle x, y \rangle - F(x). \tag{4.4}$$

L'algorithme que l'on considère ne va pas utiliser la formulation (4.3) du problème mais sa **formulation primale-duale**

$$\min_{\phi} \max_{z} \langle K\phi, z \rangle + G(\phi) - F^*(z). \tag{4.5}$$

Définissons à présent **l'opérateur proximal** de F, pour un paramètre $\sigma > 0$

$$\text{prox}_{\sigma F} : \begin{array}{ccc} \mathbb{R}^n & \to & \mathbb{R}^n \\ x & \mapsto & \underset{y}{\text{argmin}} \ \left(\frac{1}{2}|x-y|^2 + \sigma F(y)\right). \end{array}$$

L'application $y \mapsto \frac{1}{2}|x-y|^2 + \sigma F(y)$ est fortement convexe. En effet, une fonction propre f est **fortement convexe** de constante $\beta \in \mathbb{R}^+$ si

$$\forall x \in \text{dom}\, f, \ \forall y \in \text{dom}\, f, \ \forall \alpha \in]0,1[$$
$$f(\alpha x + (1-\alpha)y) + \alpha(1-\alpha)\frac{\beta}{2}||x-y||^2 \leq \alpha f(x) + (1-\alpha)f(y)$$

Une fonction fortement convexe étant strictement convexe, $y \mapsto \frac{1}{2}|x-y|^2 + \sigma F(y)$ ne possède qu'un seul minimum. Par ailleurs, en définissant le **sous-différentiel** ∂F de la fonction F en x par

$$\partial F(x) = \{t \in H | (\forall y \in H) f(y) \leq f(x) + < t, (y-x) >\},$$

une autre définition de l'opérateur proximal s'écrit :

$$\hat{y} = \text{prox}_{\sigma f}(x) \Leftrightarrow x - \hat{y} \in \frac{1}{\sigma}\partial f(\hat{y}) \Leftrightarrow \hat{y} \in (Id + \sigma \partial f)^{-1}(x) \tag{4.6}$$

Un élément de l'ensemble ∂F est appelé sous-gradient de F en x et la formulation ci-dessus est caractéristique des algorithmes de sous-gradient.

L'opérateur proximal admet plusieurs propriétés qui, généralement, permettent de le calculer aisément, voir [22] :

- Si $F \in \Gamma_0(X)$ est une fonction positivement homogène alors c'est la fonction support d'un ensemble convexe fermé et non vide et sa transformée de Legendre est la fonction indicatrice de cet ensemble dans le sens :

$$\iota_E(x) = \begin{cases} 0 \text{ si } x \in E \\ +\infty \text{ sinon.} \end{cases}$$

- L'opérateur proximal de la fonction indicatrice d'un ensemble non vide, fermé et convexe est la projection sur cet ensemble.
- L'opérateur proximal de la transformée de Legendre d'une fonction $F \in \Gamma_0(X)$ peut être calculé à l'aide de l'identité de Moreau [67], pour $x \in X$

$$x = \text{prox}_{\sigma F^*}(x) + \sigma \,\text{prox}_{F/\sigma}(x/\sigma). \tag{4.7}$$

  On peut donc calculer simplement l'opérateur proximal de F en fonction de celui de F^*, et inversement. On suppose que F et G sont **simples** c'est-à-dire que leurs opérateurs proximaux peuvent se calculer "aisément".

Finalement, **l'algorithme primal-dual** ([16]) que l'on va considérer par la suite est le suivant :

Algorithme 5.

Initialisation : $\tau, \sigma > 0, \theta \in [0,1],\ (\phi^0, z^0 = K\phi^0, \tilde{\phi}^0 = \phi^0)$.

Itérations :

$$\begin{aligned} z^{i+1} &= \text{prox}_{g^*}(z^i + \sigma(K\tilde{\phi}^i)) \\ \phi^{i+1} &= \text{prox}_{\iota_{C_0}}(\phi^i - \tau K^* z^{i+1}) \\ \tilde{\phi}^{i+1} &= \phi^{i+1} + \theta(\phi^{i+1} - \phi^i). \end{aligned}$$

— La première étape peut être réecrite

$$\begin{aligned} z^{i+1} &= \text{prox}_{g^*}(z^i + \sigma(K\tilde{\phi}^i)) \\ \Leftrightarrow (Id + \sigma\partial J^*)z^{i+1} &= (z^i + \sigma(K\tilde{\phi}^i)) \\ \Leftrightarrow z^{i+1} &= z^i + \sigma(K\tilde{\phi}^i - \partial J^*(z^{i+1})) \end{aligned}$$

ce qui correspond à un algorithme de sous-gradient (4.6) permettant de résoudre

$$\max_z \ \langle K\phi, z\rangle - J^*(z),$$

où $z \mapsto \langle K\phi, z\rangle - J^*(z)$ est concave car J^* est convexe.

— La seconde étape s'écrit

$$\begin{aligned} \phi^{i+1} &= \text{prox}_{\iota_{C_0}}(\phi^i - \tau K^* z^{i+1}) \\ \Leftrightarrow (Id + \sigma\partial\iota_{C_0})\phi^{i+1} &= (\phi^i - \tau K^* z^{i+1}) \\ \Leftrightarrow \phi^{i+1} &= \phi^i - \tau(K^* z^{i+1} + \partial\iota_{C_0})) \end{aligned}$$

ce qui correspond à un algorithme de sous-gradient (4.6) permettant de résoudre

$$\min_{\phi} \langle \phi, K^* z \rangle + \iota_{C_0}(\phi),$$

où $\phi \mapsto \langle \phi, K^* z \rangle + \iota_{C_0}(\phi)$ est convexe car ι_{C_0} est convexe.
L'algorithme ci-dessus résout donc bien l'équation primale duale (4.5).

Convergence de l'algorithme

On suppose qu'il existe une solution $(\hat{\phi}, \hat{z})$ au problème (4.5). Il a été prouvé [15, 16, 45] que pour $\theta = 1$ et $\sigma\tau||K||^2 < 1$, le couple (ϕ^i, z^i) calculé avec l'algorithme 5, converge vers la solution $(\hat{\phi}, \hat{z})$ de (4.5) si X et Y sont de dimension finie.
Nous avons donc la convergence de l'algorithme pour le problème discrétisé qui sera détaillé pour chaque cas dans les paragraphes suivants.

4.2.2 Application au cas $(\rho, m) = \nabla \times \phi$

Dans un premier temps, nous considérons le cas $(\rho, m) = \nabla \times \phi$ (section 3.2), dans la formulation de type surface minimale (3.22) puis dans la fonctionnelle $\mathcal{J}$ (3.18). Ces deux formulations ne sont valables que pour la dimension 1 en espace.

Minimisation de la formulation de type surface minimale

Nous détaillons tout d'abord la minimisation de la formulation de type surface minimale quand on utilise pour ρ une formulation fonction de courant (3.22)

$$\min_{\phi} \mathcal{J}_{ms}(\nabla\phi) + \iota_{C_\phi}(\phi) = \min_{\phi} ||\nabla\phi||_1 + \iota_{C_\phi}(\phi) \tag{4.8}$$

où on considère $K = \nabla$, le gradient, qui est bien linéaire et continu de $H^1(Q)$ dans $(L^2(Q))^2$, la fonctions $\mathcal{J}_{ms}(y) = ||y||_1$ et $G(y) = \iota_{C_\phi}(y)$, la fonction indicatrice de l'ensemble des conditions sur le bord pour ϕ

$$C_\phi = \{\phi,\ \phi(0,x) = \int_0^x \rho_0(s)ds = R_0(x),\ \phi(1,x) = \int_0^x \rho_1(s)ds = R_1(x),$$
$$\phi(t,0) = R_1(0) = R_0(0) = 0,\ \phi(t,1) = R_1(1) = R_0(1) = 1\}. \tag{4.9}$$

qui sont propres, convexes et semi-continues inferieurement dans $(L^2(Q))^2$ et $H^1(Q)$ respectivement. La formulation primale-duale associée à cette minimisation est

$$\min_{\phi \in H^1(Q)} \max_{z \in (L^2(Q))^2} \langle z.\nabla\phi \rangle + \iota_{C_\phi}(\phi) - \mathcal{J}^*_{ms}(z).$$

Nous allons avoir besoin dès à présent de définir le cadre discret dans lequel nous utilisons l'algorithme. Sur la grille discrète, G^c, définie dans la section 5.1, $||.||_1$ s'écrit

$$||x||_1 = \sum_{k \in G^c} |x_k| \text{ où } |x_k| = \sqrt{(x_k^1)^2 + (x_k^2)^2},$$

par conséquent nous avons,

$$\text{prox}_{\sigma \mathcal{J}_{ms}}(x) = \text{prox}_{\sigma||.||_1}(x) = (\text{prox}_{\sigma|.|}\, x_k)_{k \in G^c}.$$

L'opérateur proximal de la transformée de Legendre de la fonction $J_{ms} = |.|$ est le suivant, pour tout $c \in \mathbb{R}^2$

$$\text{prox}_{\sigma \mathcal{J}^*_{ms}}(c) = \min(c, \frac{c}{|c|}),$$

ce qui est détaillé dans l'appendice A.1. Ainsi $\text{prox}_{\sigma \mathcal{J}^*_{ms}}$ est une projection euclidienne point par point sur la boule unité de centre 0 de $\mathbb{R}^2$. Comme ι_{C_ϕ} est la fonction indicatrice d'un ensemble fermé, non vide et convexe, son opérateur proximal est la projection sur l'ensemble C_ϕ, que l'on va dénoter par la suite $\mathcal{P}_{C_\phi}$. On obtient donc dans ce cas l'algorithme suivant :

Algorithme 6 (PDMSSF)**.**

Initialisation : $\tau, \sigma > 0, \theta \in [0,1],\ (\phi^0, z^0 = \nabla\phi^0, \tilde{\phi^0} = \phi^0)$.

Itérations :

$$\begin{aligned} z^{i+1} &= \text{prox}_{\sigma \mathcal{J}^*_{ms}}(z^i + \sigma(\nabla \bar{\phi}^i)) \\ \phi^{i+1} &= \mathcal{P}_{C_\phi}(\phi^i - \tau \nabla^*.z^{i+1}) \\ \bar{\phi}^{i+1} &= \phi^{i+1} + \theta(\phi^{i+1} - \phi^i). \end{aligned}$$

Il a été prouvé dans [15] que pour $\theta = 1$ et $\sigma\tau||\nabla||^2 < 1$, ϕ^i calculé à l'aide de l'algorithme ci-dessus, converge vers la solution du problème discrétisé :

$$\min_{\phi} \sum_{k \in G^c} J_{ms}(\nabla \phi_k) + \iota_{C_\phi}(\phi_k) \tag{4.10}$$

Une analyse de cet algorithme nous dit que :

- la première étape consiste en une projection point par point sur la boule unité de centre 0 de $\mathbb{R}^2$,
- la deuxième étape consiste à appliquer les bonnes conditions sur le bord.

Minimisation de la fonctionnelle J

Nous voulons à présent minimiser (3.18)

$$\mathcal{J}(\nabla \times \phi) = \int_0^1 \int_\Omega J(\nabla \times \phi(t,x)) dx dt$$

dont la formulation primale-duale est :

$$\min_{\phi} \max_{z} \langle K\phi, z \rangle + \iota_{C_\phi}(\phi) - \mathcal{J}^*(z), \tag{4.11}$$

où $K = \nabla\times$, l'opérateur rotationnel, est un opérateur linéaire et continu de $H^1(Q)$ dans $(L^2(Q))^2$, $\mathcal{J}^* : (L^2(Q))^2 \to [0, +\infty)$, la transformée de Legendre de $\mathcal{J}$ et $\iota_{C_\phi} : (H^1(Q)) \to [0, +\infty)$, la fonction indicatrice de l'ensemble C_ϕ, qui sont propres, convexes et semi-continues inferieurement. La fonctionnelle discrète $\mathcal{J}$ s'écrit pour (ρ, m) discrétisé sur la grille centrée G^c définie dans la section 5.1 :

$$\mathcal{J}(\rho, m) = \sum_{k\in G^c} J(\rho_k, m_k), \tag{4.12}$$

où la fonctionnelle J est définie par (2.16), et alors,

$$\text{prox}_{\gamma\mathcal{J}}(x) = (\text{prox}_{\gamma J}(x_k))_{k\in G^c}.$$

Comme $\mathcal{J}$ est positivement homogène, il a été prouvé [7, 69] que la transformée de Legendre de $\mathcal{J}$ est la fonction indicatrice d'un ensemble convexe, $\mathcal{J}^* = i_{P_\mathcal{J}}$ où

$$\begin{cases} P_\mathcal{J} = \{(z_1, z_2);\ \forall k \in G^c,\ (z_1, z_2)_k \in P_J\} \\ P_J = \{(t, x) \in \mathbb{R}\times\mathbb{R}^n,\ t + \frac{|x|^2}{2} \le 0\}, \end{cases} \tag{4.13}$$

La preuve figure dans l'annexe A.2.
Cela implique que $\text{prox}_{\gamma J^*}$ est la projection sur le paraboloide P_J, que l'on va dénoter $\mathcal{P}_{P_J}$.

Proposition 10. *La projection sur le paraboloide $P_J = \{(a, b) \in \mathbb{R}\times\mathbb{R},\ a + \frac{b^2}{2} = 0\}$ d'un point à l'extérieur de P_J est*

$$\mathcal{P}_{P_J}(z_1, z_2) = \left(\frac{-\tilde{b}^2}{2}, \tilde{b}\right)$$

où $\tilde{b}$ est la solution réelle de l'équation cubique en X

$$C(X) = \frac{-X^3}{2} - (1 + z_1)X + z_2$$

dont le signe est le même que celui de z_2.

Démonstration. Pour expliciter la projection $P_J = \{(a, b) \in \mathbb{R}\times\mathbb{R},\ a + \frac{b^2}{2} = 0\}$, nous dénotons la projection de (z_1, z_2) sur le paraboloide par $(\tilde{a}, \tilde{b})$.

En remarquant que la normale au paraboloide est le vecteur $(1, \tilde{b})$ au point $(\tilde{a}, \tilde{b})$, nous obtenons l'équation suivante

$$\begin{aligned} &(\tilde{a}, \tilde{b}) - (z_1, z_2) \ //\ (1, \tilde{b}) \\ \Leftrightarrow\ &(\tilde{a} - z_1)\tilde{b} - (\tilde{b} - z_2) = 0. \end{aligned}$$

Donc nous avons le système suivant :

$$\begin{cases} (\tilde{a} - z_1)\tilde{b} - (\tilde{b} - z_2) &= 0 \\ \tilde{a} + \frac{\tilde{b}^2}{2} &= 0 \end{cases} \Leftrightarrow \begin{cases} (-\frac{\tilde{b}^2}{2} - z_1)\tilde{b} - (\tilde{b} - z_2) &= 0 \\ \tilde{a} &= -\frac{\tilde{b}^2}{2} \end{cases}$$

$$\Leftrightarrow \begin{cases} -\frac{\tilde{b}^3}{2} - z_1\tilde{b} - \tilde{b} + z_2 & = 0 \\ \tilde{a} & = -\frac{\tilde{b}^2}{2} \end{cases} \Leftrightarrow \begin{cases} \frac{-\tilde{b}^3}{2} - (1+z_1)\tilde{b} + z_2 & = 0 \\ \tilde{a} & = -\frac{\tilde{b}^2}{2} \end{cases} \tag{4.14}$$

Il faut ensuite résoudre cette équation cubique (4.14) en utilisant la méthode de Cardan. La solution qui nous intéresse est la solution réelle du même signe que z_2. □

Remarque. L'intérêt de cette projection est son exactitude et le fait que c'est une méthode directe contrairement à d'autres méthodes (méthode de Newton,...).

Nous obtenons finalement l'algorithme

Algorithme 7 (PDSF)**.**

Initialisation : $\tau, \sigma > 0, \theta \in [0,1]$, $(\phi^0, z^0 = \nabla \times \phi^0, \tilde{\phi}^0 = \phi^0)$.

Itérations :

$$\begin{aligned} z^{i+1} &= \mathcal{P}_{P_J}(z^i + \sigma(\nabla \times \tilde{\phi}^i)) \\ \phi^{i+1} &= \mathcal{P}_{C_\phi}(\phi^i - \tau\nabla^* \times z^{i+1}) \\ \tilde{\phi}^{i+1} &= \phi^{i+1} + \theta(\phi^{i+1} - \phi^i). \end{aligned}$$

Remarques.

- Comme précédemment, pour $\theta = 1$ et $\sigma\tau||\nabla \times ||^2 < 1$, ϕ^i calculé à l'aide de l'algorithme ci-dessus, converge vers la solution du problème discrétisé de minimisation de (4.12).
- Le calcul de $\mathcal{P}_{P_J}$ revient à la résolution d'une équation du troisième ordre en chaque point de grille, alors que $\mathcal{P}_{C_\phi}$ correspond simplement à appliquer les bonnes conditions sur le bord.
- En dimension, 2 il y a deux opérateurs rotationnels adjoints l'un de l'autre, l'un est vectoriel $\nabla \times \phi = (\partial_x\phi, -\partial_t\phi)$ et l'autre scalaire $\nabla^* \times (\phi_1, \phi_2) = \partial_x\phi_1 - \partial_t\phi_2$.

4.2.3 Application au cas $(\rho, m) = \nabla \times \phi + \nabla h$

Nous considérons à présent le cas $(\rho, m) = \nabla \times \phi + \nabla h$ (section 3.3), pour la formulation de type surface minimale (3.32) puis la fonctionnelle $\mathcal{J}$ (3.27). La première formulation est valable en dimension 1 en espace seulement alors que la seconde est valable en dimension 1 ou 2 en espace.

Minimisation de la formulation surface minimale (dimension 1 en espace)

Nous voulons à présent résoudre l'équation (3.32)

$$\min_\phi \mathcal{J}_{ms,h}(\nabla\phi) + \iota_{C_0}(\phi) = \min_\phi ||\nabla\phi - \nabla \times h||_1 + \iota_{C_0}(\phi) \tag{4.15}$$

où on prend $K = \nabla$, le gradient, comme dans le premier paragraphe de la section 4.2.2 et les fonctions $\mathcal{J}_{ms,h}(y) = ||y - \nabla \times h||_1$ et $G(y) = \iota_{C_0}(y)$, la fonction indicatrice de

l'ensemble des conditions sur le bord $C_0 = \{\phi,\ \phi|_{\partial Q} = 0\}$, qui sont propres, convexes et semi-continues inferieurement dans $(L^2(Q))^2$ et $H^1(Q)$ respectivement. Sa formulation primale-duale est

$$\min_{\phi} \max_{z} \langle z.\nabla\phi\rangle + \iota_{C_0}(\phi) - \mathcal{J}^*_{ms,h}(z).$$

Pour $(a,b) = \nabla\phi$:

$$\mathcal{J}_{ms,h}(a,b) = ||(a - \partial_x h, b + \partial_t h)||_1 = \sum_{k\in G^c} J_{ms,h}(a_k,b_k) = \sum_{k\in G^c} |(a_k - \partial_x h_k, b_k + \partial_t h_k)|.$$

En notant $c = (a,b)$ on obtient la proposition suivante :

Proposition 11. *Pour tout $c \in \mathbb{R}^{1+n}$, soit*

$$J_\lambda(c) = J(c+\lambda),$$

la transformée de Legendre de J_λ et son opérateur proximal s'écrivent

$$J^*_\lambda(c) = J^*(c) - \langle\lambda, c\rangle,$$

et

$$\text{prox}_{\gamma J^*_\lambda}(c) = \text{prox}_{\gamma J^*}(c + \gamma\lambda).$$

Démonstration.

$$\begin{aligned} J^*_\lambda(c) &= \max_x \langle x,c\rangle - J_\lambda(x) \\ &= \max_x \langle x,c\rangle - J(x+\lambda) \\ &= \max_x \langle x-\lambda,c\rangle - J(x) \\ &= J^*(c) - \langle\lambda,c\rangle, \end{aligned}$$

et

$$\begin{aligned} \text{prox}_{\gamma J^*_\lambda}(c) &= \operatorname*{argmin}_x \left[\frac{1}{2}|x-c|^2 + \gamma J^*_\lambda(x)\right] \\ &= \operatorname*{argmin}_x \left[\frac{1}{2}|x-c|^2 + \gamma(J^*(x) - \langle\lambda,x\rangle)\right] \\ &= \operatorname*{argmin}_x \left[\frac{1}{2}|x-\gamma\lambda-c|^2 + \gamma J^*(x)\right] \\ &= \text{prox}_{\gamma J^*}(c+\gamma\lambda). \end{aligned}$$

□

On peut à présent prouver la propostion suivante :

Proposition 12. *Pour tout $c \in \mathbb{R}^2$, l'opérateur proximal de $\mathcal{J}^*_{ms,h}$ est*

$$\text{prox}_{\gamma \mathcal{J}^*_{ms,h}}(c) = \min(c - \gamma\nabla \times h, \frac{(c - \gamma\nabla \times h)}{|c - \gamma\nabla \times h|}),$$

On observe alors que $prox_{\gamma \mathcal{J}^*_{ms,h}}$ est la projection euclidienne point par point sur une boule unité de $\mathbb{R}^2$.

Démonstration. D'après l'appendice A.1, $\text{prox}_{\gamma \mathcal{J}^*_{ms}}(c) = \min\left(c, \frac{c}{|c|}\right)$. On obtient donc en utilisant la proposition précédente :

$$\text{prox}_{\gamma \mathcal{J}^*_{ms,h}}(c) = \text{prox}_{\gamma \mathcal{J}^*_{ms}}(c + \gamma(-\nabla \times h)) = \min(c - \gamma\nabla \times h, \frac{(c - \gamma\nabla \times h)}{|c - \gamma\nabla \times h|})$$

□

Etant donné que ι_{C_0} est la fonction indicatrice d'un ensemble non vide, fermé et convexe, son opérateur est la projection sur l'ensemble C_0, que l'on va dénoter $\mathcal{P}_{C_0}$.
On obtient donc dans ce cas l'algorithme suivant

Algorithme 8 (PDMSHH)**.**

Initialisation : $\tau, \sigma > 0, \theta \in [0, 1]$, $(\phi^0, z^0 = \nabla\phi^0, \tilde{\phi^0} = \phi^0)$.

Itérations :

$$\begin{aligned} z^{i+1} &= \text{prox}_{\sigma|.|}(z^i + \sigma(\nabla\bar{\phi}^i - \nabla \times h)) \\ \phi^{i+1} &= \mathcal{P}_{C_0}(\phi^i - \tau\nabla^*.z^{i+1}) \\ \bar{\phi}^{i+1} &= \phi^{i+1} + \theta(\phi^{i+1} - \phi^i). \end{aligned}$$

Pour $\theta = 1$ et $\sigma\tau||\nabla||^2 < 1$, ϕ^i converge vers la solution du problème (4.15) discrétisé :

$$\min_{\phi} \sum_{k \in G^c} J_{ms,h}(\nabla\phi_k) + \iota_{C_0}(\phi_k), \tag{4.16}$$

et comme précédemment, les deux étapes de l'algorithme sont des projections point par point.

Minimisation de la fonctionnelle J (dimension 1 ou 2 en espace)

Nous appliquons à présent l'algorithme à (3.27) reformulée :

$$\min_{\phi} \max_{z} \langle K\phi, z\rangle + \iota_{C_0}(\phi) - \mathcal{J}^*_h(z), \tag{4.17}$$

où

— $K = \nabla\times$, le rotationnel de $H^1(Q)$ dans $(L^2(Q))^2$, $\mathcal{J}^*_h : (L^2(Q))^2 \to [0, +\infty)$ en dimension 1,

— $K = \nabla\times$, le rotationnel de $(H^1(Q))^3$ dans $(L^2(Q))^3$, $\mathcal{J}_h^* : (L^2(Q))^3 \to [0, +\infty)$ en dimension 2.

Comme rappelé dans le paragraphe 2 de la section 4.2.2, la transformée de Legendre de $\mathcal{J}$ est la fonction indicatrice d'un paraboloide et donc $\text{prox}_{\gamma J^*}$ est la projection sur ce paraboloide que l'on explicite comme suit :

Proposition 13. *La projection sur le paraboloide* $P_J = \{(a, b) \in \mathbb{R} \times \mathbb{R}^2,\ a + \frac{|b|^2}{2} = 0\}$, *d'un point à l'exterieure du paraboloide* P_J, *s'écrit*

$$\mathcal{P}_{P_J}(z_1, z_2) = \left(\frac{-\tilde{b}^2}{2}, \frac{z_2}{1 + z_1 + \tilde{b}^2/2} \right),$$

où $\tilde{b}$ *est la solution réelle positive de l'équation de degré trois en* X

$$C(X) = \frac{-X^3}{2} - (1 + z_1)X + |z_2|.$$

Démonstration. Si on appelle $(\tilde{a}, \tilde{b})$ la projection de (z_1, z_2) sur le paraboloide, comme (z_1, z_2) est sur la normale au paraboloide en $(\tilde{a}, \tilde{b})$, c'est-à-dire $(1, \tilde{b})$, on a

$$(z_1, z_2) = \lambda(1, \tilde{b}) + (\tilde{a}, \tilde{b})$$
$$(z_1 - \tilde{a}, z_2 - \tilde{b}) = \lambda(1, \tilde{b}).$$

Donc on obtient $\lambda = z_1 - \tilde{a}$ et pour la projection en 3 dimensions on doit seulement projeter $(z_1, |z_2|)$ sur le paraboloide en deux dimensions qui est l'intersection du paraboloide en trois dimensions et du plan qui contient l'axe de la variable temporelle et (z_1, z_2). On a donc

$$\mathcal{P}_{P_J}(z_1, |z_2|) = \left(\frac{-\beta^2}{2}, \beta \right)$$

où β est la solution réelle positive de l'équation cubique en X, $C(X) = \frac{-X^3}{2} - (1+z_1)X + |z_2| = 0$ et on obtient que $\tilde{a} = -\beta^2/2$. De (4.14), on déduit que

$$\lambda = z_1 - \tilde{a} = z_1 + \beta^2/2 \text{ et } \tilde{b} = \frac{z_2}{1 + \lambda} = \frac{z_2}{1 + z_1 + \beta^2/2}.$$

Il faut maintenant vérifier que l'on a bien $1 + \lambda \neq 0$. Si $1 + \lambda = 0$ alors, de (4.14) on déduit

$$z_2 - \tilde{b} = -\tilde{b} \text{ et donc } z_2 = 0.$$

La paire (z_1, z_2) est en dehors du paraboloide P_J par hypothèse donc $z_1 + |z_2|^2/2 > 0$ et dans ce cas $z_1 > 0$. De façon similaire, la paire $(\tilde{a}, \tilde{b})$ dans le paraboloide P_J donc $\tilde{a} \leq 0$. Mais de (4.14) on a que

$$z_1 = \tilde{a} + \lambda = \tilde{a} - 1 < 0$$

ce qui n'est pas possible. Donc $1 + \lambda$ n'est pas nul. □

Remarque. Comme précédemment, l'intérêt de cette projection est son exactitude et le fait que c'est une méthode directe.

On peut définir la nouvelle fonctionnelle discrète, pour $(a, b) = \nabla \times \phi$:

$$\mathcal{J}_h(a, b) = \mathcal{J}(a + \partial_t h, b + \nabla_x h) = \sum_{k \in G^c} J_h(a_k, b_k) = \sum_{k \in G^c} J(a_k + \partial_t h_k, b_k + \nabla_x h_k).$$

Cette nouvelle fonctionnelle étant écrite en fonction de J, on peut déduire de J^* la forme de J_h^* et la forme de $\text{prox}_{\gamma J_h^*}$ de celle de $\text{prox}_{\gamma J^*}$. En effet, l'opérateur proximal de J^* a été calculé dans la section 4.2.2 et est la projection sur la praboloide P_J (4.13), $\mathcal{P}_{P_J}$. D'après la proposition 11, l'opérateur proximal de J_h^* s'écrit :

$$\text{prox}_{\gamma \mathcal{J}_h^*}(c) = \text{prox}_{\gamma \mathcal{J}^*}(c + \gamma(\nabla h)) = \mathcal{P}_{P_J}(c + \gamma(\nabla h))$$

Nous obtenons finalement l'algorithme primal-dual suivant :

Algorithme 9 (PDHH)**.**

Initialisation : $\tau, \sigma > 0, \theta \in [0, 1],\ (\phi^0, z^0 = \nabla \times \phi^0, \tilde{\phi}^0 = \phi^0)$.

Itérations :

$$\begin{aligned} z^{i+1} &= \mathcal{P}_{P_J}(z^i + \sigma(\nabla \times \tilde{\phi}^i + \nabla h)) \\ \phi^{i+1} &= \mathcal{P}_{C_0}(\phi^i - \tau \nabla^* \times z^{i+1}) \\ \tilde{\phi}^{i+1} &= \phi^{i+1} + \theta(\phi^{i+1} - \phi^i). \end{aligned}$$

Comme précédemment, le calcul de la projection $\mathcal{P}_{P_J}$ est la résolution d'une équation du troisième degré en chaque point de la grille, alors que $\mathcal{P}_{C_0}$ est la mise à zéro des valeurs sur le bord.

Remarque. Analytiquement l'opérateur rotationnel est auto-adjoint. Cependant, selon la discrétisation utilisée il peut ne pas l'être. C'est pourquoi nous garderons dans les notations ∇ et ∇^*.

4.2.4 Autres applications

Changement de la fonctionnelle

Dans [82], Valkonen propose d'étendre l'algorithme primal-dual dans le cas où K est de classe $\mathcal{C}^2$ peut être **non linéaire**. Si on considère l'algorithme 9 (PDHH), l'idée est la suivante. Au lieu de minimiser $\mathcal{J}_h(K\phi)$ avec $K\phi = \nabla \times \phi$, on minimise $\mathcal{J}(K\phi)$ avec $K\phi = \nabla \times \phi + \nabla h$, l'étape faisant intervenir l'adjoint étant opéré avec le linéarisé de K. C'est-à-dire :

Algorithme primal-dual	**Algorithme primal-dual non linéaire**
$\min_{\phi} \mathcal{J}_h(K\phi) + \iota_{C_0}(\phi)$	$\min_{\phi} \mathcal{J}(K\phi) + \iota_{C_0}(\phi)$
où $K\phi = \nabla \times \phi$	où $K\phi = \nabla \times \phi + \nabla h$

Le nouvel algorithme obtenu s'écrit :

Algorithme 10 (PDNL)**.**

Initialisation : $\tau, \sigma, \theta,\ (\phi^0, z^0 = K\phi^0)$.

Itérations :

$$\begin{aligned}
\phi^{i+1} &:= \text{prox}_{\tau\iota_{C_0}}(\phi^i - \tau[\nabla K(\phi^i)]^* z^i),\\
\tilde{\phi}^{i+1} &:= \phi^{i+1} + \theta(\phi^{i+1} - \phi^i),\\
z^{i+1} &:= \text{prox}_{\sigma\mathcal{J}^*}(z^i + \sigma K(\tilde{\phi}^{i+1})),
\end{aligned}$$

où $[\nabla K]$ est la linéarisation de l'opérateur K. Notre opérateur K étant affine, sa linéarisation est simplement $[\nabla K] = \nabla\times$. On observe donc que l'algorithme 9 est le même que celui que nous venons de décrire. Les deux méthodes sont donc équivalentes dans notre cas.

Dans le cas général, en notant $\hat{\phi}, \hat{z}$ le point selle du problème primal-dual que l'on considère, pour pouvoir utiliser l'algorithme PDNL il faut vérifier les deux proprétés, non triviales, suivantes :

- Soit

$$H_{\hat{\phi}}(\phi, z) = \begin{pmatrix} \partial\iota_{C_0}(\phi) + [\nabla K(\hat{\phi})]^* z \\ \partial\mathcal{J}^*(z) - [\nabla K(\hat{\phi})]\phi - c_{\hat{\phi}} \end{pmatrix},$$

  où $c_{\hat{\phi}} := [\nabla K(\hat{\phi})]\hat{\phi} - K(\hat{\phi})$. L'inverse de $H_{\hat{\phi}}$ doit être pseudo-Lipschitz, ce que l'on appelle la propriété de Aubin (voir remarque ci-dessous).
- La variable duale $\hat{z}$ doit être petite dans le voisinage des points où K n'est pas linéaire.

Remarque. On dit que S vérifie la propriété de Aubin en $\hat{w}$ pour $\hat{u}$ si le graphe de S est localement fermé et s'il existe $\delta,\ \rho$ et l strictement positifs tels que

$$\inf_{v, w\in S(v)} ||u - v|| \leq l||w - S(u)||,\ ||u - \hat{u}|| \leq \delta,\ ||w - \hat{w}|| \leq \rho.$$

Changement de l'opérateur

Dans le cadre de l'utilisation des variables de Clebsch, il faudrait utiliser l'algorithme PDNL, en minimisant

$$\min_{\gamma} \mathcal{J}(K\gamma) + \iota_{C_\gamma}(\gamma)$$

avec

$$(\rho, m) = K\gamma = \nabla\gamma_1 \times \nabla\gamma_2 \times ... \times \nabla\gamma_{n-1}, \tag{4.18}$$

en dimension n, pour un ensemble de contraintes C_γ bien choisi.

Chapitre 5

Applications Numériques

Nous présentons dans ce chapitre quelques applications numériques des différents algorithmes développés au chapitre précédent. Pour cela nous commençons par détailler le cadre discret dans lequel nous effectuons nos implémentations. Ensuite, nous analysons les implémentations développées pour des images 1D. Nous commençons par une première validation d'un algorithme utilisant les ondelettes périodiques. Puis, nous appliquons deux formulations utilisant la minimisation de la fonctionnelle issue du transport optimal $\mathcal{J}$ à l'aide de la représentation par fonction de courant ainsi que par la décomposition de Helmholtz-Hodge. Enfin, deux autres formulations utilisent ces mêmes décompositions mais dans des formulations de type courbure minimale. Nous montrons que toutes ces formulations sont plus performantes que l'état de l'art. Dans le cas d'algorithmes pour des images 2D, nous comparons la minimisation de la fonctionnelle $\mathcal{J}$ utilisant la décomposition de Helmholtz-Hodge à l'état de l'art et montrons les performances de nos implémentations Matlab et C++. Nous finissons ce chapitre par deux applications à des images réelles.

5.1 Description du cadre discret

Nous détaillons dans la suite le cadre discret de l'implémentation des algorithmes développés dans la section 4.2.

5.1.1 Discrétisation pour des images 1D

Nous commençons par décrire les grilles discrètes utilisées pour l'implémentation des différents algorithmes développés pour des images 1D.

Grilles centrées.
L'évaluation de la variable duale $z(t,x)$ est effectuée sur une grille régulière G^{c1} de taille $M \times N$, alors que celle de la variable primale $\phi(t,x)$ est effectuée sur une grille régulière

G^{c2} de taille $(M+1)\times(N+1)$. Ces grilles sont définies par

$$\begin{aligned} G^{c1} &= \{t_i, x_j\}_{1\le i\le M,\ 1\le j\le N}, \\ G^{c2} &= \{t_{i-1/2}, x_{j-1/2}\}_{1\le i\le M+1,\ 1\le j\le N+1}, \end{aligned} \tag{5.1}$$

avec $t_i = \frac{i}{M+1}$, $t_{i-1/2} = \frac{i-1/2}{M+1}$, $x_j = \frac{j}{N+1}$, $x_{j-1/2} = \frac{j-1/2}{N+1}$ les points de discrétisation de $Q = (0,1)\times(0,1)$.

L'introduction de la grille centrée G^{c2} s'explique par la contrainte de divergence nulle du vecteur (ρ, m). En effet, dans le cas de signaux 1D, $(\rho, m) = \nabla\times\phi$ ou $(\rho, m) = \nabla\times\phi + \nabla h$ et nous utilisons des grilles décalées décrites dans le paragraphe suivant pour respecter la contrainte de divergence nulle. Il faut donc placer ϕ sur une grille qui permette au rotationnel d'être sur la grille décalée, cette grille est G^{c2}.

Grilles décalées.
La propriété la plus importante du couple (ρ, m) est qu'il est à divergence nulle. Il est donc naturel d'utiliser une grille décalée pour y définir ρ et m, et ainsi vérifier l'équation de continuité aux points de la grille centrée. Nous définissons cette grille, G^{s1}, par :

$$\begin{aligned} G^{s1}_t &= \{t_{i-1/2},\ x_j\}_{1\le i\le M+1,\ 1\le j\le N}, \\ G^{s1}_x &= \{t_i,\ x_{j-1/2}\}_{1\le i\le M,\ 1\le j\le N+1}. \end{aligned}$$

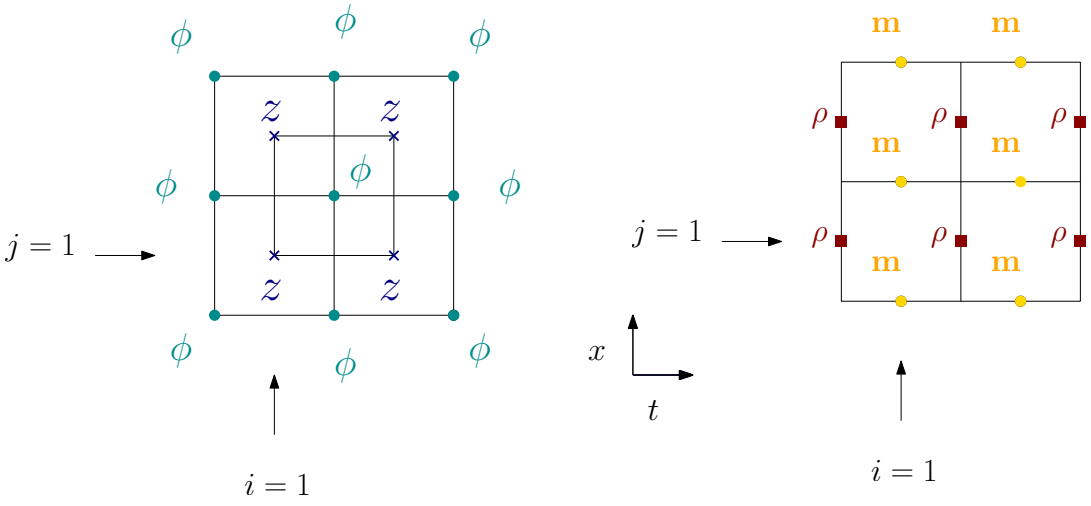


FIGURE 5.1: Répartition des valeurs de ϕ sur la grille centrée G^{c2}, de z sur G^{c1} et de (ρ, m) sur les grilles décalées G^{s1}.

Nous noterons dans la suite la valeur d'une fonction u au point (t_i, x_j) par $u(t_i, x_j) = u_{i,j}$.

Interpolation.
Pour évaluer les valeurs d'un vecteur $u = (u^1, u^2)$ sur la grille centrée G^{c1} à partir de ses valeurs, $\bar{u}$, données sur la grille décalée G^{s1}, on a besoin d'un opérateur d'interpolation, dont la première composante est donnée par :

$$\forall\ 1 \leq i \leq M,\ \forall\ 1 \leq j \leq M, \quad u^1_{i,j} = \frac{1}{2}(\bar{u}^1_{i+1/2,j} + \bar{u}^1_{i-1/2,j}),$$

la seconde par

$$\forall\ 1 \leq i \leq M,\ \forall\ 1 \leq j \leq M, \quad u^2_{i,j} = \frac{1}{2}(\bar{u}^2_{i,j+1/2} + \bar{u}^2_{i,j-1/2}),$$

et d'un opérateur adjoint évaluant les valeurs sur la grille G^{s1} d'une fonction scalaire v sur G^{c1} :

$$\bar{v}^1_{i-1/2,j} = \begin{cases} v_{1,j} & \text{si } i = 1 \\ v_{i,j} + v_{i-1,j} & \text{si } 2 \leq i \leq M \\ v_{M,j} & \text{si } i = M+1. \end{cases}$$

et

$$\bar{v}^2_{i,j-1/2} = \begin{cases} v_{i,1} & \text{si } j = 1 \\ v_{i,j} + v_{i,j-1} & \text{si } 2 \leq j \leq N \\ v_{i,N} & \text{si } j = N+1. \end{cases}$$

Les opérateurs gradient, divergence et rotationnel.
Pour discrétiser les opérateurs divergence, gradient et rotationnel, on utilise les différences finies centrées.

Le **gradient** discret, qui est une matrice de vecteurs, et l'opérateur divergence, qui est son adjoint, sont

$$\nabla = \begin{pmatrix} \partial_t \\ \partial_x \end{pmatrix} \text{ et } \nabla^*. = \begin{pmatrix} \partial_t^* \\ \partial_x^* \end{pmatrix}. = \partial_t^* + \partial_x^*.$$

L'opérateur discret de dérivée partielle par rapport à la première composante s'écrit $\partial_t : G^{c1} \to G^{s1}_t$, pour $v \in G^{c1}$:

$$(\partial_t v)_{i+1/2,j} = v_{i+1,j} - v_{i,j}, \quad 1 \leq i \leq M-1,\ 1 \leq j \leq N,$$

et la seconde s'écrit $\partial_x : G^{c1} \to G^{s1}_x$, pour $v \in G^{c1}$:

$$(\partial_x v)_{i,j+1/2} = v_{i,j+1} - v_{i,j}, \quad 1 \leq i \leq M,\ 1 \leq j \leq N-1.$$

L'opérateur de dérivée partielle adjoint pour $\bar{u} = (\bar{u}^1, \bar{u}^2) \in G^{s1}$ est défini par

$$(\partial_t^* \bar{u}^1)_{i,j} = \begin{cases} -\bar{u}^1_{1+1/2,j} & \text{si } i = 1 \\ \bar{u}^1_{i-1/2,j} - \bar{u}^1_{i+1/2,j} & \text{si } 2 \leq i \leq M-1 \\ \bar{u}^1_{M-1/2,j} & \text{si } i = M, \end{cases}$$

et

$$(\partial_x^* \bar{u}^2)_{i,j} = \begin{cases} -\bar{u}^2_{i,1+1/2} & \text{si } j = 1 \\ \bar{u}^2_{i,j-1/2} - \bar{u}^2_{i,j+1/2} & \text{si } 2 \leq j \leq N-1 \\ \bar{u}^2_{i,N-1/2} & \text{si } j = N, \end{cases}$$

Nous montrons dans l'annexe B que les deux opérateurs sont bien adjoints.

En ce qui concerne l'opérateur **rotationnel** et l'opérateur rotationnel adjoint, on considère :

$$\nabla\times = \begin{pmatrix} \partial_x \\ -\partial_t \end{pmatrix} \text{ et } \nabla^*\times = \begin{pmatrix} \partial_x^* \\ -\partial_t^* \end{pmatrix}.$$

L'opérateur discret de dérivée partielle par rapport à la première composante s'écrit $\partial_t : G^{c2} \to G_x^{s1}$, pour $\tilde{v} \in G^{c2}$:

$$(\partial_t \tilde{v})_{i,j-1/2} = \tilde{v}_{i+1/2,j-1/2} - \tilde{v}_{i-1/2,j-1/2}, \quad 1 \leq i \leq M,\ 1 \leq j \leq N+1,$$

et la seconde s'écrit $\partial_x : G^{c2} \to G_t^{s1}$, pour $\tilde{v} \in G^{c2}$:

$$(\partial_x \tilde{v})_{i-1/2,j} = \tilde{v}_{i-1/2,j+1/2} - \tilde{v}_{i-1/2,j-1/2}, \quad 1 \leq i \leq M+1,\ 1 \leq j \leq N.$$

L'opérateur de dérivée partielle adjoint pour $\bar{u} = (\bar{u}^1, \bar{u}^2) \in G^{s1}$ est défini par

$$(\partial_t^* \bar{u}^2)_{i-1/2,j-1/2} = \begin{cases} -\bar{u}^2_{1,j-1/2} & \text{si } i = 1 \\ \bar{u}^2_{i-1,j-1/2} - \bar{u}^2_{i,j-1/2} & \text{si } 2 \leq i \leq M \\ \bar{u}^2_{M,j-1/2} & \text{si } i = M+1. \end{cases}$$

et

$$(\partial_x^* \bar{u}^1)_{i-1/2,j-1/2} = \begin{cases} -\bar{u}^1_{i-1/2,1} & \text{si } j = 1 \\ \bar{u}^1_{i-1/2,j-1} - \bar{u}^1_{i-1/2,j} & \text{si } 2 \leq j \leq N \\ \bar{u}^1_{i-1/2,N} & \text{si } j = N+1. \end{cases}$$

5.1.2 Discrétisation pour des images 2D

Nous décrivons à présent les grilles discrètes utilisées pour l'implémentation des algorithmes dédiés aux images 2D détaillés dans la section 4.2.3.

Grille centrée.

La grille centrée régulière utilisée en 2D est définie par

$$G^c = \{t_i, x_j, y_k\}_{1\leq i\leq M,\ 1\leq j\leq N,\ 1\leq k\leq P},$$

en gardant les mêmes notations que pour la discrétisation 1D. Elle sert à évaluer ρ et m pour calculer la fonctionnelle $\mathcal{J}$ ou pour évaluer la variable duale z lors de l'algorithme

primal-dual.

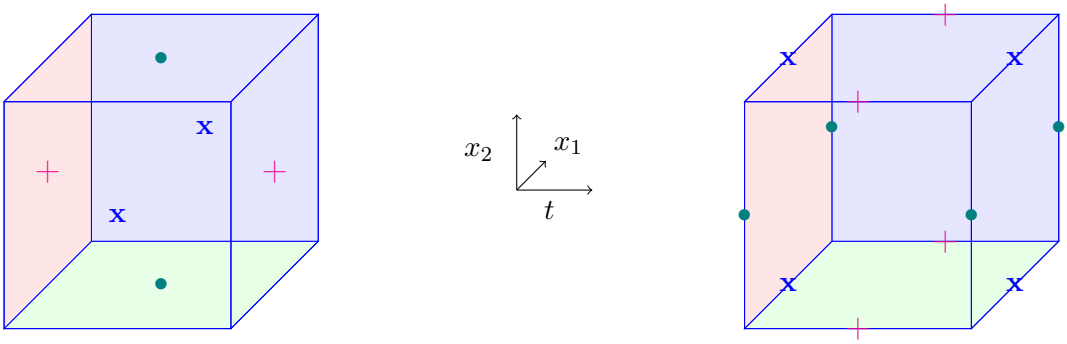
Grilles décalées.
Nous devons ensuite introduire deux grilles décalées pour évaluer les opérateurs divergence et rotationnel, comme illustré sur la figure 5.2.
La première grille G^{s1} nous permet de discrétiser (ρ, m) en accord avec la condition de divergence nulle et est définie par :

$$\begin{aligned} G_t^{s1} &= \{t_{i-1/2},\ x_j,\ y_k\}_{1\leq i\leq M+1,\ 1\leq j\leq N,\ 1\leq k\leq P}, \\ G_x^{s1} &= \{t_i,\ x_{j-1/2},\ y_k\}_{1\leq i\leq M,\ 1\leq j\leq N+1,\ 1\leq k\leq P}, \\ G_y^{s1} &= \{t_i,\ x_j,\ y_{k-1/2}\}_{1\leq i\leq M,\ 1\leq j\leq N,\ 1\leq k\leq P+1}. \end{aligned}$$

La seconde grille G^{s_2} est nécessaire pour définir ϕ afin que son rotationnel, $\nabla \times \phi$, soit positionné sur la grille G^{s_1} :

$$\begin{aligned} G_t^{s2} &= \{t_i,\ x_{j-1/2},\ y_{k-1/2}\}_{1\leq i\leq M,\ 1\leq j\leq N+1,\ 1\leq k\leq P+1}, \\ G_x^{s2} &= \{t_{i-1/2},\ x_j,\ y_{k-1/2}\}_{1\leq i\leq M+1,\ 1\leq j\leq N,\ 1\leq k\leq P+1}, \\ G_y^{s2} &= \{t_{i-1/2},\ x_{j-1/2},\ y_k\}_{1\leq i\leq M+1,\ 1\leq j\leq N+1,\ 1\leq k\leq P}. \end{aligned}$$

En effet, le couple (ρ, m) placé sur la grille G^{s_1} est égal à $\nabla \times \phi + \nabla h$.



Une cellule de la grille G^{s_1} pour l'évaluation de l'équation de continuité pour

$$(\rho, m_1, m_2) = \nabla \times \phi + \nabla h.$$

Une cellule de la grille G^{s_2} pour définir (ϕ_1, ϕ_2, ϕ_3) dont le rotationnel se trouve sur la grille décalée G^{s_1}.

FIGURE 5.2: Grilles décalées pour des images 2D.

Les opérateurs interpolation, gradient et divergence.
Ces opérateurs sont les généralisations en 3D de ceux décrits pour des signaux 1D.

Opérateurs rotationnel.
Pour utiliser l'algorithme primal-dual, il faut définir un opérateur rotationnel discret et

un opérateur rotationnel discret adjoint. Nous définissons l'opérateur rotationnel par la matrice

$$\nabla\times = \begin{pmatrix} 0 & -\partial_y & \partial_x \\ \partial_y & 0 & -\partial_t \\ -\partial_x & \partial_t & 0 \end{pmatrix}$$

et donc l'opérateur adjoint approprié doit être l'opposé du rotationnel déduit des opérateurs de dérivées partielles adjoints :

$$\nabla^*\times = \begin{pmatrix} 0 & \partial_y^* & -\partial_x^* \\ -\partial_y^* & 0 & \partial_t^* \\ \partial_x^* & -\partial_t^* & 0 \end{pmatrix}.$$

5.2 Comparaisons et tests numériques

Dans cette section, nous allons discuter des applications numériques des algorithmes développés dans le chapitre 4 à des signaux 1D et des images 2D.

5.2.1 Algorithmes pour signaux de dimension 1.

Descente de gradient sur coefficients d'ondelettes à divergence nulle

Nous étudions l'algorithme GDDWP (algorithme 2) développé dans la section 4.1.2 dans le cas des ondelettes périodiques à divergence nulle. Cet algorithme utilise une descente de gradient sur les coefficients d'ondelettes, le gradient étant calculé par projection de $\frac{m}{\rho}$ et $\frac{m^2}{\rho^2}$ sur des ondelettes particulières. Par simplicité, on notera dans les figures de cette section les différentes fonctionnelles $\mathcal{J}$ et $\mathcal{J}_w$ par $\mathcal{J}$.

Pour tester et comparer ces différents algorithmes nous allons tout d'abord étudier le cas du déplacement d'une gaussienne vers une autre gaussienne de même variance, ce qui est un exemple classique dont on peut facilement calculer la solution analytique. Nous allons donc choisir pour ρ_0 une gaussienne et pour ρ_1 une translaté de ρ_0 comme le montre la figure 5.3.

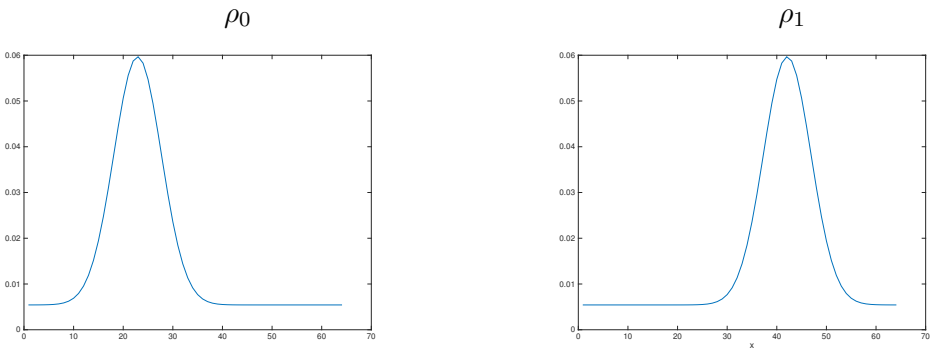


FIGURE 5.3: Représentation des deux densités ρ_0 et ρ_1, deux gaussiennes avec la même variance.

On a choisi

$$\rho_0(x) = exp\left(\frac{-(x-0.35)^2}{2*0.1^2}\right) + 0.1 \text{ et } \rho_1(x) = exp\left(\frac{-(x-0.65)^2}{2*0.1^2}\right) + 0.1 = \rho_0(x-0.3),$$

ce qui donne $T(x) = x + 0.3$ et $T_t(x) = (1-t)x + tT(x) = x + 0.3t$. On obtient alors que

$$\rho_0(x) = \rho^s(t, T_t(x)).\nabla_x(\partial T_t(x)) \tag{5.2}$$
$$\Leftrightarrow \rho_0(x) = \rho^s(t, x + 0.3t) \tag{5.3}$$
$$\Leftrightarrow \rho_0(x - 0.3t) = \rho^s(t, x) \tag{5.4}$$

et $d_2(\rho_0, \rho_1) = 0.3$.

Remarques. Dans la suite les différentes normes considérées sont toutes des normes absolues.

Nous comparons dans cette section notre algorithme à une descente de gradient standard pour la fonctionnelle $\mathcal{J}$ sur ρ et m :

Algorithme 11. *(Descente de gradient (DG))*

Initialisation : $\tau > 0, (\rho^0, m^0)$.

Itérations :

$$(\nabla_\rho J)^i = -\frac{(m^i)^2}{(\rho^i)^2} \text{ et } (\nabla_m J)^i = \frac{m^i}{\rho^i}$$
$$\rho^{i+1} = \rho^i - \tau(\nabla_\rho J)^i \text{ et } m^{i+1} = m^i - \tau(\nabla_m J)^i$$
Projection de (ρ^{i+1}, m^{i+1}) *sur la contrainte de divergence nulle*
Mise a jour des conditions sur le bord.

Nous représentons sur la figure 5.4 la valeur de la fonctionnelle $\mathcal{J}$, pour plusieurs tailles de maillage et l'erreur sur ρ au cours des itérations dans le cas de l'algorithme GDDWP

et dans le cas de l'algorithme DG. Comme on le remarque sur cette figure, l'algorithme GDDWP converge très lentement vers la solution analytique, ce qui est dû à deux facteurs. Tout d'abord, la fonctionnelle est très 'aplatie', et donc la descente de gradient a une vitesse de convergence très lente. Ceci est observé sur la figure 5.4 qui montre l'erreur sur ρ dans le cas de la minimisation de $\mathcal{J}$ n'utilisant que ρ et m (algorithme 11).

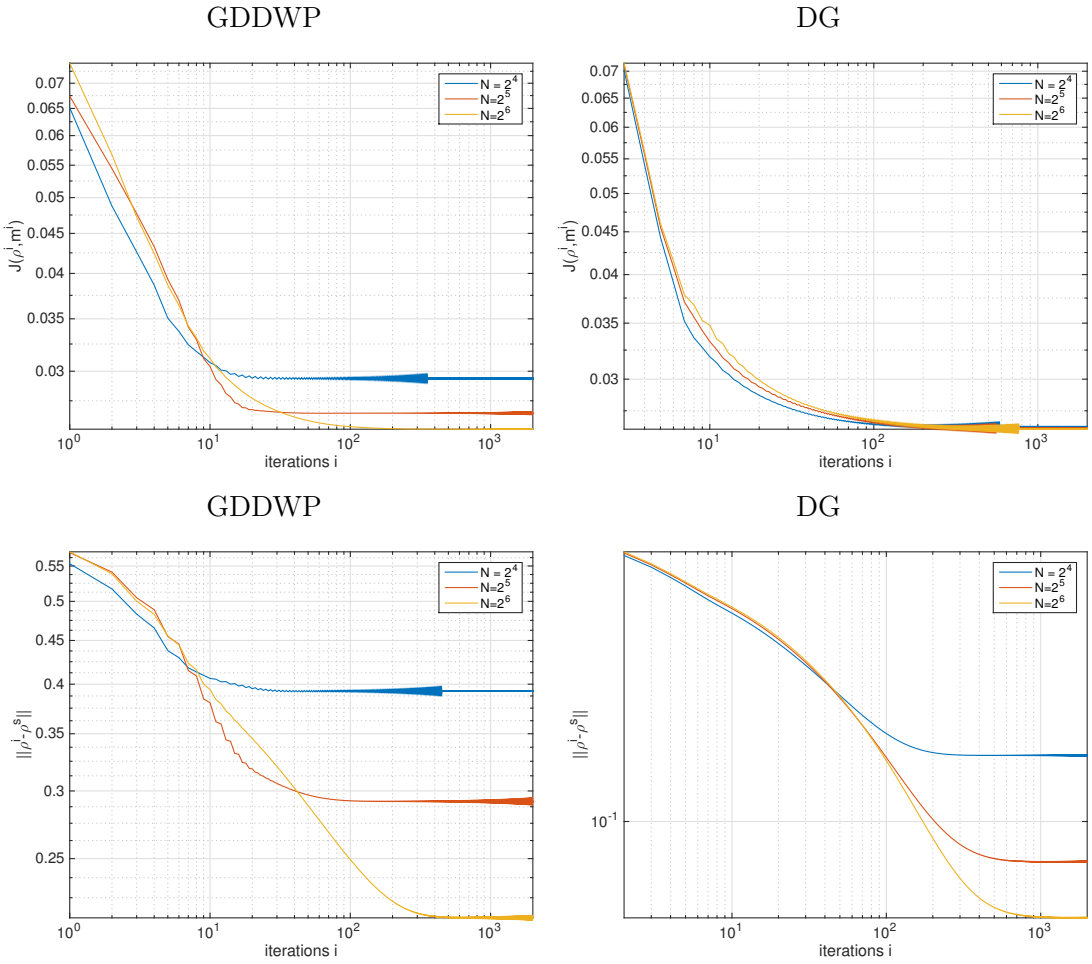


FIGURE 5.4: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s et de la fonctionnelle $\mathcal{J}$ pour différentes tailles de maillage dans le cas de la figure 5.3. Figures de gauche : descente de gradient sur les coefficients d'ondelettes. Figures de droite : descente de gradient sur (ρ, m).

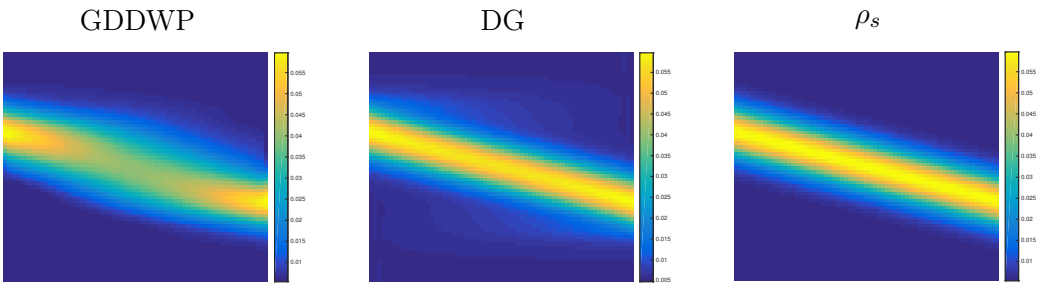


FIGURE 5.5: Représentation de la densité $\rho(t,x)$ obtenue après 2000 itérations dans le cas de la figure 5.3.

La deuxième source de ralentissement de l'algorithme provient du choix des ondelettes. Nous utilisons des ondelettes périodiques, qui interviennent dans deux des étapes de l'algorithme. Dans le calcul du gradient, nous décomposons $\frac{m}{\rho}$ et $\frac{m^2}{\rho^2}$ sur des ondelettes périodiques, puis nous recomposons $\tilde{m}$ et $\tilde{\rho}$ à l'aide de bases d'ondelettes périodiques et donc cela périodise $\tilde{m}$ et $\tilde{\rho}$. L'étape de forçage de la prise en compte des conditions sur le bord va à l'encontre de cette périodicité et donc ralentit la convergence de l'algorithme.

FIGURE 5.6: Représentation de la densité $\rho(t,x)$ obtenue après 1000 itérations dans le cas $\rho_0 = \rho_1$.

La figure 5.6 représente la densité $\rho(t,x)$ où nous avons pris $\rho_0 = \rho_1$ dans le cas de la densité ρ_0 de la figure 5.3.

On observe alors que lorsque ρ est périodique l'algorithme DGDWWP converge bien plus rapidement que l'algorithme DG (figure 5.7), car dans cet exemple la périodicité des ondelettes n'est pas un problème.

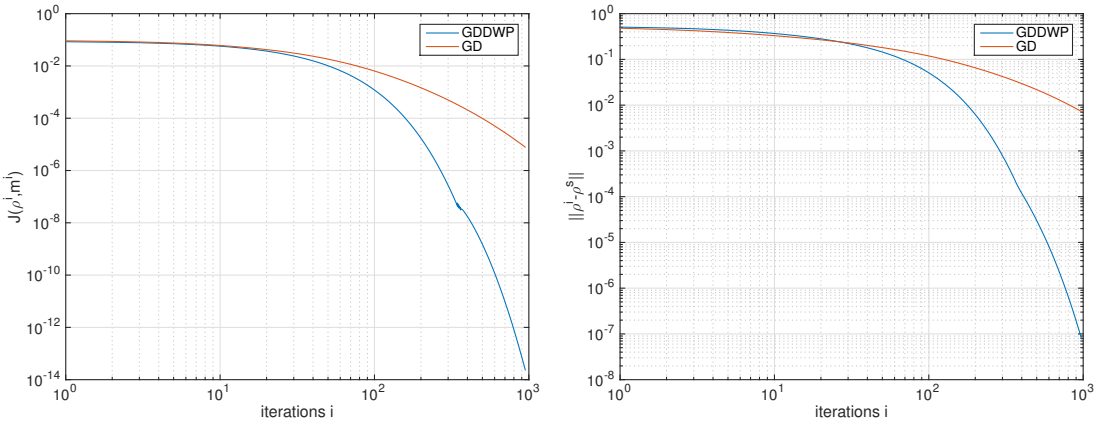


FIGURE 5.7: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s et de la fonctionnelle dans le cas de la figure 5.6.

Nous avons donc développé un premier algorithme permettant de minimiser la fonctionnelle $\mathcal{J}$ à l'aide d'ondelettes à divergence nulle. La vitesse de convergence n'est pas très intéressante avec les ondelettes périodiques mais cela a permis de montrer la validité numérique de l'utilisation des décompositions en ondelettes. Une piste intéressante est l'utilisation les ondelettes à divergence nulle respectant les conditions de Dirichlet sur le bord pour résoudre ce problème [52].

Algorithmes primaux-duaux

Les algorithmes développés dans la section 4.2 que nous souhaitons étudier sont

- PDHH : minimisation de la fonctionnelle $\mathcal{J}_h$ en utilisant $(\rho, m) = \nabla \times \phi + \nabla h$ (algorithme 9)
- PDSF : minimisation de la fonctionnelle $\mathcal{J}$ en utilisant $(\rho, m) = \nabla \times \phi$ (algorithme 7),
- PDMSSF : minimisation de la fonctionnelle $\mathcal{J}_{ms}$ en utilisant $(\rho, m) = \nabla \times \phi$ (algorithme 6),
- PDMSHH : minimisation de la fonctionnelle $\mathcal{J}_{ms,h}$ en utilisant $(\rho, m) = \nabla \times \phi + \nabla h$ (algorithme 8),

ainsi que l'algorithme primal-dual développé dans [69] par Peyré *et al.* et dénoté PD-POP [1], qui consiste en la minimisation de la fonctionnelle $\mathcal{J}$ sur le couple (ρ, m). Par simplicité, on notera dans les figures de cette section les différentes fonctionnelles $\mathcal{J}$, $\mathcal{J}_h$ et $\mathcal{J}_{ms}$ par $\mathcal{J}$.

Pour tester et comparer ces différents algorithmes nous allons tout d'abord étudier le cas du déplacement d'une gaussienne vers une autre gaussienne de même variance, ce qui est un exemple classique dont on peut facilement calculer la solution analytique. Nous allons donc choisir pour ρ_0 une gaussienne isotrope et pour ρ_1 un translaté de ρ_0 comme

1. https://github.com/gpeyre/2013-SIIMS-ot-splitting

le montre la figure 5.8.

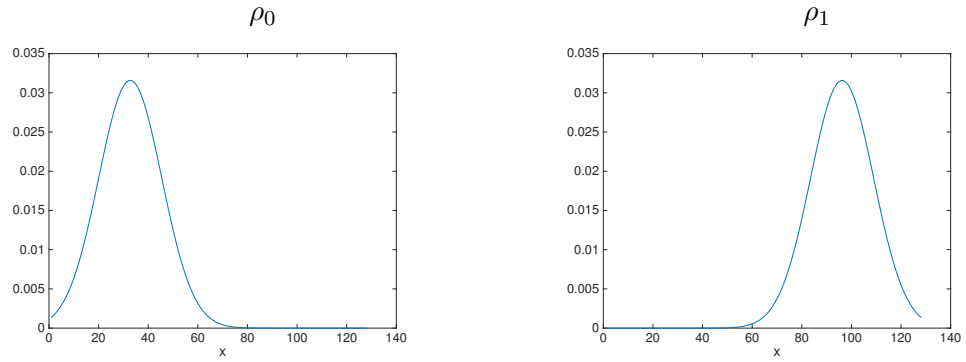


FIGURE 5.8: Représentation des deux densités ρ_0 et ρ_1, deux gaussiennes avec la même variance.

On a choisi

$$\rho_0(x) = exp\left(\frac{-(x-0.25)^2}{2*0.1^2}\right) \text{ et } \rho_1(x) = exp\left(\frac{-(x-0.75)^2}{2*0.1^2}\right) = \rho_0(x-0.5),$$

ce qui donne $T(x) = x + 0.5$ et $T_t(x) = (1-t)x + tT(x) = x + 0.5t$. On obtient alors que

$$\rho_0(x) = \rho^s(t, T_t(x)).\nabla_x(\partial T_t(x)) \tag{5.5}$$
$$\Leftrightarrow \rho_0(x) = \rho^s(t, x + 0.5t) \tag{5.6}$$
$$\Leftrightarrow \rho_0(x - 0.5t) = \rho^s(t, x) \tag{5.7}$$

et $d_2(\rho_0, \rho_1) = 0.5$.
Dans la suite les différentes normes considérées sont toutes des normes absolues. Comme on le remarque sur la figure 5.9, aucun des algorithmes ne converge vers la solution analytique, ce qui est dû à deux facteurs que nous étudierons plus tard. En faisant varier le nombre de points de discrétisation N, comme sur la figure 5.10, on peut mieux observer cette non-convergence des différents algorithmes car l'erreur sature en 10^{-2}.

Remarque. Sans que l'on ait besoin de faire d'opération particulière, la contrainte de positivité de ρ tend à être respectée au cours des itérations comme c'est le cas sur la figure 5.10. Ce sera également le cas pour tous les autres exemples étudiés par la suite.

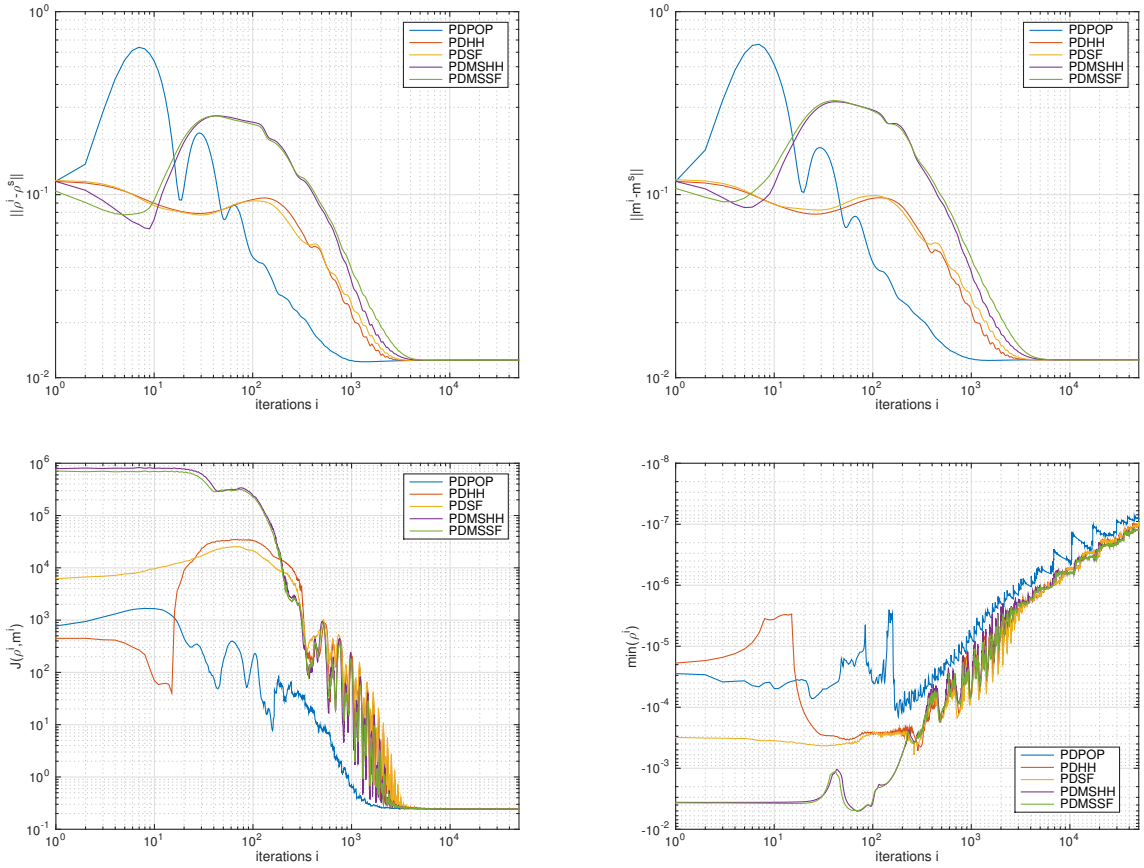


FIGURE 5.9: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s (haut gauche), entre m et m^s (haut droit), fonctionnelle $\mathcal{J}$ (bas gauche) et la valeur minimum de ρ dans le cas de la figure 5.8.

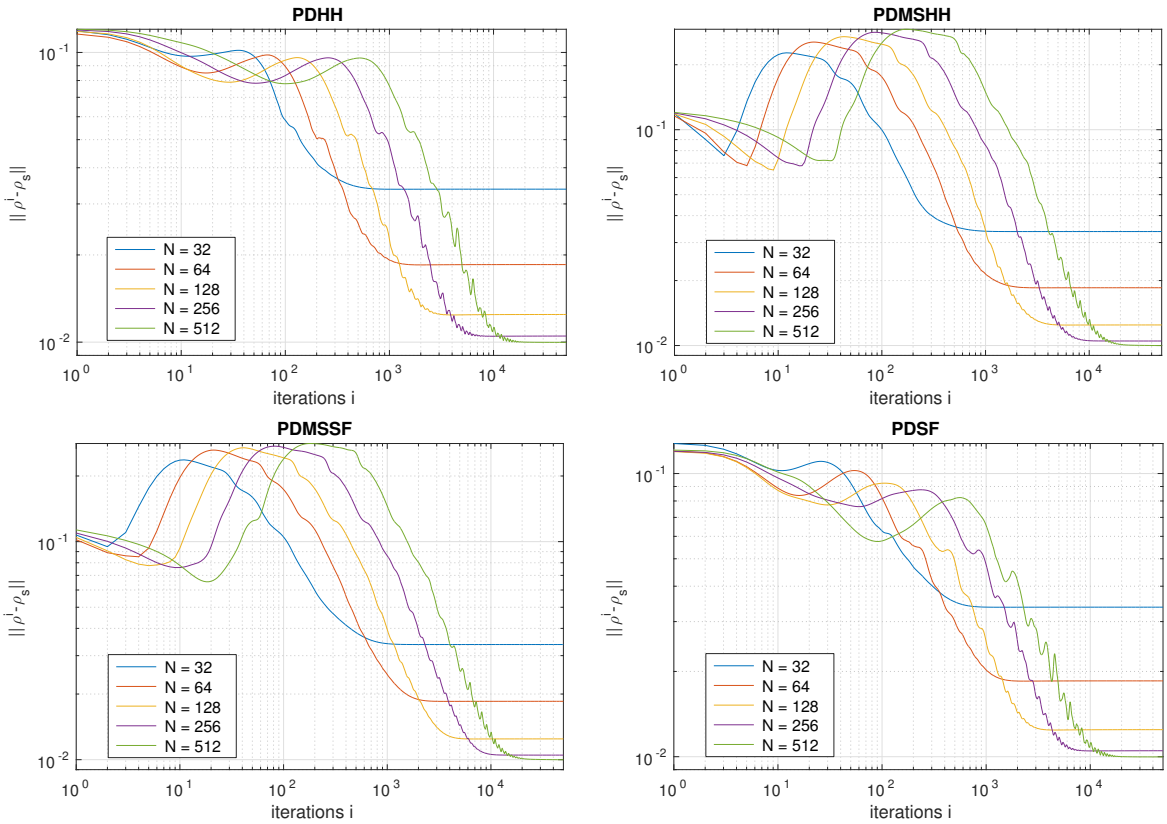


FIGURE 5.10: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s pour différentes tailles de maillage dans le cas de la figure 5.8.

Dans la suite, pour pouvoir comparer un peu mieux les différents algorithmes nous allons construire une solution de référence qui sera utilisée à la place de la solution analytique et qui est représentée en noir et gris sur la figure 5.11, le noir étant 0 et le blanc 1.

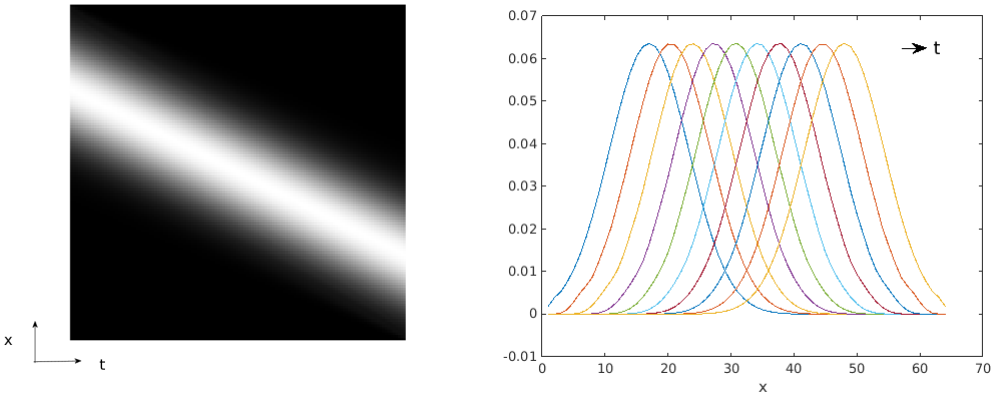


FIGURE 5.11: Représentation de deux vues différentes de la densité $\rho(t,x)$ obtenue après 10^6 itérations, pour ρ_0 et ρ_1 deux gaussiennes avec la même variance.

On commence par calculer une solution de référence (ρ^s, m^s) du problème discrétisé en exécutant pendant 10^6 itérations chaque algorithme que l'on souhaite étudier. On compare ensuite ces solutions avec la solution calculée analytiquement dans le tableau 5.1.

	$\|\|\rho^i - \rho^s\|\|$	$\|\|\rho^i - \rho^s\|\|_\infty$
$PDPOP$	0.0124589943	0.03087598
$PDHH$	0.0124589935	0.03087600
$PDSF$	0.0124589931	0.03087602
$PDMSHH$	**0.01245653840**	**0.0308716683**
$PDMSSF$	0.01245653848	0.0308716687

TABLE 5.1: Comparaison de l'erreur sur ρ après 10^6 itérations de chaque algorithme avec $\sigma = 20$ et $N = M = 128$ dans le cas de la figure 5.8.

On se rend donc compte que les solutions de référence les plus proches de la solution analytique sont celles calculées à l'aide des algorithmes PDMSHH et PDMSSF, c'est-à-dire calculées en minimisant la fonctionnelle $\mathcal{J}_{ms}$ (4.15). Nous utiliserons dans la suite la solution calculée avec PDMSHH pour comparer les différents algorithmes.
Il est possible d'observer les différences entre les solutions de référence calculées par les différents algorithmes sur les images 5.12.

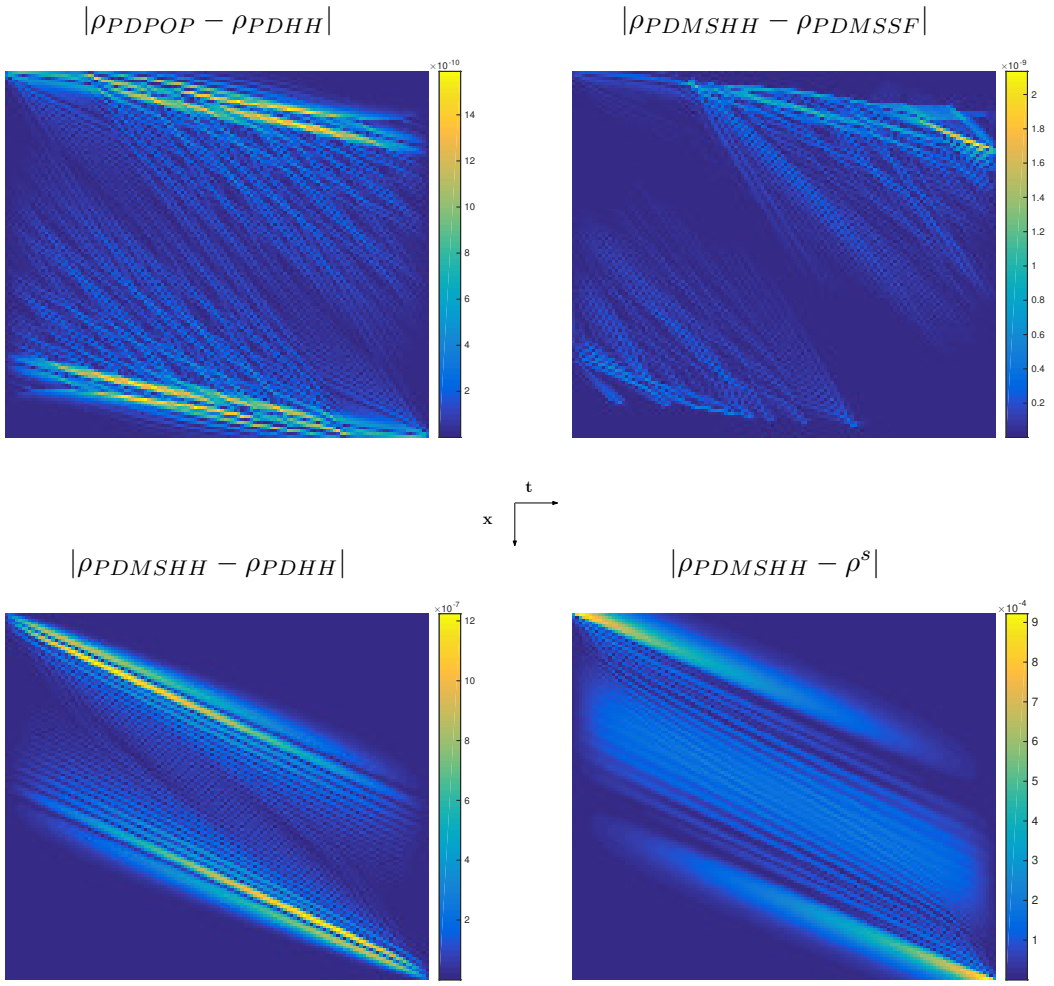


FIGURE 5.12: Comparaison entre les différentes solutions calculées par les différents algorithmes dans le cas de la figure 5.8.

Comme on le remarque sur la figure ci-dessus, l'erreur entre deux solutions calculées en minimisant la même fonctionnelle (soit $\mathcal{J}$, soit $\mathcal{J}_{ms}$) est en 10^{-9}, qu'on utilise pour représenter le couple (ρ, m) la décomposition de Helmholtz-Hodge $(\nabla \times \phi + \nabla h)$, une fonction de courant $(\nabla \times \phi)$ ou aucune décomposition. Alors que l'erreur entre un algorithme minimisant $\mathcal{J}$ ou $\mathcal{J}_h$ et un algorithme minimisant $\mathcal{J}_{ms}$ ou $\mathcal{J}_{ms,h}$ est en 10^{-6}.
L'erreur par rapport à la solution analytique est due à deux phénomènes. Tout d'abord, le premier phénomène est l'apparition d'oscillations le long des lignes de niveaux de ρ. Par ailleurs, on observe une erreur importante en les deux points $((t = 0, x = 0)$ et $(t = 1, x = 1))$ situés sur le bord du domaine où ρ est supérieure à zéro et cette erreur se propage. Cela s'explique par les opérateurs adjoints que l'on doit utiliser dans les différents algorithmes et qui créeent sur le bord en $x = 0$ ou $x = 1$ des discontinuités, plus

ou moins importantes selon la valeur de ρ en ce bord.

Courbes de convergence pour différents σ

Pour comparer les convergences des différents algorithmes et étudier leurs performances, il faut tout d'abord choisir tous les paramètres correctement. Le seul paramètre à choisir sans condition particulière est σ, qui intervient dans la première étape de l'algorithme comme paramètre de descente le long de la fonctionnelle $\mathcal{J}$. Il n'y a pas d'étude théorique indiquant quel σ utiliser dans quelle condition. Nous allons donc comparer les courbes de convergence des différents algorithmes avec le même σ, pour observer son influence sur ces derniers. Les figures 5.13 et 5.14 nous montrent respectivement l'erreur sur ρ et la minimisation de la fonctionnelle $\mathcal{J}$ pour $\sigma = 2, 5, 15$ ou 20.

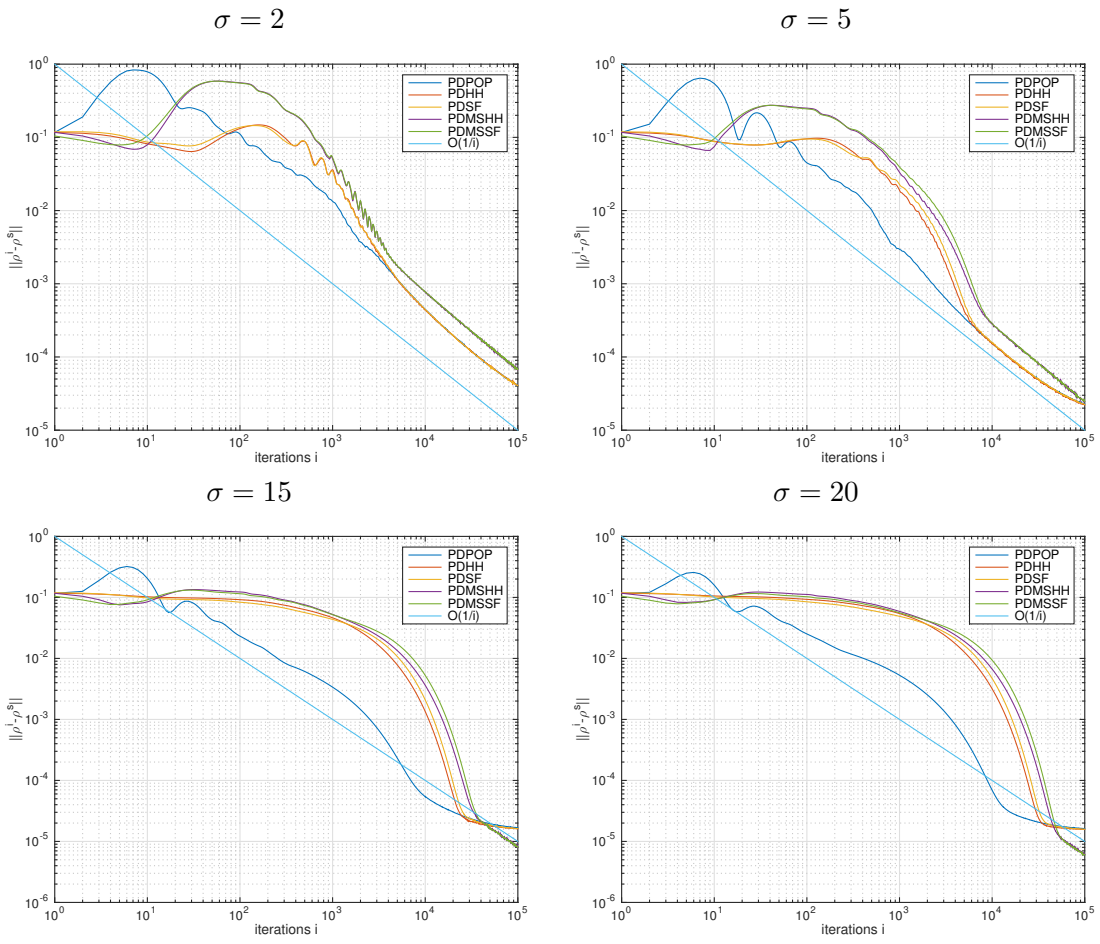


FIGURE 5.13: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s pour différents σ dans le cas de la figure 5.8.

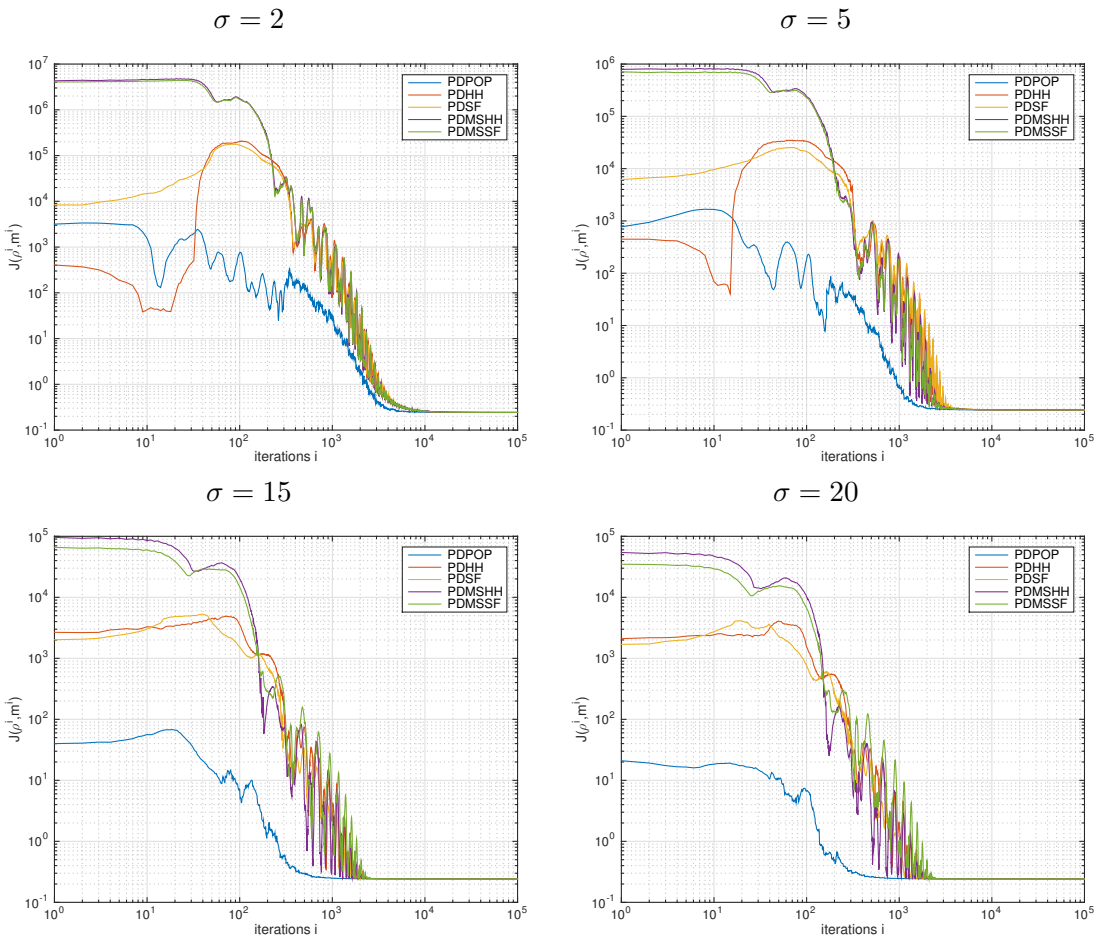


FIGURE 5.14: Comparaison à chaque itération de la fonctionnelle pour différents σ dans le cas de la figure 5.8.

On remarque qu'en augmentant la valeur de σ, pour les algorithmes que l'on a développés, la convergence en norme L^2 de ρ est de plus en plus lente pendant les premières itérations puis de plus en plus rapide dans les itérations suivantes. De ce fait, si l'on souhaite obtenir une erreur grossière, il suffira d'utiliser un σ de petite valeur, alors que si l'on souhaite atteindre une erreur plus petite il faudra choisir un σ de valeur plus grande.

Dans la suite nous utiliserons $\sigma = 5$, car il représente un bon compromis pour l'étude que nous souhaitons faire de nos algorithmes. La convergence n'est pas trop lente pendant les premières itérations mais devient plutôt rapide par la suite.
En ce qui concerne la fonctionnelle $\mathcal{J}$, sa minimisation est plus rapide quand on augmente σ. Cela s'explique par le fait que σ est le paramètre de descente dans la première étape de l'algorithme primal-dual, qui est la projection sur le paraboloide induit par la fonctionnelle $\mathcal{J}$. Or, plus il augmente plus τ, qui est le paramètre de descente de la seconde étape de l'algorithme diminue (car $\sigma\tau||K|| < 1$).

Remarque. Malheureusement, choisir un σ intéressant dans le cas de la gaussienne ne veut pas dire que ce σ sera le plus intéressant pour un autre cas. Le choix de sa valeur dépend de la taille du signal qui nous intéresse, des densités ρ_0 et ρ_1 considérées et de l'initialisation de l'algorithme.

Courbes de convergence pour différentes initialisations

Nous observons à présent l'influence de l'initialisation sur la convergence des différents algorithmes. Pour cela, sur la figure 5.15, nous utilisons comme initialisation :

— départ 0 : nous prenons $\phi = 0$ comme initialisation,
— départ 1 : nous prenons ϕ tel que l'erreur en ρ est de 10^{-1},
— départ 2 : nous prenons ϕ tel que l'erreur en ρ est de 10^{-2}.

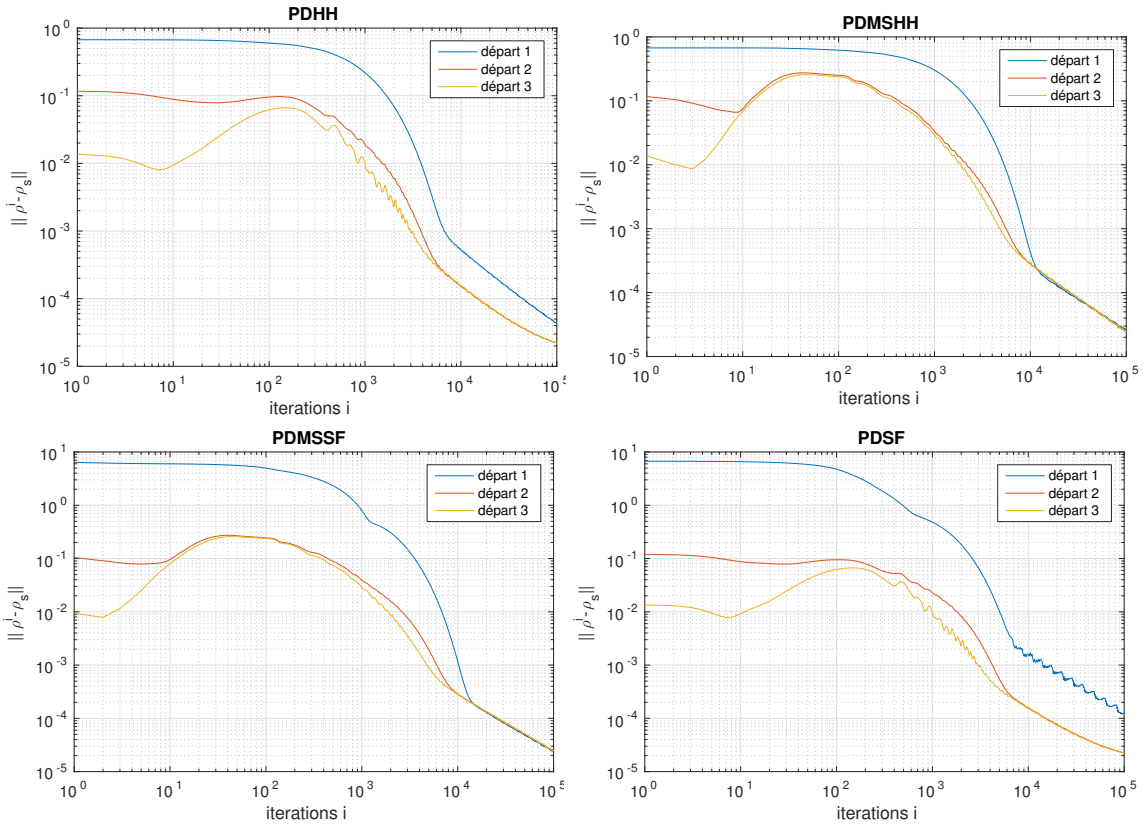


FIGURE 5.15: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s pour différentes initialisations dans le cas de la figure 5.8.

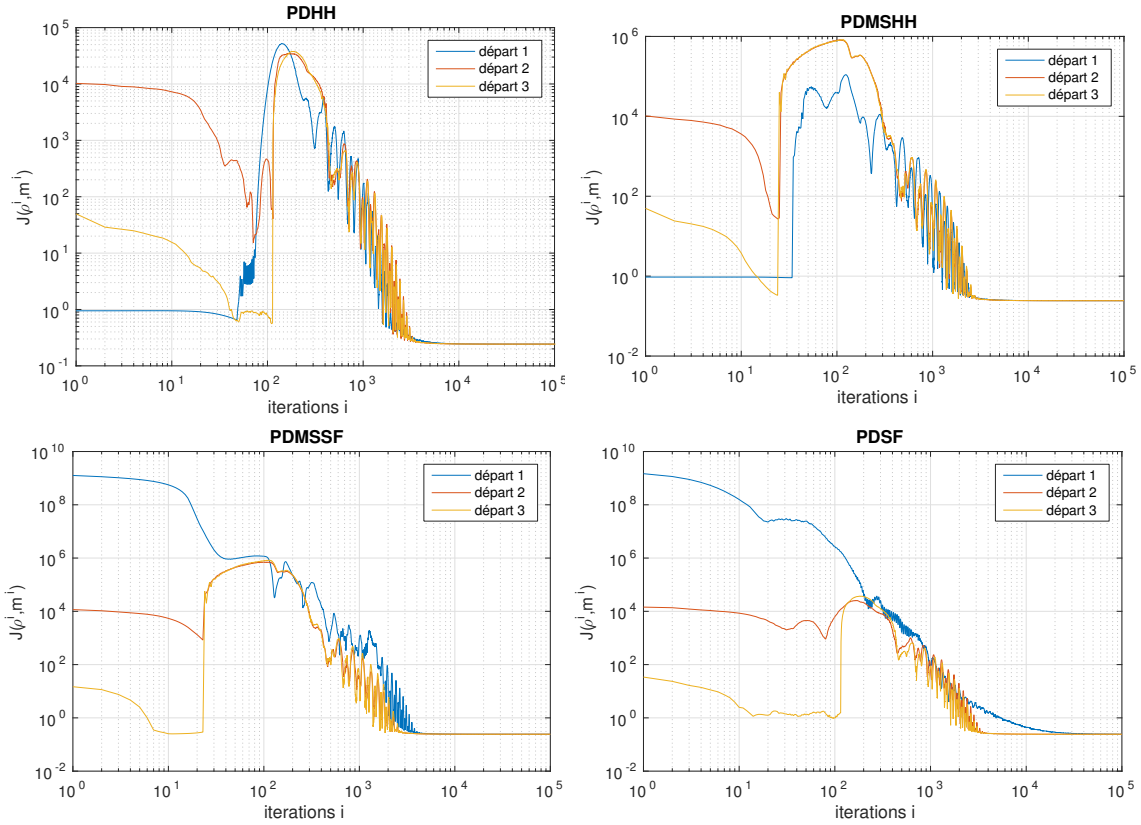


FIGURE 5.16: Comparaison à chaque itération de la fonctionnelle pour différentes initialisations dans le cas de la figure 5.8.

On remarque que tous les algorithmes ont le même taux de convergence après un certain nombre d'itérations et que seulement les algorithmes minimisant la fonctionnelle $\mathcal{J}_h$ (3.27), lorsqu'ils ont comme initialisation $\phi = 0$, ont besoin de plus d'itérations pour atteindre une erreur donnée.

Convergence des variables primale et duale et projection sur le paraboloide

La dernière étude de convergence pour l'exemple des gaussiennes concerne la convergence des variables primale et duale des différents algorithmes, ainsi que l'évolution du nombre de points présents dans le paraboloide dont $\mathcal{J}^*$ est la fonction indicatrice (4.13), avant la projection au cours des itérations, comme représenté sur la figure 5.17.

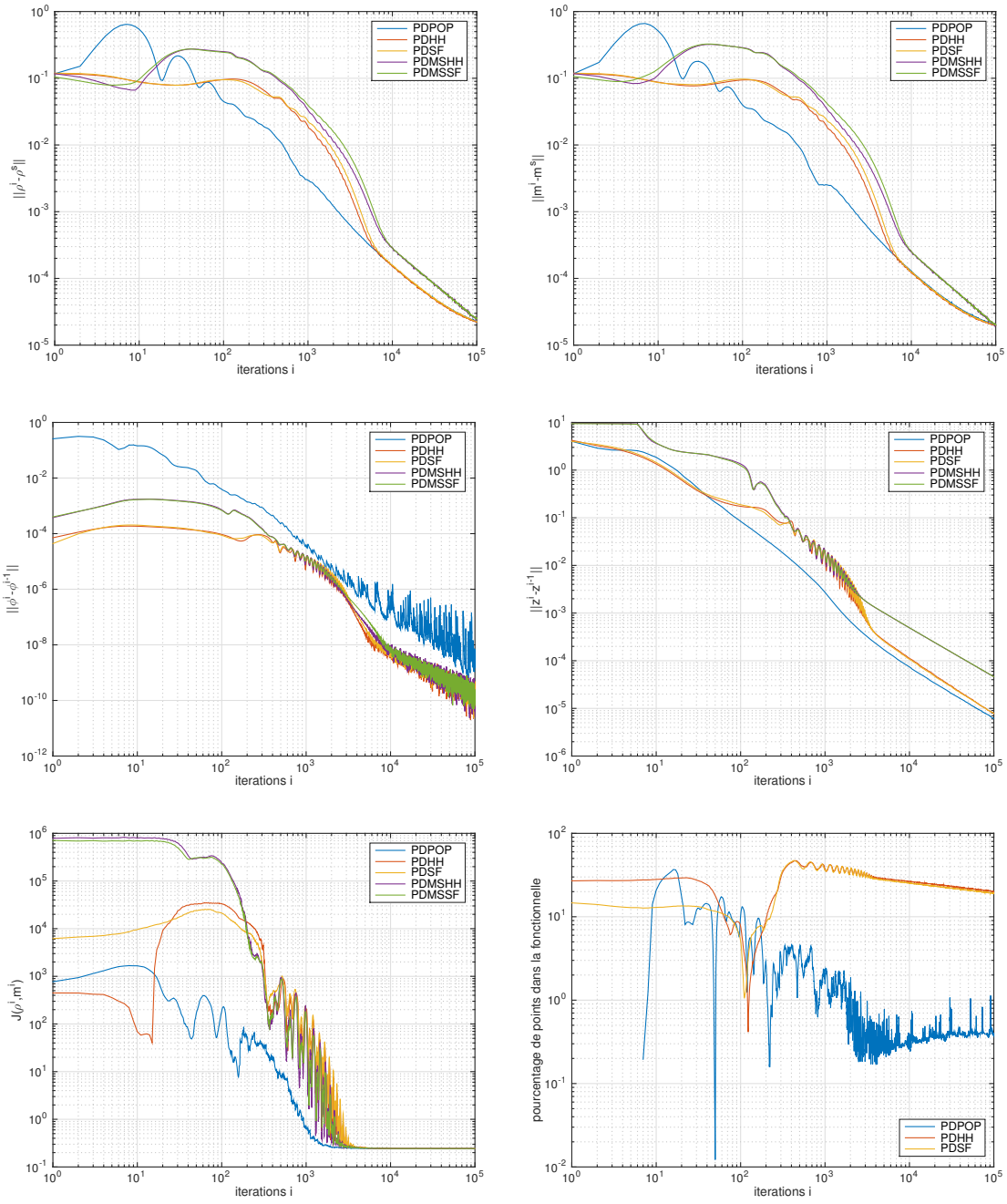


FIGURE 5.17: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s (haut gauche), entre m et m^s (haut droit), de l'erreur entre deux itérations de ϕ (milieu gauche), de z (milieu droit), de la minimisation de la fonctionnelle $\mathcal{J}$ (bas gauche) et du nombre de points restants dans le paraboloide après une itération (bas droit), dans le cas de la figure 5.8.

On se rend compte que lorsque l'erreur sur ϕ commence à osciller, l'erreur sur m et ρ se stabilise. De plus, après 5000 itérations, lorsque la différence entre deux ϕ calculés de façon consécutive par les algorithmes n'est plus que de 10^{-6}, celle entre deux z est autour de 10^{-3}. Cela est dû aux paramètres utilisés. La variable duale z est mise à jour à l'aide de σ qui vaut 5, alors que ϕ est mis à jour avec τ qui dans le cas de l'exemple ci-dessus vaut 3.10^{-6} pour nos algorithmes et 5.10^{-1} pour PDPOP. De plus, il y a une grande différence entre l'initialisation de z et ϕ, particulièrement pour les algorithmes travaillant dans l'espace des contraintes.

Enfin on observe que les algorithmes restant dans la contrainte de divergence nulle (PDHH et PDSF) ont bien plus de points qui restent dans le paraboloide P_J (4.13) à chaque itération mais que cela baisse au cours des itérations. En ce qui concerne l'algorithme PDPOP, bien moins de points restent dans le paraboloide après une itération, mais cela augmente au cours des itérations. Par exemple à la 100 000ième itération, 20% des points (∼3277 points) sont dans le paraboloide P_J dans le cas de l'algorihme PDHH, 19% (∼3113 points) dans le cas de l'algorithme PDSF et seulement 0,4% (∼66 points) pour l'algorithme PDPOP. Cette différence est importante car la projection sur le paraboloide est très coûteuse.

Complexité

Pour étudier la complexité, il faut se rappeler de la forme générale de l'algorithme :

Algorithme primal-dual
— Initialisation : $\tau, \sigma > 0, \theta \in [0,1],\ (\phi^0, z^0 = K\phi^0, \tilde{\phi}^0 = \phi^0)$.
— Itérations :

$$\begin{aligned} z^{i+1} &= \text{prox}_{\mathcal{J}^*}(z^i + \sigma(K\tilde{\phi}^i)) \\ \phi^{i+1} &= \text{prox}_{\iota_{C_0}}(\phi^i - \tau K^* z^{i+1}) \\ \tilde{\phi}^{i+1} &= \phi^{i+1} + \theta(\phi^{i+1} - \phi^i). \end{aligned}$$

Dans toute la suite, N représente le nombre de points de discrétisation en temps, qui est le même en espace et ainsi N^2 est le nombre de points de la grille centrée G^{c1} (5.1) :
— la première étape est une projection point par point pour tous les algorithmes utilisés et est en $O(N^2)$,
— la deuxième étape est une projection sur la contrainte de divergence nulle pour l'algorithme PDPOP et donc la résolution d'une équation de Poisson en 2 dimensions en utilisant une transformée en cosinus rapide de complexité $O(N^2 log(N))$.
— la deuxième étape pour les autres algorithmes consiste à mettre les bonnes valeurs sur le bord car nous sommes déjà à divergence nulle et la complexité est en $O(N)$.
Les nouveaux algorithmes ont ainsi une complexité numérique linéaire en le nombre de points total de discrétisation, N^2.

La figure 5.18 représente la complexité numérique des différents algorithmes étudiés dans cette thèse ainsi que de certaines des fonctions les composant. Le temps CPU nécessaire pour une itération des différents algorithmes ou pour les fonctions les constituant, est représenté en fonction du nombre de points de discrétisation dans une direction, N. Les calculs ont été effectués sur un bi-processeur Intel Core i7 avec 2.8GHz.
Nous dénoterons dans la figure 5.18 par K_{PDPOP} l'opérateur K utilisé dans l'algorithme $PDPOP$ et par K_{PDHH} celui utilisé dans l'algorithme PDHH, qui a la même complexité que ceux utilisés pour les algorithmes $PDSF$, $PDMSSF$ et $PDMSHH$. Ce que nous dénotons $proj_{div}$ est la projection sur la contrainte de divergence nulle qui correspond à la seconde étape de l'algorithme PDPOP, la résolution d'une équation de Poisson en deux dimensions.

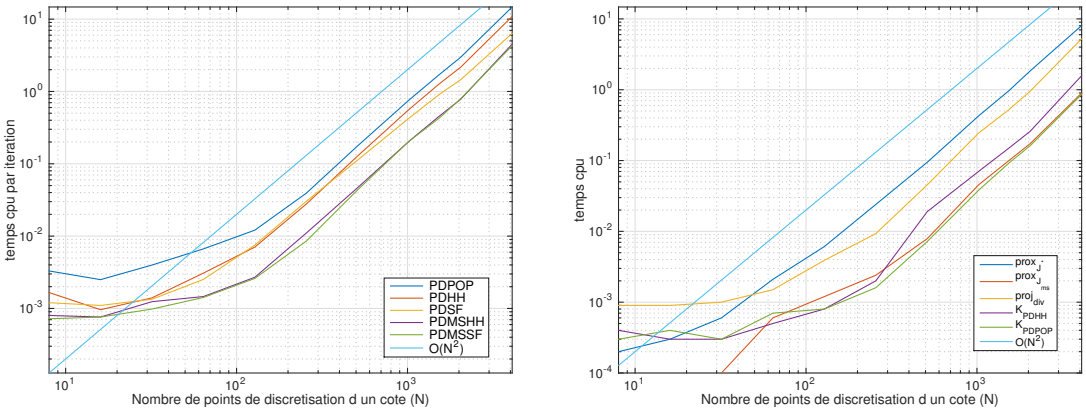


FIGURE 5.18: Comparaison des complexités numériques des différents algorithmes (gauche) et des fonctions les composant (droite) dans le cas de la figure 5.8.

En moyenne, on observe une accélération par rapport à l'algorithme PDPOP de
— 39% pour l'algorithme PDHH,
— 49% pour l'algorithme PDSF,
— 73% pour l'algorithme PDMSHH,
— 75% pour l'algorithme PDMSSF.

Les performances de l'algorithme PDHH s'expliquent par l'absence de projection sur la contrainte de divergence nulle. De plus, comme on l'a vu dans la figure 5.17, cet algorithme nécessite la projection de moins de points que l'algorithme PDPOP ce qui est coûteux. On perd en revanche un peu de temps à cause de notre opérateur K_{PDHH} qui comprend un rotationnel et une interpolation alors que l'opérateur K_{PDPOP} utilisé dans l'algorithme PDPOP est une interpolation.
Les performances des algorithmes PDMSSF et PDMSHH sont dues à l'utilisation de la projection sur la boule unité au lieu de $prox_{J^*}$ qui est une projection sur un paraboloide nécessitant la résolution d'une équation de degré trois en chaque point de la grille. On

remarque les différences de performance sur la figure 5.18 (gauche).
La différence entre les performances des algorithmes PDSF et PDHH est due au nombre de points à projeter sur le paraboloide. En effet, lorsqu'on augmente le nombre de points de discrétisation en gardant les mêmes paramètres, le nombre de points restant dans le paraboloide entre chaque itération varie de façon différente selon les algorithmes.

Conclusions pour l'exemple des gaussiennes.

Nous terminons ce paragraphe avec deux tableaux. Le premier montre le nombre d'itérations et le temps CPU nécessaires pour atteindre une erreur donnée en ρ, pour les différents algorithmes. Le second montre l'accélération des algorithmes que nous avons développés pour cet exemple en comparaison avec l'algorithme PDPOP proposé par [69].

$\|\|\rho_i - \rho_s\|\|$	$PDPOP$	$PDHH$	$PDSF$	$PDHHMS$	$PDMSSF$
10^{-2}	505 (7")	1507 (10")	1820 (12")	2169 (6")	2633 (8")
10^{-3}	2277 (30")	3828 (25")	4221 (29")	5575 (15")	6051 (15")
10^{-4}	15097 (3'02")	14927 (2'00")	15225 (1'50")	25750 (1'10")	25994 (1'05")

Nombre d'itérations et temps CPU pour une erreur donnée

$\|\|\rho_i - \rho_s\|\|$	$PDHH$	$PDSF$	$PDMSHH$	$PDMSSF$
10^{-2}	x	x	14%	x
10^{-3}	17%	3%	50%	50%
10^{-4}	34%	40%	62%	64%

Accélération par rapport à l'algorithme PDPOP pour une erreur donnée

TABLE 5.2: Évaluation des performances pour une erreur donnée dans le cas de la figure 5.8. Dans le second tableau, les 'x' représentent une absence d'accélération.

Nous remarquons que les algorithmes que nous avons développés nécessitent tous un plus grand nombre d'itérations pour converger que l'algorithme PDPOP, sauf PDHH quand on souhaite obtenir une erreur de 10^{-4}, comme on peut le voir sur la figure 5.17. Cependant, la rapidité d'exécution de ces algorithmes, comme mis en avant sur la figure 5.18 compense ce problème pour tous les algorithmes pour les erreurs 10^{-3} et 10^{-4} et même pour l'erreur 10^{-2} dans le cas de l'algorithme PDMSHH.

Tests avec d'autres densités ρ_0 et ρ_1

Densités irrégulières Nous allons maintenant étudier le transport entre deux densités non régulières comme le montre la figure 5.19, pour lesquelles, comme dans le cas des gaussiennes, nous pouvons calculer la solution analytique. On a choisi $\rho_0(x) = \rho_1(x-0.5)$

ce qui donne $T(x) = x + 0.5$ et $T_t(x) = (1-t)x + tT(x) = x + 0.5t$ comme précédemment dans le cas des gaussiennes. On obtient alors à nouveau

$$\rho_s(t,x) = \rho_0(x - 0.5t) \text{ et } d_2(\rho_0, \rho_1) = 0.5.$$

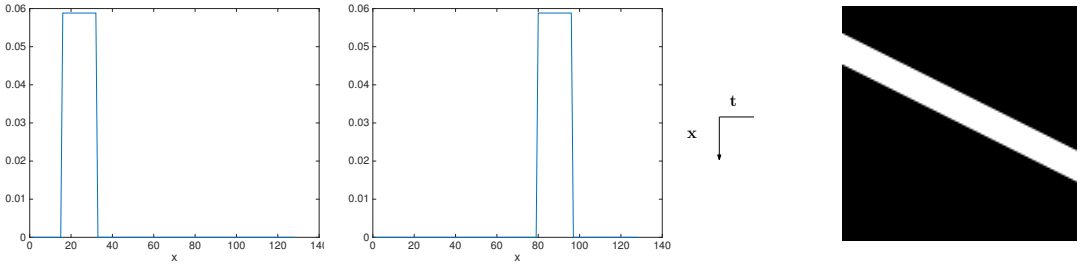


FIGURE 5.19: Représentation des deux densités ρ_0 et ρ_1 et de la densité $\rho(t,x)$ obtenue après 10^6 itérations.

On observe dans le tableau suivant qu'aucun des algorithmes ne converge vers la solution analytique, ce qui est dû à l'apparition d'oscillations, importantes au niveau des irrégularités, comme on le remarque sur les figures 5.20. Dans la suite, pour comparer un peu mieux les différents algorithmes, nous allons construire une solution de référence qui sera utilisée à la place de la solution analytique. On commence donc par calculer une solution de référence (ρ^s, m^s) du problème discrétisé en faisant tourner pendant 10^6 itérations chaque algorithme à étudier. On compare ensuite ces solutions avec la solution calculée analytiquement (ρ^i, m^i) dans le tableau 5.3.

	$\|\|\rho^i - \rho^s\|\|$	$\|\|\rho^i - \rho^s\|\|_\infty$
$PDPOP$	0.21346160	0.66242041
$PDHH$	0.21346136	0.66242156
$PDSF$	0.21345906	0.66242057
$PDMSHH$	**0.21312624**	0.662164764
$PDMSSF$	0.21312634	**0.66216392**

TABLE 5.3: Comparaison de l'erreur par rapport à la solution analytique après 10^6 itérations de chaque algorithme avec $\sigma = 20$ et $N = M = 128$ dans le cas de la figure 5.19.

On se rend compte que les solutions de référence les plus proches de la solution analytique sont celles calculées à l'aide des algorithmes PDMSHH et PDMSSF.
On observe les différences entre les solutions de référence calculées par les différents algorithmes sur les images 5.20.

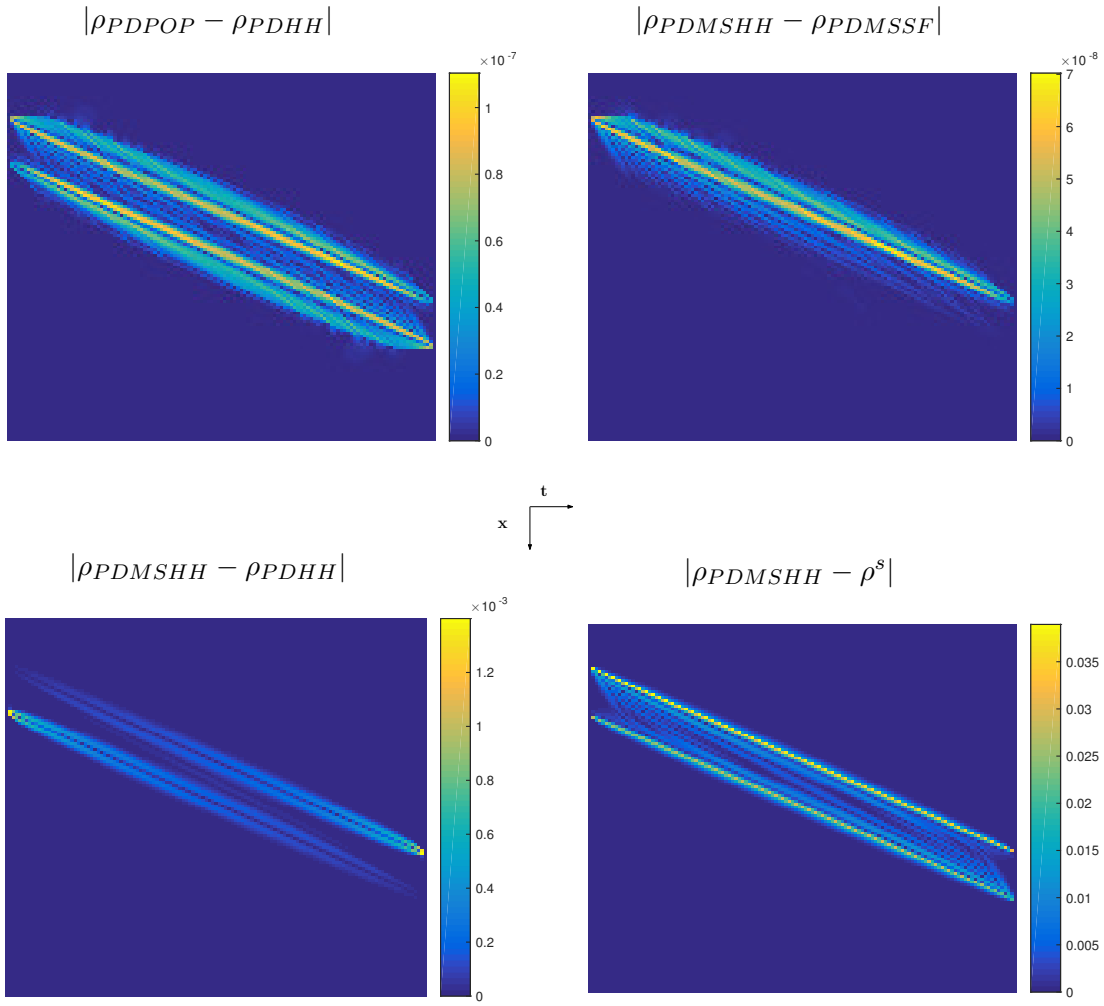


FIGURE 5.20: Comparaison entre les différentes solutions calculées par les différents algorithmes.

Comme précédemment, on remarque sur la figure ci-dessus que l'erreur entre deux solutions calculées en minimisant la même fonctionnelle (soit $\mathcal{J}$ soit $\mathcal{J}_{ms}$) est en 10^{-7}, que l'on utilise pour représenter le couple (ρ, m) la décomposition de Helmholtz-Hodge $(\nabla \times \phi + \nabla h)$, une fonction de courant $(\nabla \times \phi)$ ou aucune décomposition. En revanche, l'erreur entre un algorithme minimisant $\mathcal{J}$ ou $\mathcal{J}_h$ et un algorithme minimisant $\mathcal{J}_{ms}$ ou $\mathcal{J}_{ms,h}$ est en 10^{-3}.
L'erreur par rapport à la solution analytique est due à l'apparition d'oscillations le long des lignes de niveaux de ρ. Par ailleurs, les oscillations deviennent plus importantes aux points d'irrégularité de la solution.

Nous avons gardé les mêmes paramètres dans cet exemple que dans celui des gaus-

siennes et, par conséquent, nos remarques sur les convergences de ϕ et z sont toujours valables ici. Cependant, on remarque que le comportement des courbes est légèrement différent. Ainsi, bien que les erreurs soient les mêmes pour les itérations 1 et 10^5, la convergence est plus lente pendant les premières itérations puis plus rapide pendant les suivantes, ce qui ressemble au comportement que l'on obtient quand on augmente σ.

Enfin, on observe que les algorithmes restant dans la contrainte de divergence nulle (PDHH et PDSF) ont un peu plus de points qui restent dans le paraboloide P_J (4.13) à chaque itération mais que la différence est bien moins importante que dans l'exemple des gaussiennes (figure 5.8). Comme précédemment, ce nombre baisse au cours des itérations. Le comportement des algorithmes restant dans la contrainte de divergence nulle est donc très similaire dans cet exemple et dans celui des gaussiennes alors que celui de l'algorithme PDPOP ne l'est pas. Par exemple à la 10^5 ième itération, 77% des points (∼12605 points) sont dans le paraboloide P_J dans le cas de l'algorithme PDHH, 77% (∼12601 points) dans le cas de l'algorithme PDSF et seulement 71% (∼11692 points) pour l'algorithme PDPOP.

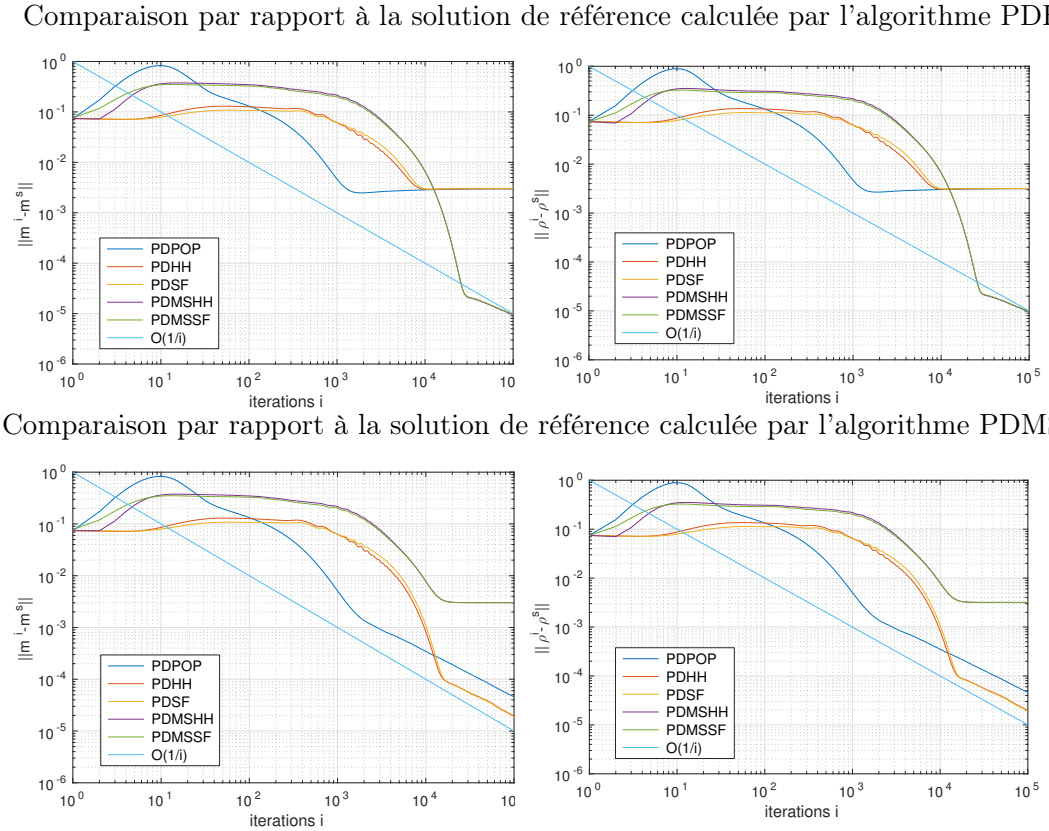


FIGURE 5.21: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s (haut gauche), entre m et m^s (bas droit), dans le cas de la figure 5.19.

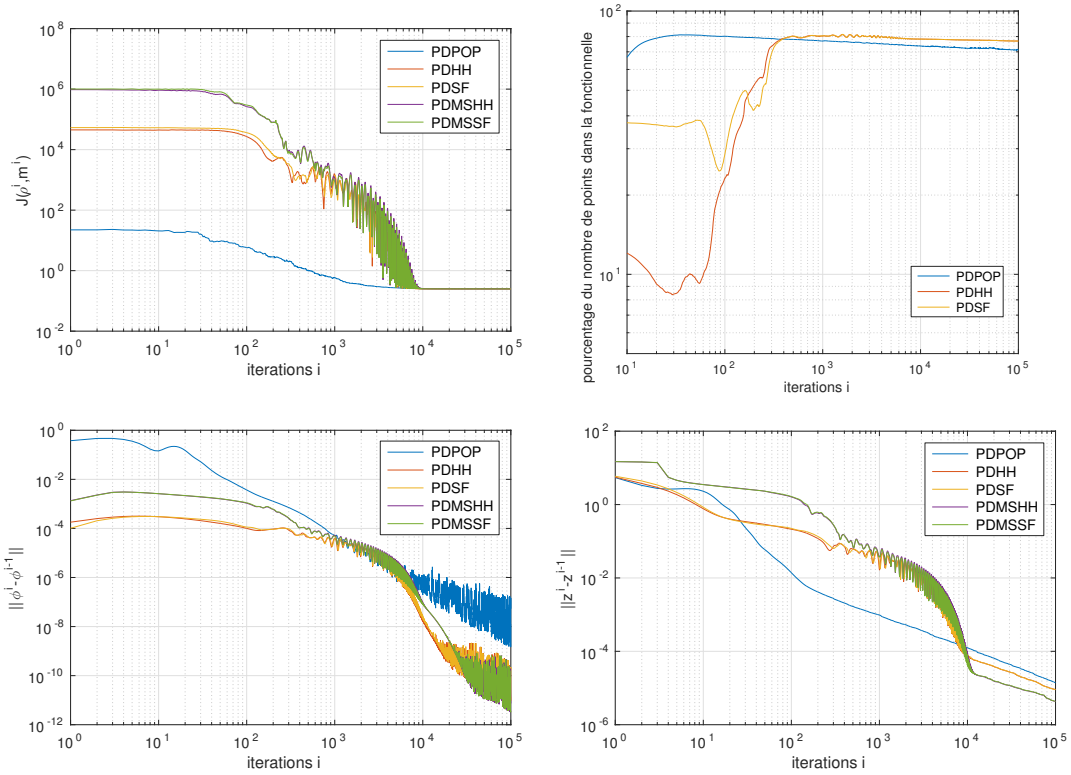


FIGURE 5.22: Comparaison à chaque itération de l'erreur entre deux itérations de ϕ (haut gauche), de z (haut droit), de la minimisation de la fonctionnelle $\mathcal{J}$ (bas gauche) et du nombre de points restant dans le paraboloide après une itération (bas droit), dans le cas de la figure 5.19.

Test pour deux densités lisses et nulles sur le bord. Dans le premier exemple, si l'on prolonge ρ_0 et ρ_1 sur $\mathbb{R}$ par zéro on se rend compte d'une petite irrégularité en $t = 0$ pour ρ_0 et en $t = 1$ pour ρ_1. Nous reprenons ici ce premier exemple mais nous "éloignons" ces gaussiennes du bord du domaine, pour obtenir des densités ρ_0 et ρ_1 lisses. On représente les courbes de convergence de ρ par rapport à la solution analytique, pour $\sigma = 5$, pour les différents algorithmes sur la figure 5.23.

Comme on peut le remarquer les algorithmes convergent de façon satisfaisante vers la solution analytique. De plus, contrairenent aux exemples précédents, ce sont les algorithmes PDPOP, PDHH et PDSF qui convergent le mieux après 10^5 itérations.

Nous observons à présent sur la figure 5.24 les différences entre les solutions de référence calculées après 10^5 itérations et la solution analytique. On remarque comme précédemment que les erreurs sont des oscillations le long des lignes de niveaux de ρ ; cependant, l'erreur qui provenait du bord en x à $t = 0$ et $t = 1$ pour le premier exemple avec des gaussiennes est bien moins importante car la valeur de ρ_0 et ρ_1 au bord est plus faible.

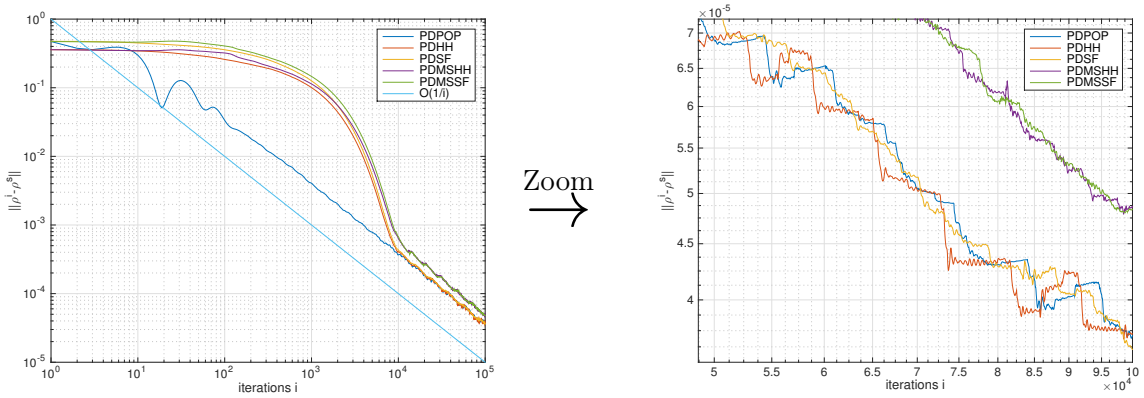


FIGURE 5.23: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s.

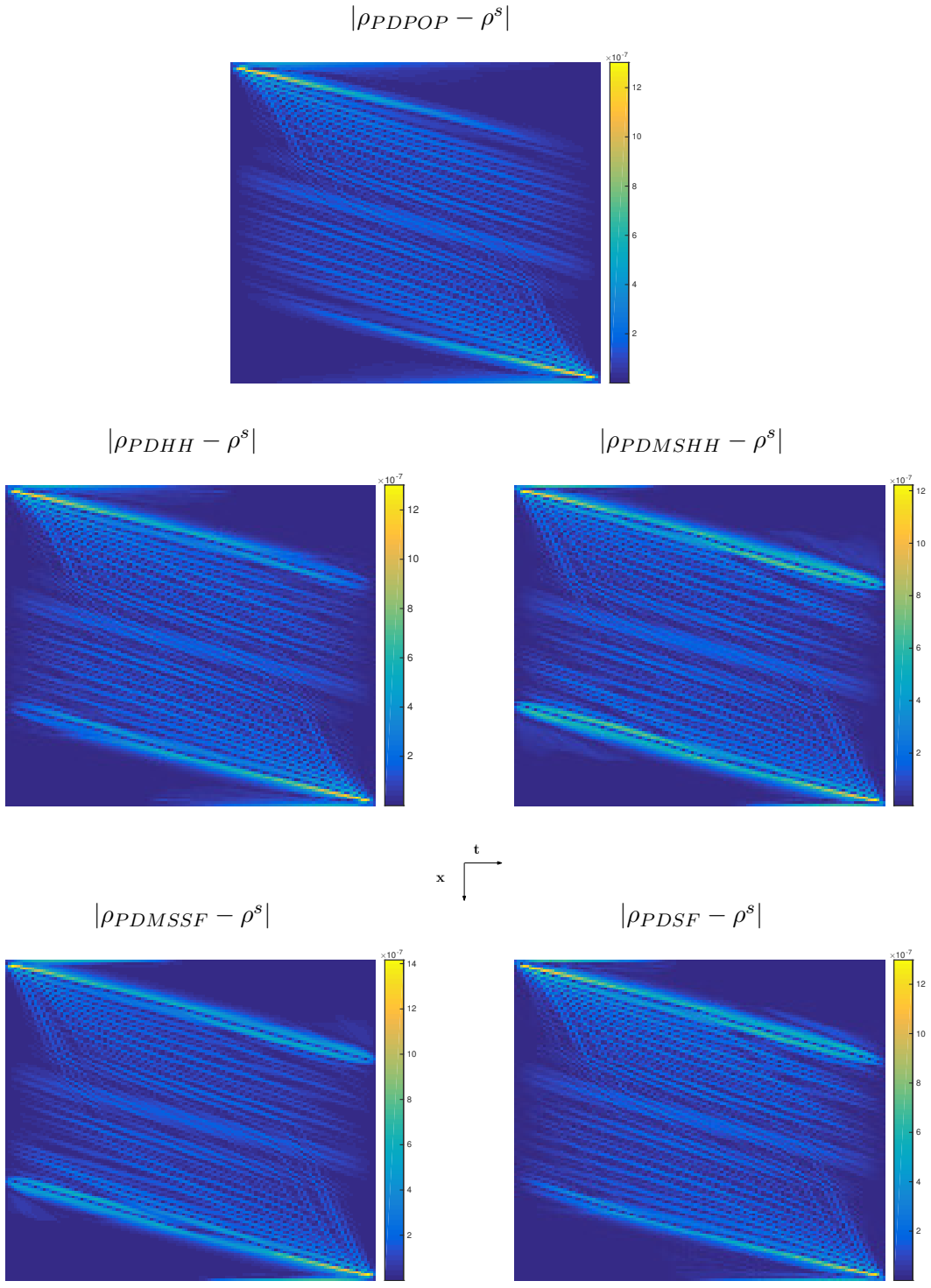


FIGURE 5.24: Comparaison entre les différentes solutions de référence calculées par les différents algorithmes et la solution analytique.

Test pour un signal oscillant et un signal à support compact.

L'étude de signaux 1D inclut, entre autres, l'analyse de signaux sonores. Ceci peut être fait par exemple grâce à l'analyse de Fourier. Pour vérifier que nos algorithmes donnent des résultats cohérents pour ce type de signaux, nous transportons un signal triangulaire à support compact sur un signal oscillant (figure 5.26). Les figures montrent les résultats obtenus sur une grille de 256^2 points après 10^6 itérations pour ρ_0 un signal triangulaire et ρ_1 la valeur absolue d'une fonction sinus. On remarque qu'il y a bien un déplacement en ligne droite (bien observable sur la figure 5.25), comme attendu (voir section 2.4.1).

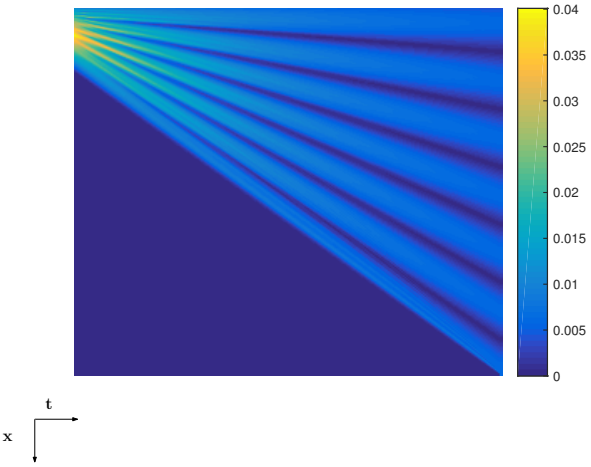
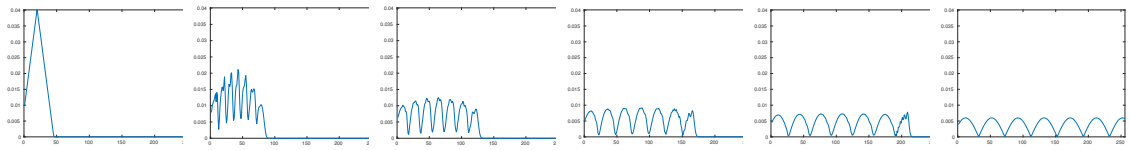


FIGURE 5.25: Représentation de la densité $\rho(t)$ obtenue après 10^6 itérations dans le cas de la figure 5.26.



t = 0 t = 0.2 t = 0.4 t = 0.6 t = 0.8 t = 1

FIGURE 5.26: Représentation de la densité $\rho(t)$ obtenue après 10^6 itérations d'une densité à support compact sur une densité oscillante.

On peut également observer sur la figure 5.26 que c'est à $t = 0$ que l'algorithme "découpe" les 7 oscillations de ρ_1 et que par la suite elles ne vont que se déplacer et se lisser. Si on échange les rôles de ρ_0 et ρ_1, les oscillations resteront présentes jusqu'à $t = 1$.

Test pour deux densités de masses différentes.

Pour vérifier que nos algorithmes donnent des résultats cohérents pour des signaux de masses différentes, nous transportons, comme dernière application, un signal de masse un sur un signal de masse deux (figure 5.28).

Les figures 5.27 et 5.28 montrent les résultats obtenus sur une grille de 128^2 points après 10^6 itérations pour ρ_0 une gaussienne de masse un et ρ_1 une gaussienne de masse deux.

Nous obtenons les résultats présentés à l'aide des algorithmes PDHH et PDMSHH. En effet, dans nos formulations utilisant la fonction de courant du couple (ρ, m), la fonction de courant ϕ doit être constante sur le bord $x = 1$ et égale à la masse de ρ_0 et ρ_1. On ne peut donc pas considérer deux densités de masses différentes.

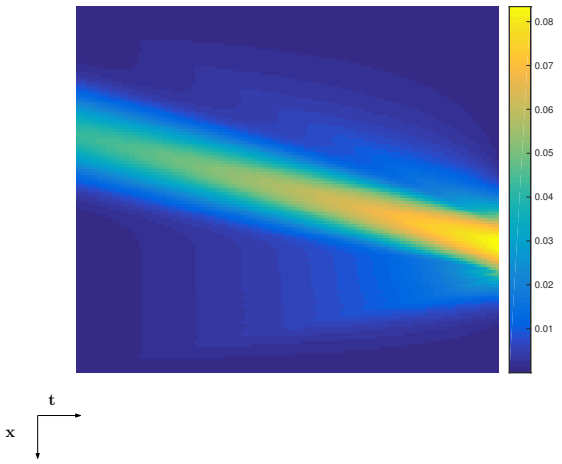
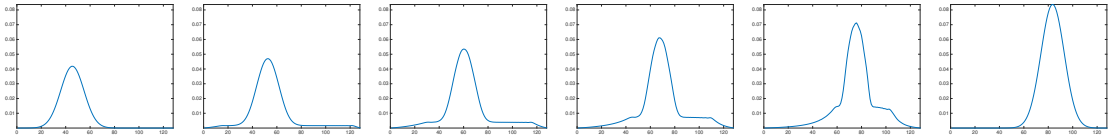


FIGURE 5.27: Représentation de la densité $\rho(t)$ obtenue après 10^6 itérations dans le cas de la figure 5.28.



t = 0 t = 0.2 t = 0.4 t = 0.6 t = 0.8 t = 1

FIGURE 5.28: Représentation de la densité $\rho(t)$ obtenue après 10^6 itérations pour deux densités de masses différentes.

5.2.2 Applications aux images de dimension 2.

Nous souhaitons à présent étudier l'algorithme PDHH (algorithme 9) développé pour des images 2D, qui consiste en la minimisation de la fonctionnelle $\mathcal{J}_h$ en utilisant la décomposition de Helmholtz-Hodge $(\rho, m) = \nabla \times \phi + \nabla h$. Nous comparons cet algorithme à un algorithme primal-dual développé dans [69] par Peyre *et al.* et dénoté PDPOP [2], qui consiste en la minimisation de la fonctionnelle $\mathcal{J}$ sur le couple (ρ, m).

Comme dans la section précédente, nous commençons notre analyse par le déplacement d'une gaussienne vers une autre gaussienne de même variance, comme le montre la figure 5.29.

2. https://github.com/gpeyre/2013-SIIMS-ot-splitting

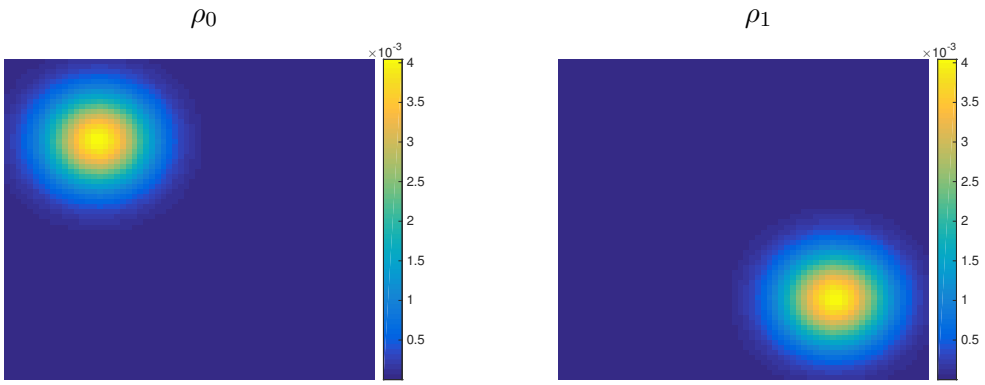


FIGURE 5.29: Représentation des deux densités ρ_0 et ρ_1, deux gaussiennes isotropes avec la même variance.

On a choisi

$$\rho_0(x,y) = exp\left(\frac{-\left((x-0.25)^2+(y-0.25)^2\right)}{2*0.1^2}\right)$$
$$\rho_1(x,y) = exp\left(\frac{-\left((x-0.75)^2+(y-0.75)^2\right)}{2*0.1^2}\right) = \rho_0(x-0.5, y-0.5),$$

ce qui donne, comme dans le cas 1D

$$\rho^s(t,x,y) = \rho_0(x-0.5t, y-0.5t) \text{ et } d_2(\rho_0,\rho_1) = \sqrt{|0.5|^2+|0.5|^2} = 1/\sqrt{2}. \tag{5.8}$$

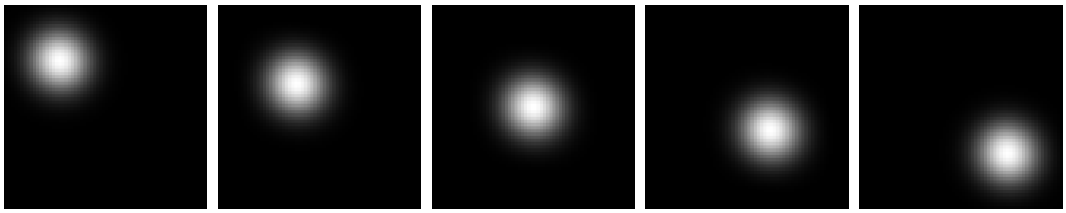


FIGURE 5.30: Représentation de la densité $\rho(t)$ obtenue après 10^6 itérations.

Comme dans la section précédente les différentes normes considérées sont toutes des normes absolues. On remarque à nouveau sur la figure 5.31, qu'aucun des algorithmes ne converge vers la solution analytique car l'erreur sature a une valeur de 3.10^{-2}.

Remarque. La contrainte de positivité de ρ tend à être respectée au cours des itérations comme c'est le cas sur la figure 5.31. De plus, les variables ρ et m ayant des comportements

semblables nous ne représenterons que les courbes de convergence de ρ dans les exemples à venir.

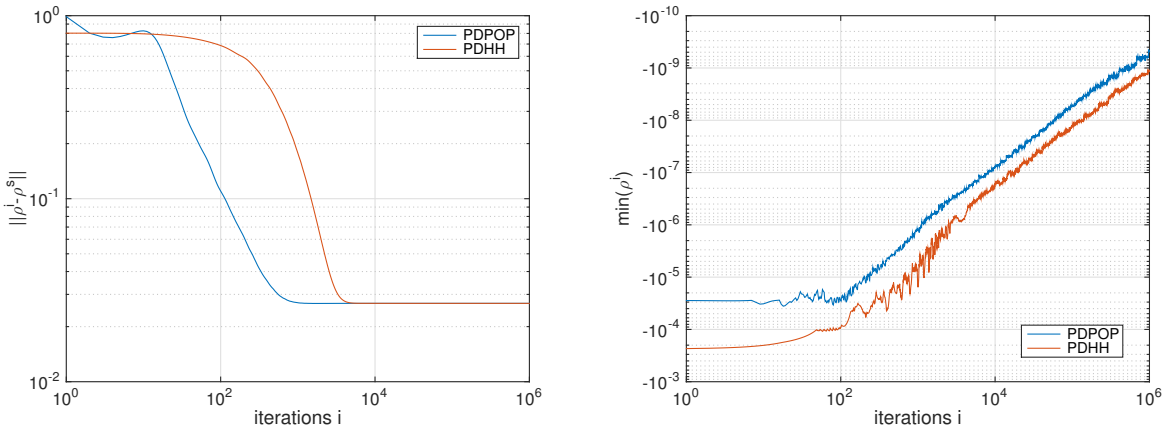


FIGURE 5.31: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s, et de la valeur minimum de ρ dans le cas de la figure 5.29.

Dans la suite, pour pouvoir comparer un peu mieux les différents algorithmes nous allons construire une solution de référence qui sera utilisée à la place de la solution analytique et qui est représentée en noir et gris sur la figure 5.30, le noir étant 0 et le blanc 1.

On commence donc par calculer une solution de référence (ρ^s, m^s) du problème discrétisé en exécutant pendant 10^6 itérations chaque algorithme que l'on souhaite étudier. On compare ensuite ces solutions avec la solution calculée analytiquement dans le tableau 5.4.

	$\|\|\rho^i - \rho^s\|\|$	$\|\|\rho^i - \rho^s\|\|_\infty$
$PDPOP$	0.01372571	**0.0220917**
$PDHH$	**0.01372560**	0.0220933

TABLE 5.4: Comparaison de l'erreur sur ρ après 10^6 itérations de chaque algorithme avec $\sigma = 70$ et $M = N = P = 64$ dans le cas de la figure 5.29.

On se rend compte que la solution de référence la plus proche en terme de norme L^2 de la solution analytique est celle calculée à l'aide de algorithme PDHH. Nous utiliserons dans la suite la solution calculée avec PDHH pour comparer les différents algorithmes.
Il est possible d'observer les différences entre les solutions de référence calculées par les différents algorithmes sur les images 5.32.

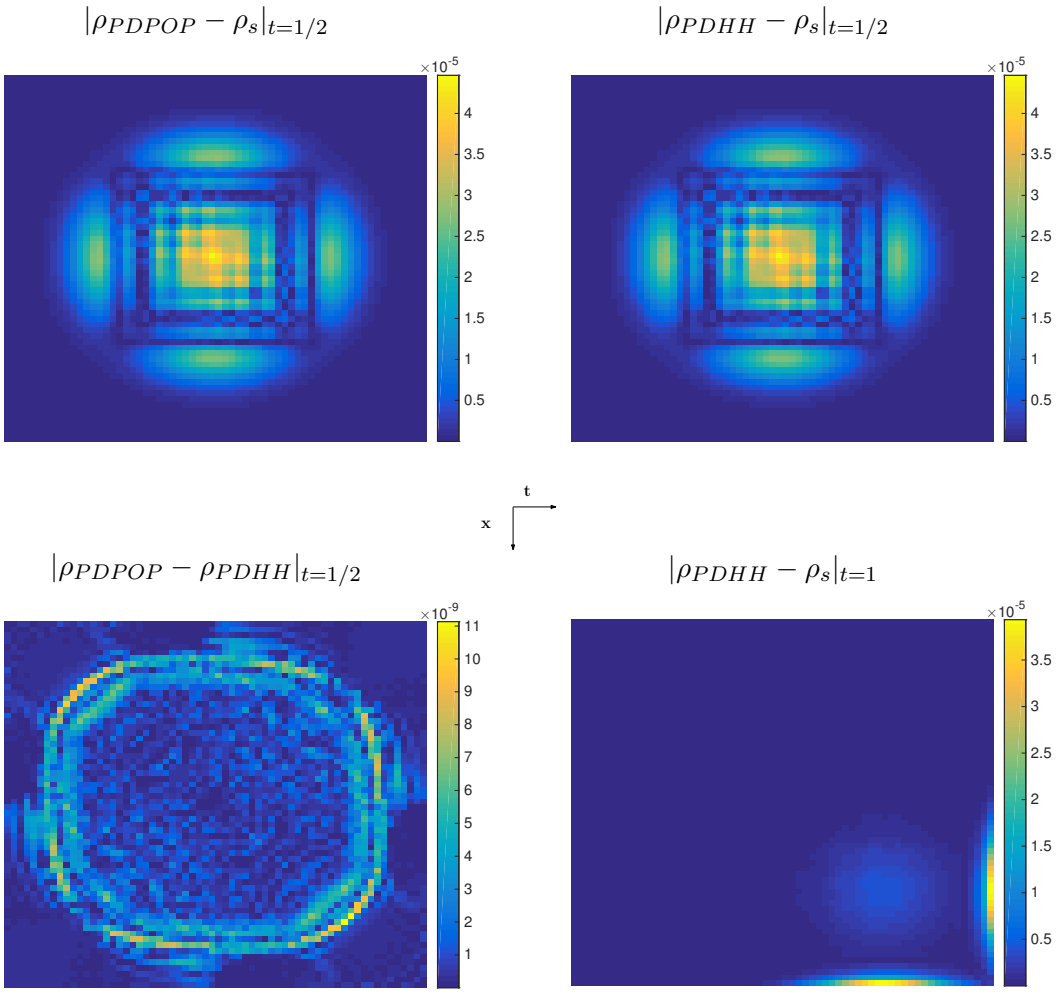


FIGURE 5.32: Comparaison entre les solutions calculées par les différents algorithmes dans le cas de la figure 5.29.

Comme on le remarque sur la figure ci-dessus, l'erreur entre les deux solutions calculées est en 10^{-9} et répartie sur presque l'ensemble du domaine.
Les erreurs par rapport à la solution analytique sont dues à deux phénomènes. Tout d'abord, elles sont dues à l'apparition d'oscillations le long des lignes de niveaux de ρ, comme dans le cas 1D. Par ailleurs, on observe une erreur importante sur le bord du domaine où ρ est supérieure à zéro et cette erreur se propage. Cela s'explique par les opérateurs adjoints que l'on doit utiliser dans les différents algorithmes et qui créeent sur le bord en x des discontinuités, plus ou moins importantes selon la valeur de ρ en ce bord.

Courbes de convergence pour différents σ

Pour comparer les convergences des différents algorithmes et étudier leurs performances, il faut tout d'abord choisir tous les paramètres correctement. Le seul paramètre à choisir sans condition particulière est σ, qui intervient dans la première étape de l'algorithme comme paramètre de descente le long de la fonctionnelle $\mathcal{J}$. Il n'y a pas d'étude théorique indiquant quel σ utiliser dans quelle condition. Nous allons donc comparer les courbes de convergence des différents algorithmes avec le même σ, pour observer son influence sur ces derniers. Les figures 5.33 et 5.34 nous montrent respectivement l'erreur sur ρ et la minimisation de la fonctionnelle $\mathcal{J}$ pour $\sigma = 20, 70, 100$ ou 130.

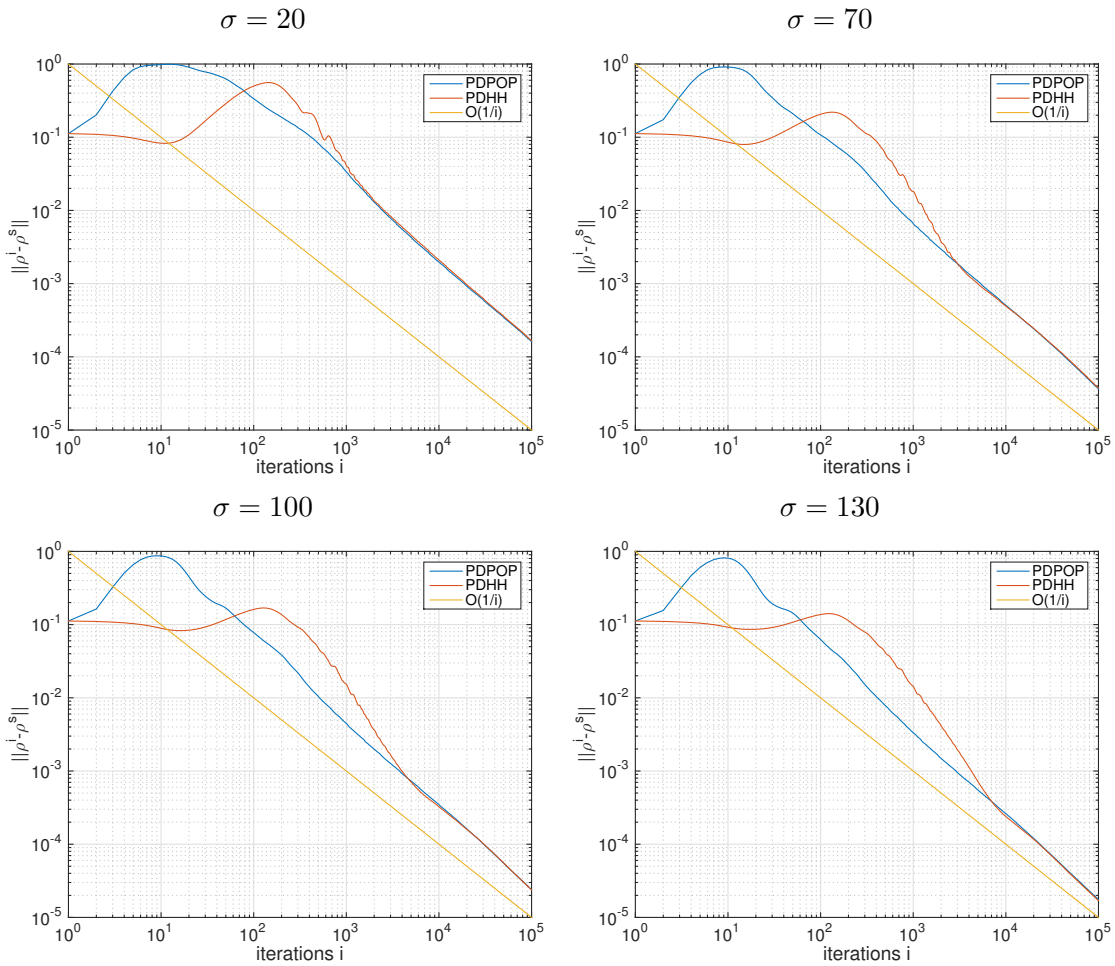


FIGURE 5.33: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s pour différents σ dans le cas de la figure 5.29.

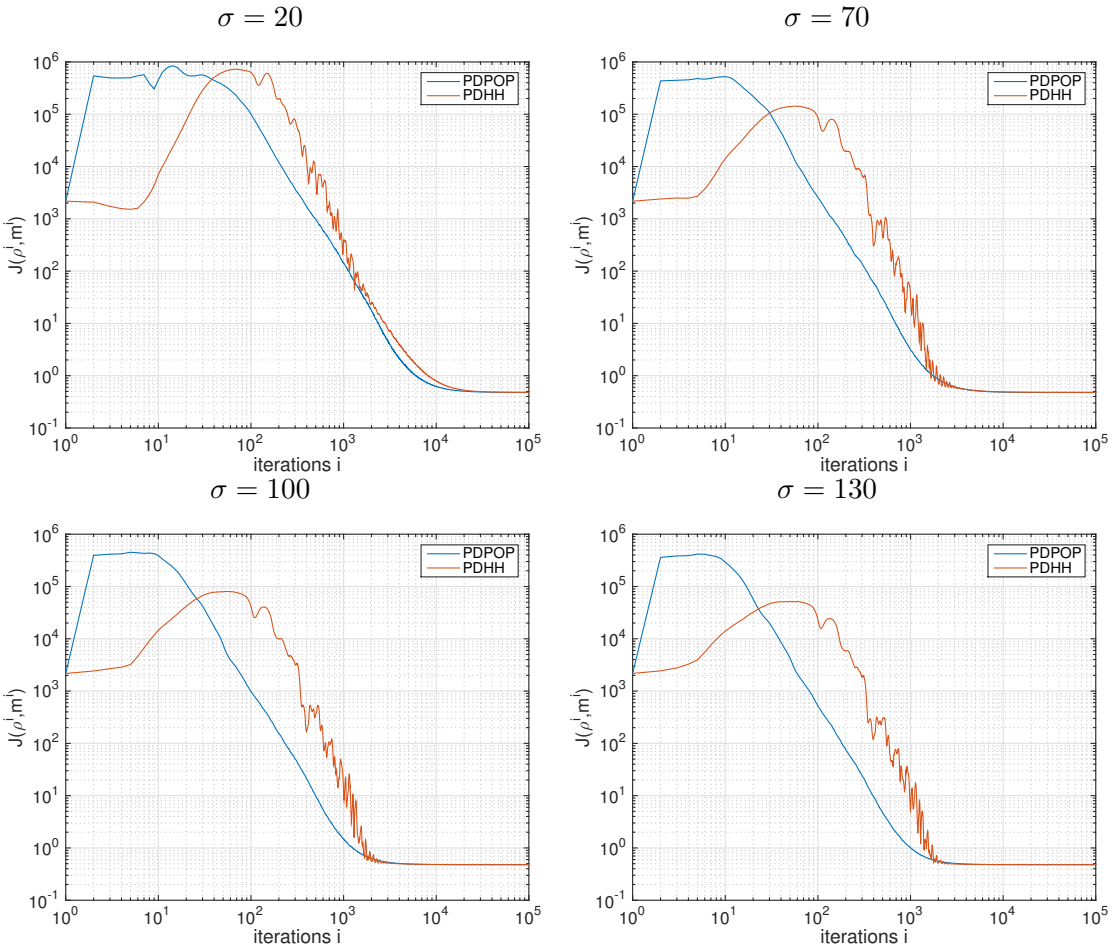


FIGURE 5.34: Comparaison à chaque itération de la fonctionnelle pour différents σ dans le cas de la figure 5.29.

En augmentant la valeur de σ, pour l'algorithme que l'on a développé, la convergence en norme L^2 de ρ est de plus en plus lente pendant les premières itérations puis de plus en plus rapide dans les itérations suivantes. De ce fait, si l'on souhaite obtenir une erreur grossière, il suffira d'utiliser un σ de petite valeur, alors que si l'on souhaite atteindre une erreur plus fine il faudra choisir un σ de valeur plus grande.

Dans la suite nous utiliserons $\sigma = 100$, car il représente un bon compromis pour l'étude que nous souhaitons faire de nos algorithmes. La convergence n'est pas trop lente pendant les premières itérations mais devient plutôt rapide par la suite.
En ce qui concerne la fonctionnelle $\mathcal{J}$, sa minimisation est plus rapide quand on augmente σ. Cela s'explique par le fait que σ est le paramètre de descente dans la première étape de l'algorithme primal-dual, qui est la projection sur le paraboloide induit par la

fonctionnelle $\mathcal{J}$. Or, plus il augmente plus τ, qui est le paramètre de descente de la seconde étape de l'algorithme diminue (car $\sigma\tau||K|| < 1$).

Remarque. Malheureusement, choisir un σ intéressant dans le cas de la gaussienne ne veut pas dire que ce σ sera le plus intéressant pour un autre cas. Le choix de sa valeur dépend de la taille du signal qui nous intéresse, des densités ρ_0 et ρ_1 considérées et de l'initialisation de l'algorithme.

Courbes de convergence pour différentes initialisations

Nous observons à présent l'influence de l'initialisation sur la convergence des différents algorithmes. Pour cela, sur la figure 5.35, nous utilisons comme initialisation :

— départ 0 : nous prenons $\phi^0 = 0$ comme initialisation,
— départ 1 : nous prenons ϕ^0 tel que l'erreur en ρ est de 10^{-1},
— départ 2 : nous prenons ϕ^0 tel que l'erreur en ρ est de 10^{-2}.

On remarque que l'algorithme PDPOP convergence de la même façon quelque soit l'initialisation choisie. L'algorithme PDHH en revanche est plus sensible au choix de l'initialisation. Pour le choix $\phi^0 = 0$, la convergence de ρ est plus lente alors que celle de $\mathcal{J}$ est plus rapide, ce qui est un comportement similaire à celui d'un choix de σ plus grand.

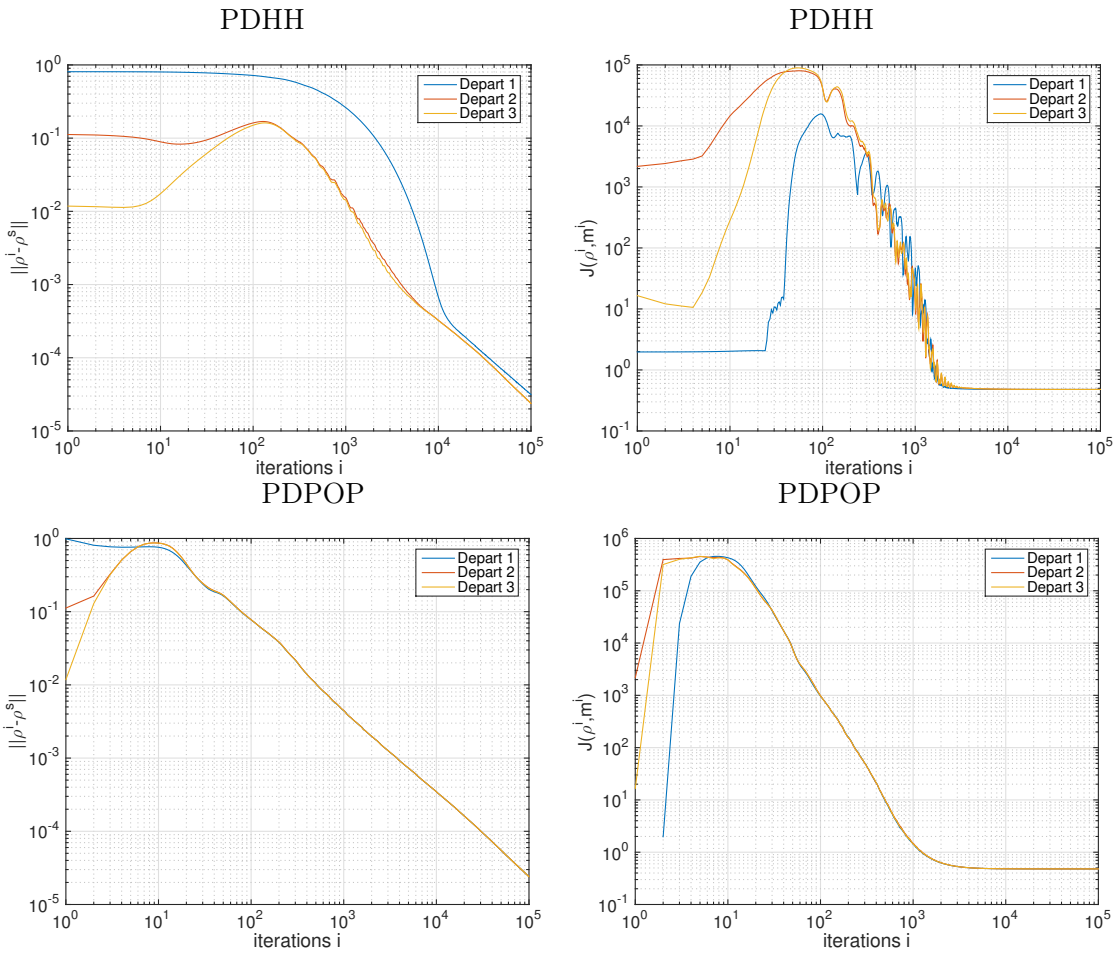


FIGURE 5.35: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s pour différentes initialisations dans le cas de la figure 5.29.

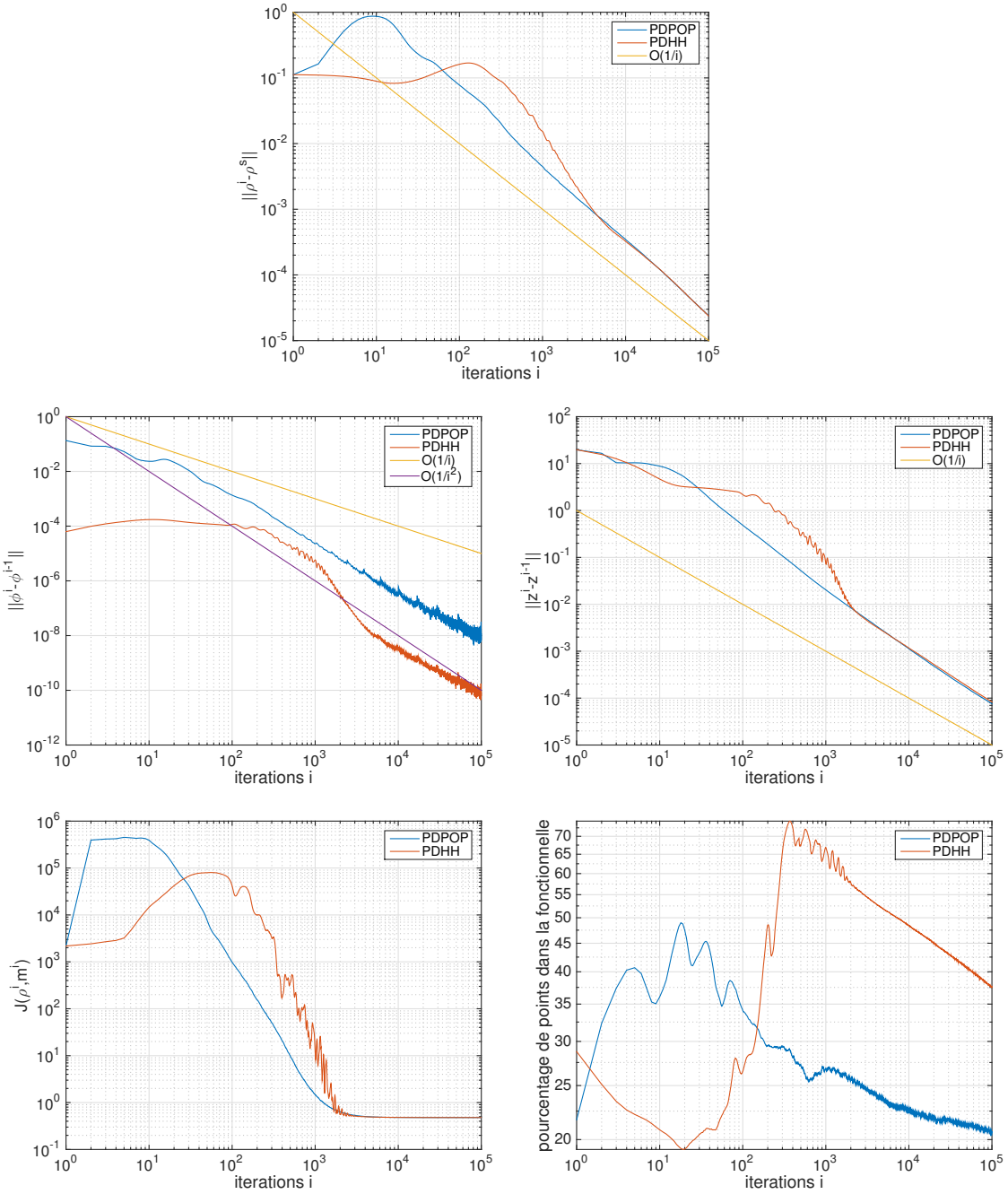


FIGURE 5.36: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s (haut), de l'erreur entre deux itérations de ϕ (milieu gauche), de z (milieu droit), de la minimisation de la fonctionnelle $\mathcal{J}$ (bas gauche) et du nombre de points restants dans le paraboloide après une itération (bas droit), dans le cas de la figure 5.29.

Convergence des variables primale et duale et projection sur le paraboloide

La dernière étude de convergence pour l'exemple des gaussiennes concerne la convergence des variables primale et duale des différents algorithmes, ainsi que l'évolution du nombre de points présents dans le paraboloide dont $\mathcal{J}^*$ est la fonction indicatrice (4.13), avant la projection au cours des itérations, comme représenté sur la figure 5.36.

On se rend compte qu'après 5000 itérations toutes les courbes de convergence de l'algorithme PDHH ont atteint un taux de convergence constant. La courbe de la fonctionnelle est alors horizontale et la différence entre deux itérations de ϕ est de 10^{-8} alors que celle en z est de 10^{-3}. Comme dans le cas 1D, on peut expliquer cette différence par la différence importante dans l'ordre de grandeur des paramètres associés aux deux étapes de l'algorithme. La première étape permet de mettre à jour ϕ et a pour paramètre $\sigma = 100$, quand la deuxième étape mettant à jour la variable z a pour paramètre $\tau = 3.10^{-7}$.

Par ailleurs, on observe que l'algorithme PDHH, restant dans la contrainte de divergence nulle, a bien plus de points qui restent dans la fonctionnelle à chaque itération mais que cela baisse au cours des itérations. En ce qui concerne l'algorithme PDPOP, moins de points restent dans le paraboloide après une itération, et cela baisse au cours des itérations. Après 100 000 itérations, 37% des points ($\sim$98 422 points) sont dans la fonctionnelle dans le cas de l'algorihme PDHH et 21% ($\sim$54 076 points) pour l'algorithme PDPOP. Ce point est important car la projection sur le paraboloide est très coûteuse et moins il y a de points dans le paraboloide plus il faut en projeter.

Complexité

L'étude de la complexité est la même qu'en 1D. La forme générale de l'algorithme est :

Algorithme primal-dual

— Initialisation : $\tau, \sigma > 0, \theta \in [0,1]$, $(\phi^0, z^0 = K\phi^0, \tilde{\phi}^0 = \phi^0)$.

— Itérations :

$$\begin{aligned} z^{i+1} &= \text{prox}_{\mathcal{J}^*}(z^i + \sigma(K\tilde{\phi}^i)) \\ \phi^{i+1} &= \text{prox}_{\iota_{C_0}}(\phi^i - \tau K^* z^{i+1}) \\ \tilde{\phi}^{i+1} &= \phi^{i+1} + \theta(\phi^{i+1} - \phi^i). \end{aligned}$$

Dans la suite, N représente le nombre de points de discrétisation en temps, qui est le même dans les deux directions en espace et ainsi N^3 est le nombre de points de la grille centrée G^{c1} (5.1) :

— la première étape est une projection point par point pour tous les algorithmes utilisés et est en $O(N^3)$,

— la **deuxième étape pour l'algorithme PDPOP** est une projection sur la contrainte de divergence nulle et donc la résolution d'une équation de Poisson

en 3 dimensions en utilisant une transformée en cosinus rapide de complexité $O(N^3 log(N))$.
— la **deuxième étape pour l'algorithme PDHH** consiste à mettre les bonnes valeurs sur le bord car nous sommes déjà à divergence nulle et la complexité est en $O(N^2)$.

Ainsi les algorithmes que nous avons développés ont une complexité numérique linéaire en le nombre de points total de discrétisation, N^3.

La figure 5.37 représente la complexité numérique des différents algorithmes étudiés dans cette thèse ainsi que de certaines des fonctions les composant. Le temps CPU nécessaire pour une itération des différents algorithmes ou pour les fonctions les constituant, est représenté en fonction du nombre de point de discrétisation dans une direction, N.

Ce que nous dénoterons PDPOP^{gh} est l'algorithme PDPOP développé par Peyre *et al.* et disponible sur `https://github.com/gpeyre/2013-SIIMS-ot-splitting`. Il utilise une implémentation de la projection sur le paraboloide différente de celle que nous avons mise en place dans les algorithmes PDPOP et PDHH. On peut ainsi en comparant la complexité numérique de PDPOP à celle de PDHH voir l'influence de notre nouvelle formulation du problème de transport optimal, car la seule différence entre ces deux algorithme est la descente sur (ρ, m) (PDPOP) ou sur $\nabla \times \phi + \nabla h$ (PDHH). En comparant PDPOP^{gh} et PDHH on observe également l'accélération de notre implémentation par rapport à l'état de l'art.

Les calculs ont été effectués sur un bi-processeur Intel Core i7 avec 2.8GHz.
Nous dénoterons dans la figure 5.37 par K_{PDPOP} l'opérateur K utilisé dans l'algorithme $PDPOP$ et par K_{PDHH} celui utilisé dans l'algorithme PDHH. Ce que nous dénotons $proj_{div}$ est la projection sur la contrainte de divergence nulle qui correspond à la seconde étape de l'algorithme PDPOP, c'est-à-dire la résolution d'une équation de Poisson en trois dimensions.

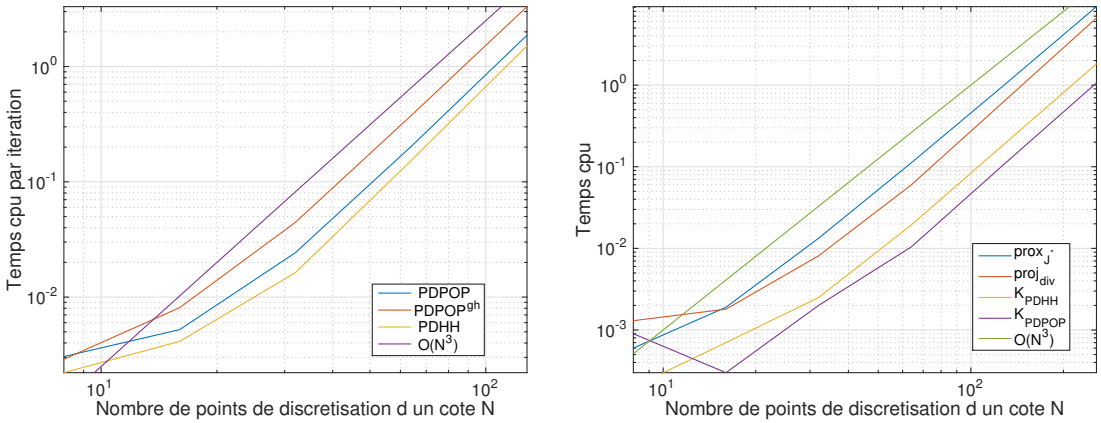


FIGURE 5.37: Comparaison des complexités numériques des différents algorithmes (gauche) et des fonctions les composant (droite) dans le cas de la figure 5.29.

En moyenne, on observe que PDHH est une accélération de
- 25% de l'algorithme PDPOP.
- 56% de l'algorithme PDPOP^{gh}.

Les performances de l'algorithme PDHH s'expliquent par l'absence de projection sur la contrainte de divergence nulle. De plus, comme on l'a vu dans la figure 5.36, cet algorithme nécessite la projection de moins de points que l'algorithme PDPOP ce qui est coûteux. On perd en revanche un peu de temps à cause de notre opérateur K_{PDHH} qui comprend un rotationnel et une interpolation alors que l'opérateur K_{PDPOP} utilisé dans l'algorithme PDPOP est une interpolation. Par ailleurs, l'opérateur K_{PDPOP} est 18% plus rapide que l'opérateur K_{PDHH} en 1D alors qu'il est 41% plus rapide en 2D ce qui explique que notre algorithme ait des performences moins bonnes en deux dimensions. On l'observe sur la figure 5.37 (gauche).

Conclusion pour l'exemple des gaussiennes.

Nous terminons ce paragraphe avec deux tableaux. Le premier montre le nombre d'itérations et le temps CPU nécessaires pour atteindre une erreur donnée en ρ, pour les différents algorithmes. Le second montre l'accélération de l'algorithme que nous avons développé pour cet exemple en comparaison avec les algorithmes PDPOP et PDPOP^{gh} proposés par [69].
Nous remarquons que l'algorithme PDHH nécessite un plus grand nombre d'itérations pour converger que l'algorithme PDPOP, quelque soit l'erreur souhaitée dans cet exemple, comme on peut le voir sur la figure 5.36. Cependant, la rapidité d'exécution de cet algorithme, comme mis en avant sur la figure 5.37 compense ce problème pour tous les algorithmes pour les erreurs 10^{-3} et 10^{-4} et même pour l'erreur 10^{-2} dans le cas de l'algorithme PDPOP^{gh}.

$\|\|\rho^i - \rho^s\|\|$	PDPOP^{gh}	$PDPOP$	$PDHH$
10^{-2}	514 (3'31")	514 (1'54")	1243 (3'11")
10^{-3}	3 761 (25'46")	3761 (14'04")	3 985 (10'10")
10^{-4}	30 349 (3 :27'30")	30 349 (1 :54'56")	30 569 (1 :21'56")

Nombre d'itérations et temps CPU pour une erreur donnée

$\|\|\rho^i - \rho^s\|\|$	$PDHH$%PDPOP^{gh}	$PDHH$%$PDPOP$
10^{-2}	9%	x
10^{-3}	61%	28%
10^{-4}	61%	29%

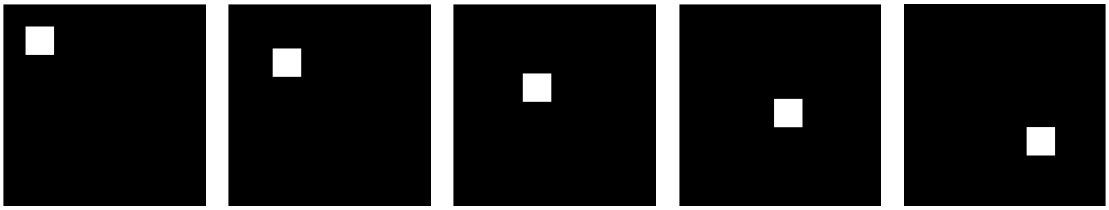
Accélération pour une erreur donnée

TABLE 5.5: Évaluation des performances pour une erreur donnée dans le cas de la figure 5.29. Dans le second tableau, les 'x' représentent une absence d'accélération.

Tests avec d'autres densités ρ_0 et ρ_1

Densités irrégulières Nous allons maintenant étudier le transport entre deux densités non régulières comme le montre la figure 5.38, pour lesquelles, comme dans le cas des gaussiennes, nous pouvons calculer la solution analytique. On choisit $\rho_0(x, y) = \rho_1(x + 0.5, y + 0.5)$ ce qui donne à nouveau

$$\rho^s(t, x, y) = \rho_0(x - 0.5t, y - 0.5t) \text{ et } d_2(\rho_0, \rho_1) = \sqrt{|0.5|^2 + |0.5|^2} = 1/\sqrt{2}.$$



t = 0 t = 0.25 t = 0.5 t = 0.75 t = 1

FIGURE 5.38: Représentation de la densité $\rho(t)$ obtenue après 10^6 itérations.

On observe dans le tableau suivant que l'algorithme ne converge pas vers la solution analytique, ce qui est dû à l'apparition d'oscillations, importantes au niveau des irrégularités, comme on le remarque sur les figures 5.39. Dans la suite, pour comparer un peu mieux les différents algorithmes, nous allons construire une solution de référence qui sera utilisée à la place de la solution analytique. On commence donc par calculer une solution de référence (ρ^s, m^s) du problème discrétisé en faisant tourner pendant 10^6 itérations chaque algorithme à étudier. On compare ensuite ces solutions avec la solution calculée analytiquement dans le tableau 5.6.

	$\|\|\rho^i - \rho^s\|\|$	$\|\|\rho^i - \rho^s\|\|_\infty$
$PDPOP$	0.2948891	0.82144
$PDHH$	**0.2948835**	**0.82139**

TABLE 5.6: Comparaison de l'erreur par rapport à la solution analytique après 10^6 itérations de chaque algorithme avec $\sigma = 100$ et $N = M = 64$ dans le cas de la figure 5.38.

On se rend compte que la solution de référence la plus proche de la solution analytique est celle calculée à l'aide de l'algorithme PDHH.
On observe les différences entre les solutions de référence calculées par les différents algorithmes sur les images 5.39.

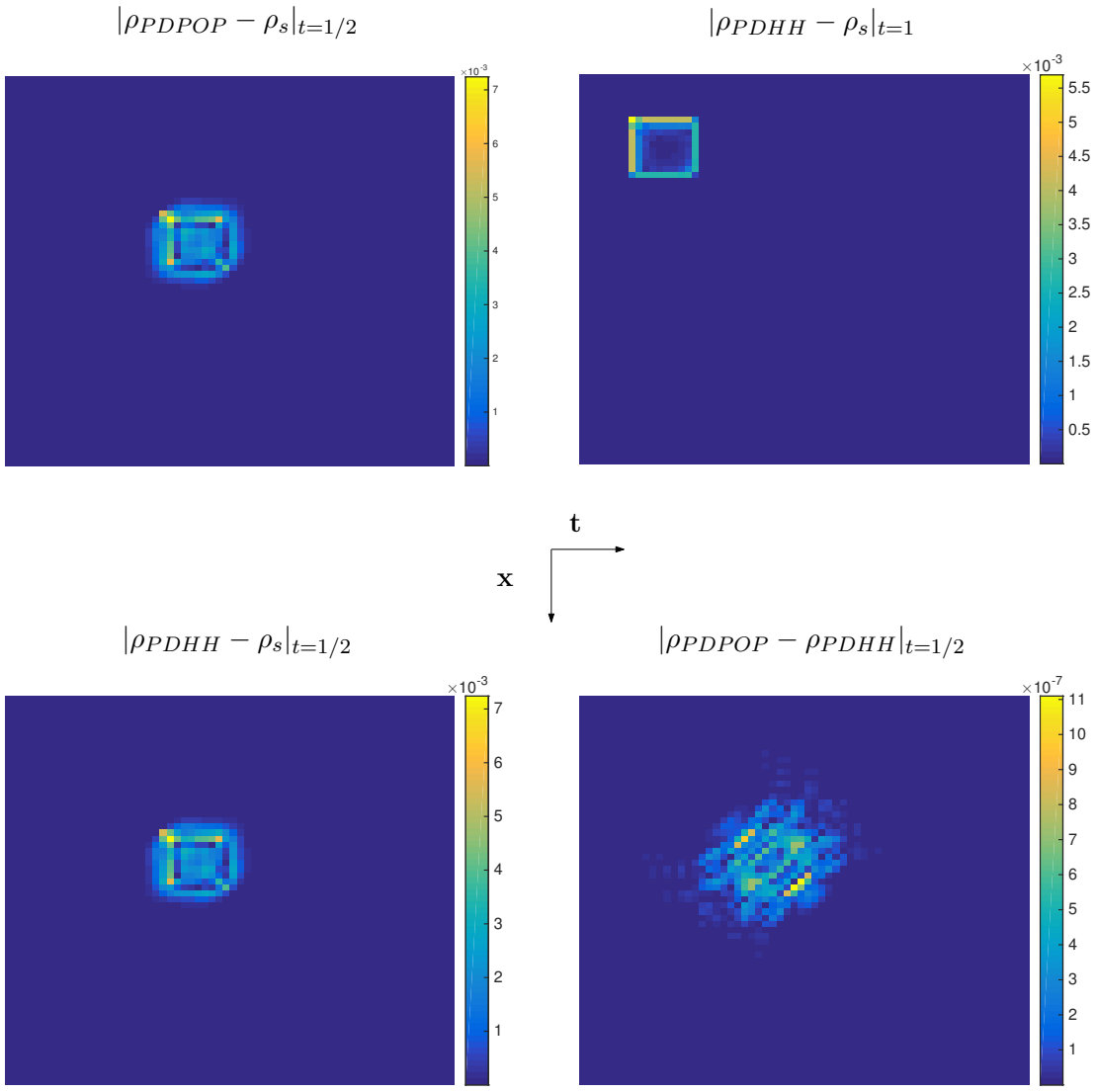


FIGURE 5.39: Comparaison entre les différentes solutions calculées par les différents algorithmes.

Comme précédemment, on remarque sur la figure ci-dessus que l'erreur entre deux solutions calculées en minimisant la même fonctionnelle est en 10^{-7}, que l'on utilise pour représenter le couple (ρ, m) la décomposition de Helmholtz-Hodge $(\nabla \times \phi + \nabla h)$ ou aucune décomposition. En revanche, l'erreur entre un des deux algorithmes et la solution analytique est en 10^{-3}.
L'erreur par rapport à la solution analytique est due à l'apparition d'oscillations le long des lignes de niveaux de ρ. Par ailleurs, les oscillations deviennent plus importantes aux points d'irrégularité de la solution.

Les courbes de la figure 5.40 sont obtenues en gardant les mêmes paramètres que dans l'exemple des gaussiennes et, par conséquent, nos remarques sur le lien entre les convergences de ϕ et z et les valeurs des paramètres σ et τ sont toujours valables ici. Cependant, on remarque que le comportement des courbes est légèrement différent. Ainsi, bien que les erreurs soient les mêmes pour les itérations 1 et 10^5, la convergence est plus lente pendant les premières itérations puis plus rapide pendant les suivantes, ce qui ressemble au comportement que l'on obtient quand on augmente σ.

Enfin, on observe que l'algorithme PDHH a un peu plus de points qui restent dans la fonctionnelle à chaque itération mais que la différence est moitié moins importante que dans l'exemple des gaussiennes (figure 5.36). Comme précédemment, ce nombre baisse au cours des itérations pour l'algorithme PDHH. Le comportement de l'algorithme PDPOP est quant à lui différent puisque le nombre de points dans la fonctionnelle diminue puis augmente. À la 10^5 ième itération, 92% des points ($\sim$240 876 points) sont dans la fonctionnelle dans le cas de l'algorihme PDHH, 84% ($\sim$221 163 points) dans le cas de l'algorithme PDPOP.

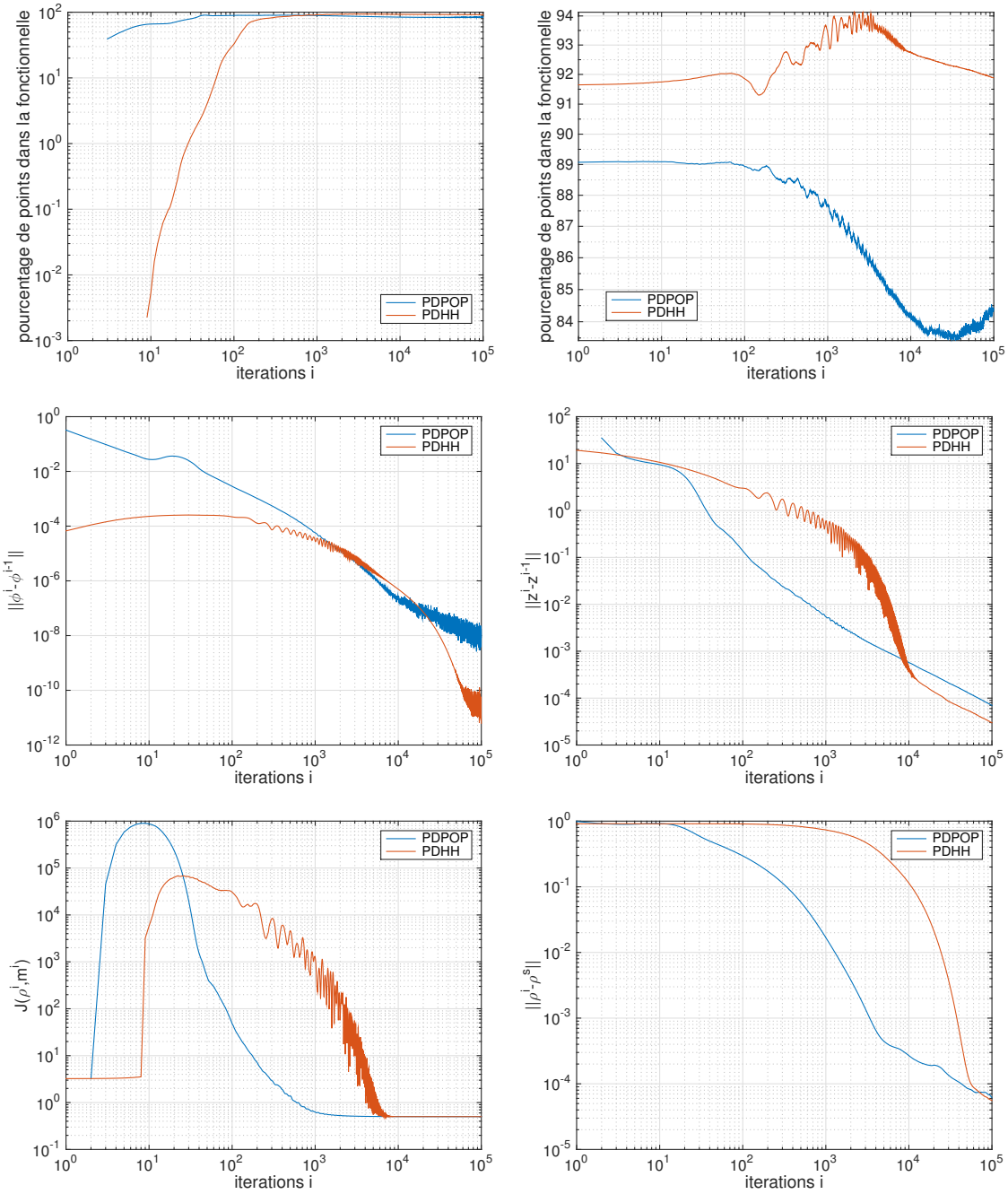


FIGURE 5.40: Comparaison à chaque itération du nombre de points restants dans le paraboloide après une itération (haut gauche) et zoom sur la figure de gauche, de l'erreur entre deux itérations de ϕ (milieu gauche), de z (milieu droit), de la minimisation de la fonctionnelle $\mathcal{J}$ (bas gauche) et de l'erreur L^2 entre ρ et ρ^s (bas droit) , dans le cas de la figure 5.38.

Test pour deux densités lisses et nulles sur le bord. Dans le premier exemple, si l'on prolonge ρ_0 et ρ_1 sur $\mathbb{R}$ par zéro on se rend compte d'une petite irrégularité en $t = 0$ pour ρ_0 et en $t = 1$ pour ρ_1. Nous reprenons ici ce premier exemple mais nous "éloignons" ces gaussiennes du bord du domaine, pour obtenir des densités ρ_0 et ρ_1 lisses. On représente les courbes de convergence de ρ par rapport à la solution analytique, pour $\sigma = 100$, pour les différents algorithmes sur la figure 5.41.

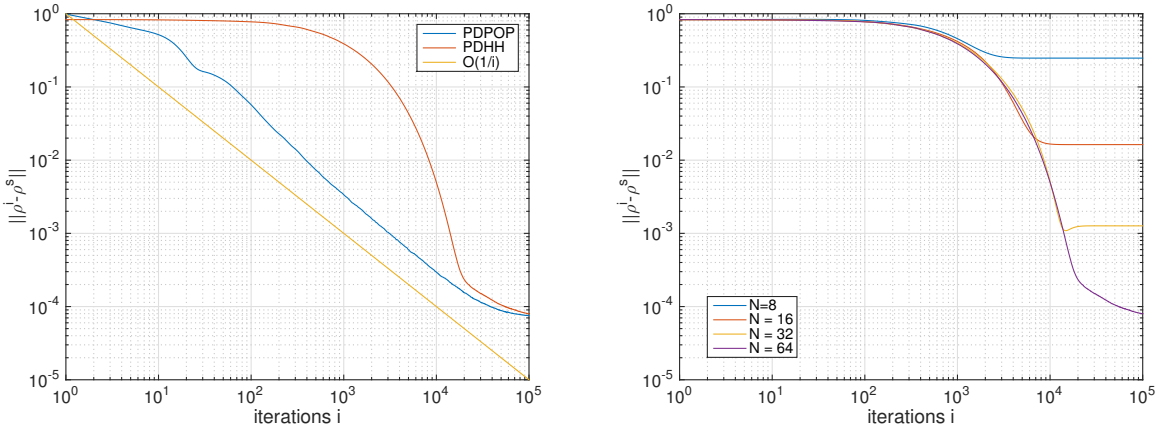


FIGURE 5.41: Comparaison à chaque itération de l'erreur L^2 entre ρ et ρ^s pour une grille de 64^3 points (gauche) et de l'erreur L^2 entre ρ et ρ^s pour l'algorithme PDHH et différentes tailles de grille.

Comme on peut le remarquer les algorithmes convergent de façon satisfaisante vers la solution analytique.

Nous observons à présent sur la figure 5.42 les différences entre les solutions de référence calculées après 10^6 itérations et la solution analytique. On remarque comme précédemment que les erreurs sont des oscillations le long des lignes de niveaux de ρ ; cependant, l'erreur qui provenait du bord en $x = 0$ et $x = 1$ à $t = 0$ et $t = 1$ pour le premier exemple avec des gaussiennes est bien moins importante car la valeur de ρ_0 et ρ_1 au bord est plus faible.

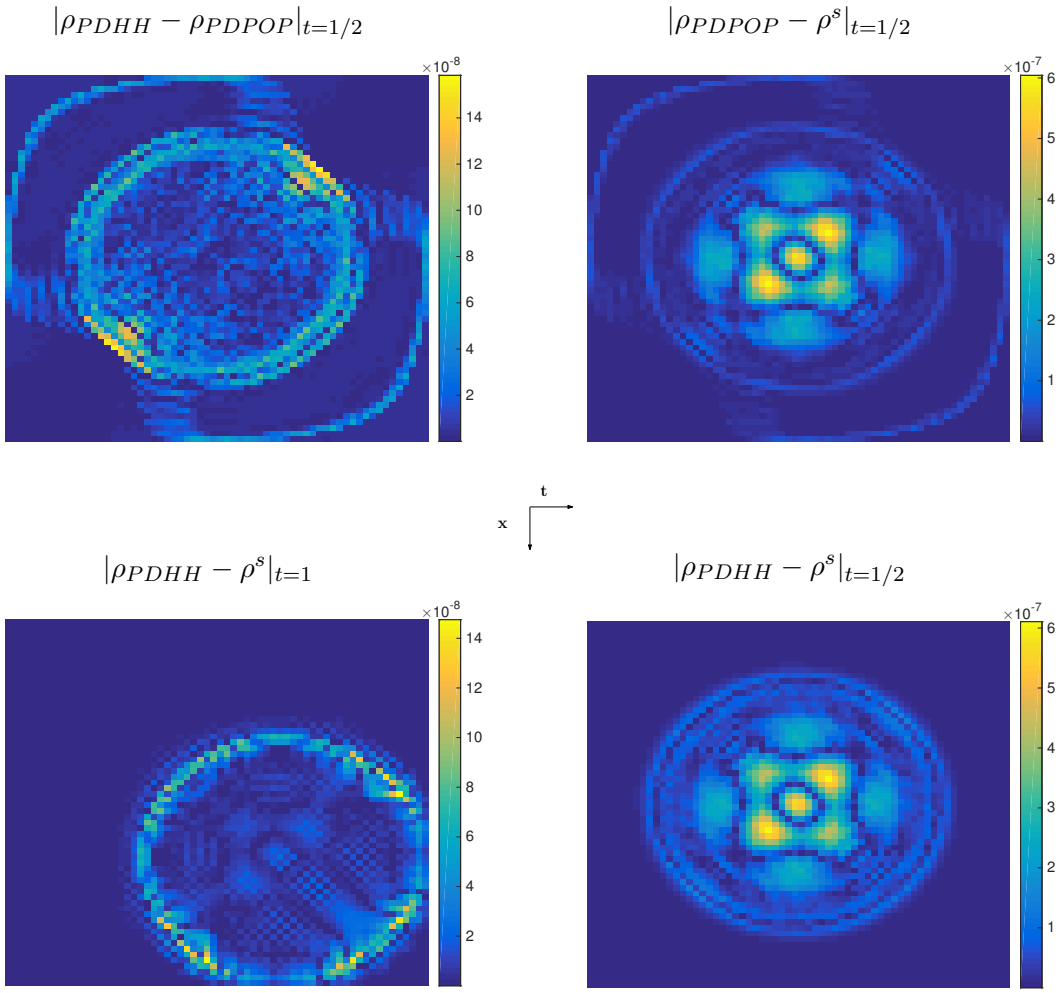


FIGURE 5.42: Comparaison entre les différentes solutions de référence calculées par les différents algorithmes et la solution analytique.

Test quand on privilégie une direction. Dans la section 3.2.2, nous avons discuté de la possibilité d'annuler une des composantes de ϕ dans l'algorithme PDHH (algorithme 9) développé pour des images 2D, qui consiste en la minimisation de la fonctionnelle $\mathcal{J}_h$ en utilisant la décomposition de Helmholtz-Hodge $(\rho, m) = \nabla \times \phi + \nabla h$. Nous illustrons dans la figure 5.43 la densité $\rho(t)$ obtenue avec l'algorithme quand on annule la première composante de ϕ, la deuxième ou la dernière.

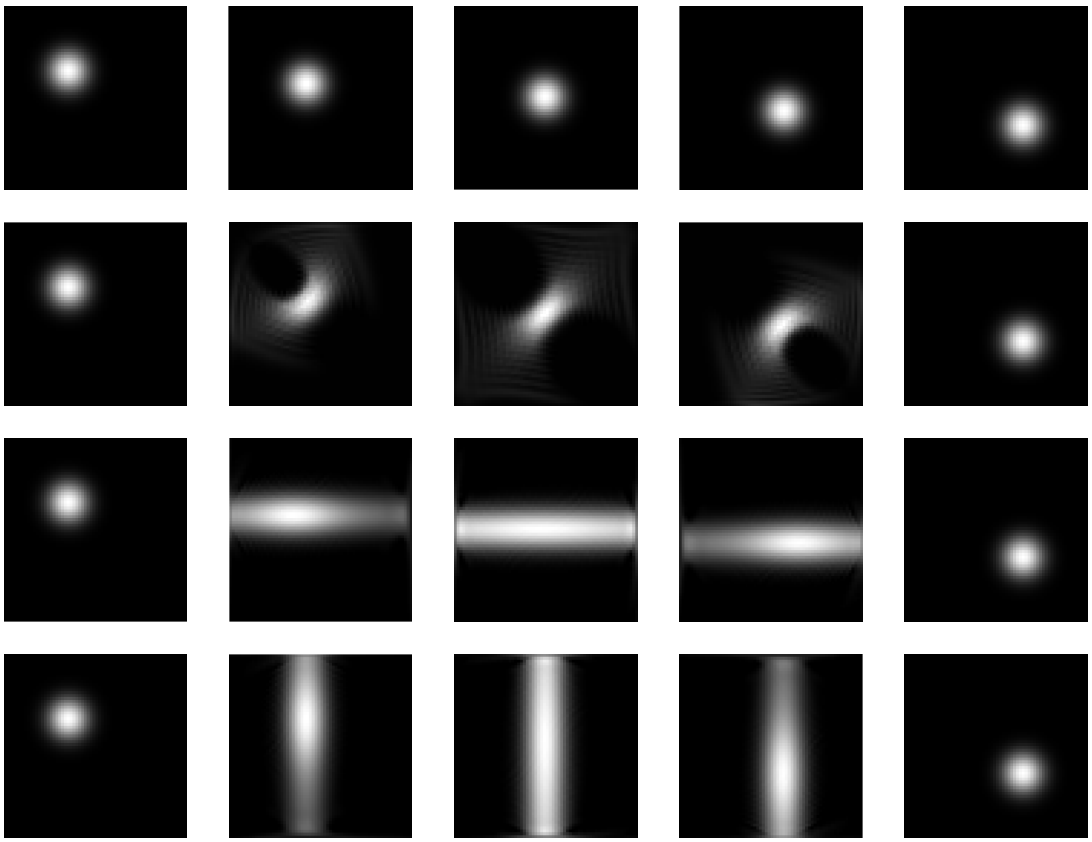


FIGURE 5.43: Représentation de la densité $\rho(t)$ obtenue avec l'algorithme PDHH (première ligne) quand on annule la première composante de ϕ (deuxième ligne), la deuxième (troisième ligne) ou la dernière (quatrième ligne).

On observe bien les directions privilégiées selon la composante de ϕ annulée comme abordé dans la section 3.2.2.

5.3 Implémentation en C++

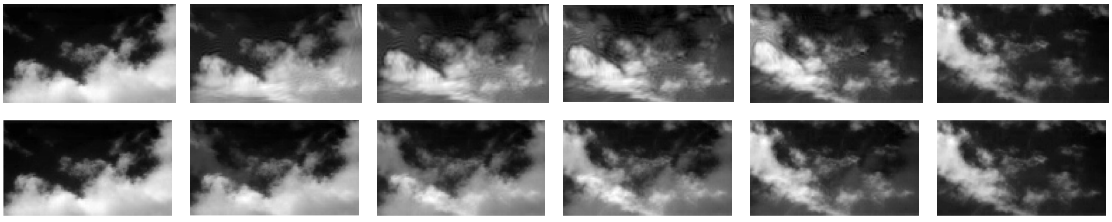
L'implémentation à l'aide de Matlab devient rapidement très lente et ne peut être utilisée pour des images de tailles trop grandes. Pour pallier ce problème nous avons implémenté l'algorithme PDHH pour des images en 2D en C++ en utilisant le code développé par Nicolas Boneel, parallélisé à l'aide d'OpenMP, que l'on peut télécharger sur la page `http://liris.cnrs.fr/ nbonneel/FastTransport/`.
Si on considère à nouveau l'exemple du tableau 5.5 et que l'on considère maintenant le temps réel (`wall time`) d'exécution des implémentations Matlab et C++ pour l'algorithme PDHH on obtient :

$\|\|\rho^i - \rho^s\|\|$	Matlab	C++	Accélération
10^{-2}	2'37"	13"	×12
10^{-3}	8'10"	40"	×12
10^{-4}	1 :27'14"	5'44"	×15

TABLE 5.7: Comparaison du temps d'exécution des codes C++ et Matlab dans le cas de l'algorithme PDHH.

5.4 Applications aux images réelles

Nous représentons sur la figure 5.44 le transport optimal d'une image $\rho_0(x)$ représentant des nuages, sur une image ρ_1 de même nature, toutes les deux de taille 120×68, ainsi que l'interpolation L^2 de ces deux images.



t = 0 t = 0.2 t = 0.4 t = 0.6 t = 0.8 t = 1

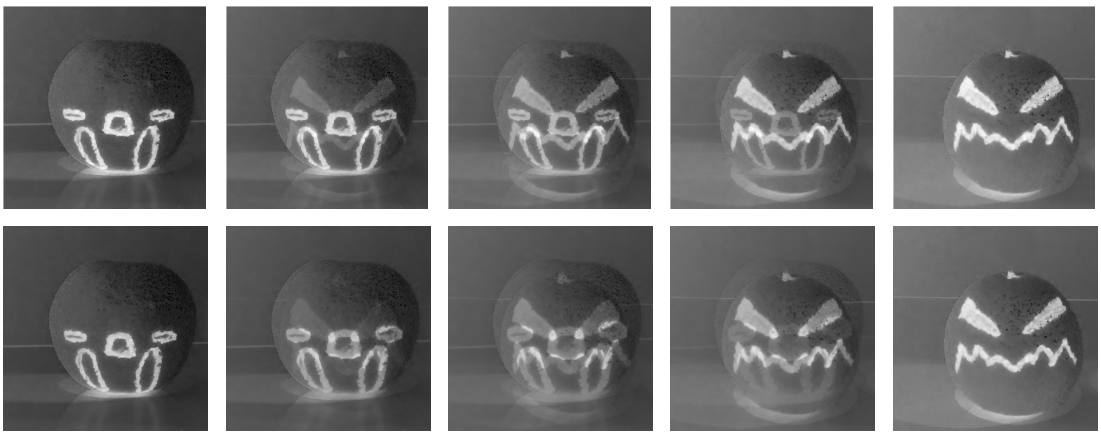
FIGURE 5.44: La première ligne représente la densité issue du transport optimal de ρ_0 sur ρ_1. La seconde ligne représente l'interpolation L^2 entre ρ_0 et ρ_1.

Nous représentons sur la figure 5.45 le transport optimal de la photographie d'une mandarine décorée sur la photographie d'une autre mandarine décorée, de taille 628×628, ainsi que l'interpolation L^2 de ces deux images.

t = 0 t = 0.25 t = 0.5 t = 0.75 t = 1

FIGURE 5.45: La première ligne représente la densité issue du transport optimal de ρ_0 sur ρ_1. La seconde ligne représente l'interpolation L^2 entre ρ_0 et ρ_1.

Nous représentons ensuite le transport de ces deux mêmes photographies quand on inverse leurs densités, figure 5.46.



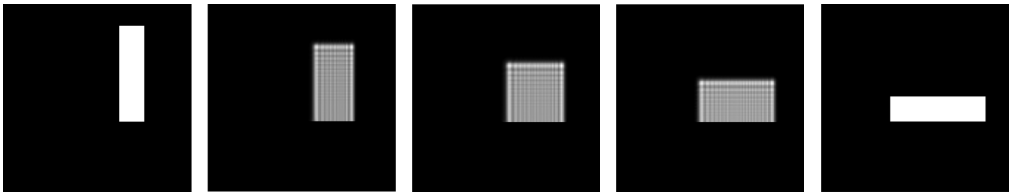
t = 0 t = 0.25 t = 0.5 t = 0.75 t = 1

FIGURE 5.46: La première ligne représente la densité issue du transport optimal de ρ_0 sur ρ_1. La seconde ligne représente l'interpolation L^2 entre ρ_0 et ρ_1.

Le comportement que l'on observe peut être expliqué par les remarques suivantes, liées au fait que l'on utilise une formulation classique du transport optimal, qui minimise une fonctionnelle sur l'ensemble des images et donc est globale et non locale.

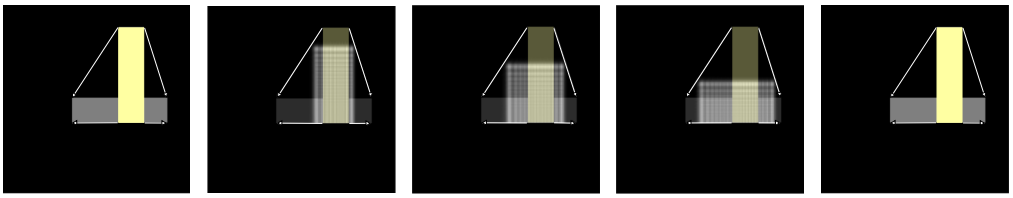
5.4.1 Comportements observés lors du transport de deux images

— Le transport s'effectue en lignes droites qui ne se croisent pas et donc ne respecte pas toujours la nature de ce qui se trouve dans l'image. Dans l'exemple ci-dessous on peut voir que le transport d'un rectangle sur un rectangle peut faire apparaitre un carré.



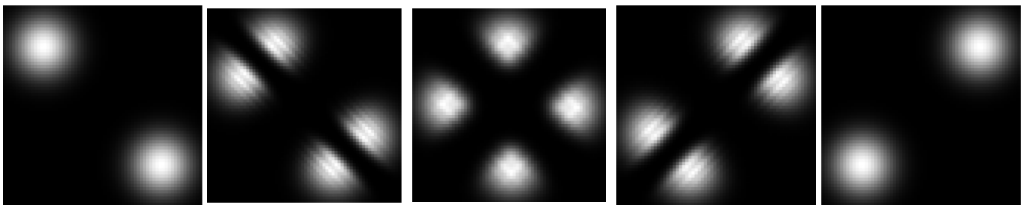
t = 0 t = 0.2 t = 0.4 t = 0.6 t = 0.8

Dans l'image ci-dessous nous superposons à chaque instant ρ_0, ρ_1 et $\rho(t)$.



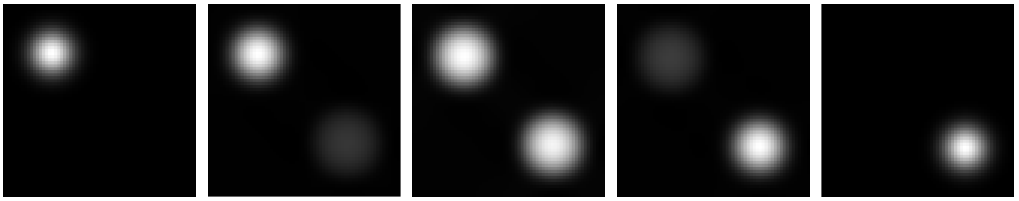
t = 0 t = 0.2 t = 0.4 t = 0.6 t = 0.8

— Cela peut également entrainer des divisions dans les objets représentés, comme dans le transport de deux gaussiennes sur deux autres gaussiennes de même variance.



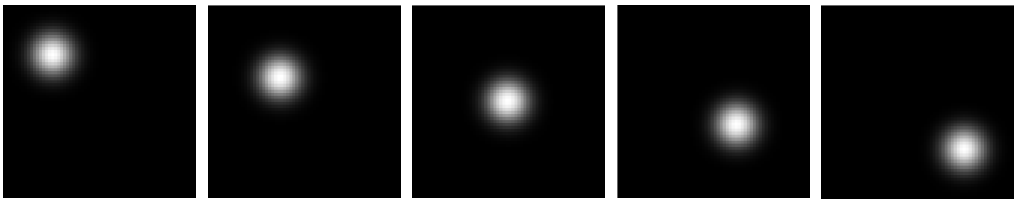
t = 0 t = 0.25 t = 0.5 t = 0.75 t = 1

— Même si l'arrière plan de l'image ne nous intéresse pas il est très souvent, dans des images réelles, non nul et donc aura un impact sur le transport de tout ce qui se trouve dans l'image car le transport se fait en analysant des informations globales. Nous transportons dans l'exemple ci-dessous une gaussienne à laquelle nous avons ajouté une constante pour que toute la partie noire de l'image ne soit pas de valeur nulle.



t = 0 t = 0.25 t = 0.5 t = 0.75 t = 1

Nous représentons à présent le même transport sans la constante et donc la partie noire vaut zéro.



t = 0 t = 0.25 t = 0.5 t = 0.75 t = 1

Il faut donc appliquer des pré traitements aux images que l'on souhaite étudier à l'aide de cette formulation pour obtenir des résultats plus proches de ce que l'on souhaite obtenir, comme un seuillage ou un découpage de l'image pour ne prendre en compte que ce que l'on souhaite déplacer.

Chapitre 6

Conclusion

Nous avons développé dans cette thèse de nouvelles formulations pour le problème du transport optimal dynamique, permettant de travailler directement dans l'espace des contraintes. Nous avons également prouvé que la fonctionnelle à minimiser dans ce problème a de meilleures propriétés de convexité dans l'espace des contraintes. Pour travailler dans cet ensemble contenant une contrainte de divergence nulle, nous avons considéré trois formulations.
La première implique l'utilisation d'ondelettes à divergence nulle et nous l'avons implémenté pour des ondelettes périodiques. L'algorithme obtenu converge lentement à cause de la non périodicité des vecteurs décomposés, mais démontre la faisabilité de la méthode et permet d'envisager l'utilisation d'ondelettes respectant les conditions au bord comme nous l'avons détaillé dans le chapitre 4.
L'étude de cette première formulation nous a menés à nous intéresser aux représentations par fonction de courant munies de relèvement au bord ainsi qu'à l'utilisation de la décomposition de Helmholtz-Hodge. Nous avons prouvé l'équivalence pour des signaux 1D entre la résolution d'une équation de type courbure minimale sur chaque ligne de niveau du potentiel muni des conditions de Dirichlet appropriées, et la minimisation de la fonctionnelle issue du problème de transport optimal.
Ces formulations donnent des algorithmes performants et simples à implémenter, à l'aide de l'algorithme primal-dual développé par Chambolle et Pock. Dans le cas des images 2D, la décomposition de Helmholtz-Hodge donne également lieu à des algorithmes plus performants que l'état de l'art, que nous avons développés en C++ et parallélisés avec OpenMP. Cette méthode est utilisable pour des images de densités différentes ou à support compact.

Ces travaux de thèse ont des perspectives naturelles dans les différentes directions suivantes :

- Améliorer l'implémentation des algorithmes impliquant les bases d'ondelettes à divergence nulle en utilisant les ondelettes respectant les conditions sur le bord.
- Mettre en oeuvre des algorithmes plus performants que l'algorithme de gradient

et l'algorithme primal-dual étudiés.
— Utiliser de nouvelles formulations respectant la contrainte de divergence nulle telle que celle impliquant les variables de Clebsch et donc permettant une représentation pour des images en plus grande dimension (images 3D ou 4D).
— Intégrer nos nouvelles formulations dans des algorithmes plus complexes pour des densités de masses différentes ou respectant la physique des objets étudiés.

Annexe A

Calcul de quelques opérateurs proximaux

A.1 Pour $\mathcal{J}_{ms} = ||.||_1$

Nous voulons calculer l'opérateur proximal de la fonction $\mathcal{J}_{ms}$ définie par $\mathcal{J}_{ms}(x) = ||x||_1$. Sur la grille discrète G^c sur laquelle on se place, définie dans la section 5.1, $||.||_1$ s'écrit

$$||x||_1 = \sum_{k \in G^c} |x_k| \text{ où } |x_k| = \sqrt{(x_k^1)^2 + (x_k^2)^2},$$

où nous noterons $F(x) = |x|$. Par conséquent nous avons,

$$\text{prox}_{\sigma \mathcal{J}_{ms}}(x) = \text{prox}_{\sigma||.||_1}(x) = (\text{prox}_{\sigma|.|}\, x_k)_{k \in G^c}.$$

On cherche donc

$$y' = \text{prox}_F(x) = \underset{y}{\text{argmin}} |x - y|^2 + F(y) = \underset{y}{\text{argmin}} |x - y|^2 + |y|. \tag{A.1}$$

Pour $|y'| \neq 0$

$$\begin{aligned} y' = \text{argmin}_y \frac{1}{2}|y - x|^2 + |y| &\Leftrightarrow y' - x + \frac{y'}{|y'|} = 0 \\ &\Leftrightarrow x = y' + \frac{y'}{|y'|} \qquad (A.2) \\ &\Leftrightarrow x = (y')(1 + \frac{1}{|y'|}). \qquad (A.3) \end{aligned}$$

De plus,

$$|y'| > 0 \;\Leftrightarrow\; |y'| + 1 > 1 \;\Leftrightarrow\; (1 + \frac{1}{|y'|})|y'| > 1 \;\Leftrightarrow\; |x| > 1.$$

Remarquons également que pour $|x| > 1$, on a $|y'-x| = 1$ d'après (A.2) et $(y')(1+\frac{1}{|y'|}) = x$ par (A.3). Alors y' et x sont colinéaires avec $|y' - x| = 1$, ce qui donne

$$y' = x \pm \frac{x}{|x|}.$$

En remplaçant y' par son expression dans (A.1) on obtient que le minimum est atteint pour
$y = x - \frac{x}{|x|}$. On obtient donc

$$y' = \begin{cases} x - \frac{x}{|x|} & \text{for } |x| > 1 \\ 0 & \text{sinon.} \end{cases}$$

et on peut écrire

$$\text{prox}_F(x) = \underset{y}{\text{argmin}} |x - y|^2 + |y| = \max\left(0, 1 - \frac{1}{|x|}\right) x$$

ou pour $\gamma > 0$

$$\text{prox}_{\gamma F}(x) = \underset{y}{\text{argmin}} |x - y|^2 + \gamma|y| = \max\left(0, 1 - \frac{\gamma}{|x|}\right) x.$$

Grâce à l'identité de Moreau on obtient

$$\begin{aligned} \text{prox}_{\gamma F^*}(x) &= x - \gamma \,\text{prox}_{F/\gamma}(x/\gamma) \\ &= x - \gamma \max\left(0, 1 - \frac{1}{\gamma|x/\gamma|}\right)(x/\gamma) \\ &= x - \max\left(0, 1 - \frac{1}{|x|}\right) x \\ &= x + \min\left(0, \frac{1}{|x|} - 1\right) x \\ &= \min\left(x, \frac{x}{|x|}\right) \end{aligned}$$

A.2 Opérateur proximal de $\mathcal{J}$

La fonctionnelle $\mathcal{J}$ est définie par

$$\mathcal{J}(\rho, m) = \int_0^1 \int_\Omega J(\rho(t,x), m(t,x)) dx dt,$$

où

$$\forall (\rho, m) \in \mathbb{R} \times \mathbb{R}^n,\ J(\rho, m) = \begin{cases} \frac{|m|^2}{2\rho}, & \text{si } \rho > 0, \\ 0, & \text{si } (\rho, m) = (0,0), \\ +\infty, & \text{sinon .} \end{cases}$$

La transformée de Legendre de la fonction, J^*, est définie par

$$J^*(a,b) = \max_{(\rho,m)\in\mathbb{R}\times\mathbb{R}^n} \langle a,m\rangle + \langle b,\rho\rangle - J(\rho,m),$$

Pour $\rho < 0$, on obtient $J = +\infty$ et $J^* = -\infty$ et pour $\rho = 0$, $J^* = 0$. Dans le cas $\rho > 0$,

$$J^*(a,b) = \max_{(\rho,m)\in\mathbb{R}\times\mathbb{R}^n} \langle a,m\rangle + \langle b,\rho\rangle - \frac{m^2}{2\rho}.$$

On obtient donc comme conditions d'optimalité :

$$a = \frac{m}{\rho} \text{ et } b = -\frac{m^2}{2\rho^2}.$$

Si on

$$\begin{aligned} J^*(a,b) &= \max_{(\rho,m)\in\mathbb{R}\times\mathbb{R}^n} \rho\left(\langle a,\frac{m}{\rho}\rangle + b - \frac{m^2}{2\rho^2}\right) \\ &= \max_{(\rho,m)\in\mathbb{R}\times\mathbb{R}^n} \rho(\langle a,a\rangle + 2b), \end{aligned}$$

ce qui donne 0 si $||a||^2 + 2b \leq 0$ et $+\infty$ sinon.

Ainsi, la transformée de Legendre de la fonctionnelle $\mathcal{J}$ est la fonction indicatrice d'un ensemble convexe, $\mathcal{J}^* = i_{P_{\mathcal{J}}}$ où

$$\begin{cases} P_{\mathcal{J}} = \{(z_1,z_2);\ \forall k \in G^c,\ (z_1,z_2)_k \in P_J\} \\ P_J = \{(t,x) \in \mathbb{R}\times\mathbb{R}^n,\ t + \frac{|x|^2}{2} \leq 0\}, \end{cases}$$

Cela implique que $\text{prox}_{\gamma J^*}$ est la projection sur le paraboloide P_J, que l'on va dénoter $\mathcal{P}_{P_J}$.

Annexe B

Opérateurs discrets de dérivée partielle adjoints

Nous allons montrer que les opérateurs discrets de dérivée partielle par rapport à la première composante, ∂_t et ∂_t^* sont bien adjoints l'un de l'autre. La démonstration est similaire pour tous les autres opérateurs définis dans la section 5.1.1. L'opérateur discret de dérivée partielle par rapport à la première composante s'écrit $\partial_t : G^{c1} \to G_t^{s1}$, pour $u \in G^{c1}$:

$$(\partial_t u)_{i+1/2,j} = u_{i+1,j} - u_{i,j}, \quad 1 \leq i \leq M-1,\ 1 \leq j \leq N.$$

L'opérateur de dérivée partielle adjoint pour $w = (w^1, w^2) \in G^{s1}$ est défini par

$$(\partial_t^* w^1)_{i,j} = \begin{cases} -w^1_{1+1/2,j} & \text{si } i = 1 \\ w^1_{i-1/2,j} - w^1_{i+1/2,j} & \text{si } 2 \leq i \leq M-1 \\ w^1_{M-1/2,j} & \text{si } i = M. \end{cases}$$

Ces opérateurs vérifient pour $w \in G_t^{s1}$

$$
\begin{aligned}
&\langle w, \partial_t(u)\rangle - \langle \partial_t^*(w), u\rangle \\
&= \sum_{i=1}^{M-1}\sum_{j=1}^{N} \big(w_{i+1/2,j}, u_{i+1,j} - u_{i,j}\big) \\
&- \left(-\sum_{j=1}^{N}(w_{1+1/2,j}, u_{1,j}) + \sum_{i=2}^{M-1}\sum_{j=1}^{N}(w_{i-1/2,j} - w_{i+1/2,j}, u_{i,j}) + \sum_{j=1}^{N}(w_{M-1/2,j}, u_{M,j}) \right) \\
&= \sum_{i=1}^{M-1}\sum_{j=1}^{N} \big(w_{i+1/2,j}, u_{i+1,j}\big) - \sum_{i=1}^{M-1}\sum_{j=1}^{N} \big(w_{i+1/2,j}, u_{i,j}\big) \\
&- \left(-\sum_{j=1}^{N}(w_{1+1/2,j}, u_{1,j}) + \sum_{i=2}^{M-1}\sum_{j=1}^{N}(w_{i-1/2,j} - w_{i+1/2,j}, u_{i,j}) + \sum_{j=1}^{N}(w_{M-1/2,j}, u_{M,j}) \right)
\end{aligned}
$$

En faisant le changement de variable $k = i + 1$ dans la première somme on obtient

$$
\begin{aligned}
&\langle w, \partial_t(u)\rangle - \langle \partial_t^*(w), u\rangle \\
&= \sum_{k=2}^{M}\sum_{j=1}^{N} \big(w_{k-1/2,j}, u_{k,j}\big) - \sum_{i=1}^{M-1}\sum_{j=1}^{N} \big(w_{i+1/2,j}, u_{i,j}\big) \\
&- \left(-\sum_{j=1}^{N}(w_{1+1/2,j}, u_{1,j}) + \sum_{i=2}^{M-1}\sum_{j=1}^{N}(w_{i-1/2,j} - w_{i+1/2,j}, u_{i,j}) + \sum_{j=1}^{N}(w_{M-1/2,j}, u_{M,j}) \right) \\
&= \sum_{j=1}^{N} \big(w_{M-1/2,j}, u_{M,j}\big) - \sum_{j=1}^{N} \big(w_{1+1/2,j}, u_{1,j}\big) \\
&- \left(-\sum_{j=1}^{N}(w_{1+1/2,j}, u_{1,j}) + \sum_{j=1}^{N}(w_{M-1/2,j}, u_{M,j}) \right) \\
&= 0
\end{aligned}
$$

Les opérateurs ∂_t et ∂_t^* sont donc bien adjoints.

Bibliographie

[1] L. Ambrosio, L.A. Caffarelli, M.G. Crandall et N. Evans, L.C.and Fusco : Calculus of Variations and Nonlinear Partial Differential Equations. Springer, 2008.

[2] L. Ambrosio, N. Fusco et D. Pallara : Functions of bounded variation and free discontinuity problems, volume 254. Clarendon Press Oxford, 2000.

[3] S. Angenent, S. Haker et A. Tannenbaum : Minimizing flows for the monge–kantorovich problem. SIAM journal on mathematical analysis, 35(1):61–97, 2003.

[4] P. Appell : Mémoire sur les déblais et les remblais des systèmes continus ou discontinus : présenté à l'Académie des Sciences pour le concours du prix Bordin pour 1884. Académie des Sciences de l'Institut National de France, 1887.

[5] G. Battle, P. Federbush et al. : Divergence-free vector wavelets. Michigan Math. J, 40(1):181–195, 1993.

[6] H. H. Bauschke et P. L. Combettes : Convex analysis and monotone operator theory in Hilbert spaces. Springer, 2011.

[7] J-D. Benamou et Y. Brenier : A computational fluid mechanics solution to the monge-kantorovich mass transfer problem. Numerische Mathematik, 84(3):375–393, 2000.

[8] J.-D. Benamou, B. Froese et A. Oberman : Numerical solution of the optimal transportation problem using the monge–ampere equation. Journal of Computational Physics, 260:107–126, 2014.

[9] N. Bonneel, J. Rabin, G. Peyré et H. Pfister : Sliced and radon wasserstein barycenters of measures. Journal of Mathematical Imaging and Vision, 51(1):22–45, 2015.

[10] A. Bouharguane, A. Iollo et L. Weynans : Numerical solution of the monge–kantorovich problem by density lift-up continuation. ESAIM : Mathematical Modelling and Numerical Analysis, 49(6):1577–1592, 2015.

[11] A. Bouharguane, E. Maitre, E. Oudet et N. Papadakis : Multiphysics optimal transportation and image analysis. hal preprint : hal-00740671, 2012.

[12] Y. Brenier : Polar factorization and monotone rearrangement of vector-valued functions. Communications on pure and applied mathematics, 44(4):375–417, 1991.

[13] P. Cardaliaguet, G. Carlier et B. Nazaret : Geodesics for a class of distances in the space of probability measures. Calculus of Variations and Partial Differential Equations, 48(3-4):395–420, 2013.

[14] J. A. Carrillo et J. S. Moll : Numerical simulation of diffusive and aggregation phenomena in nonlinear continuity equations by evolving diffeomorphismss. SIAM J. Sci. Comput., 31:4305–4329, 2009.

[15] A. Chambolle : An algorithm for total variation minimization and applications. Journal of Mathematical imaging and vision, 20(1-2):89–97, 2004.

[16] A. Chambolle et T. Pock : A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision, 40(1):120–145, 2011.

[17] R. Chartrand, B. Wohlberg, K. Vixie et E. Bollt : A gradient descent solution to the monge-kantorovich problem. Applied Mathematical Sciences, 3(22):1071–1080, 2009.

[18] P-A. Chiappori, R. J. McCann et L. P. Nesheim : Hedonic price equilibria, stable matching, and optimal transport : equivalence, topology, and uniqueness. Economic Theory, 42(2):317–354, 2010.

[19] L. Chizat, B. Schmitzer, G. Peyré et F.-X. Vialard : An interpolating distance between optimal transport and fischer-rao. arXiv preprint arXiv :1506.06430, 2015.

[20] A. Cohen, I. Daubechies et J.-C. Feauveau : Biorthogonal bases of compactly supported wavelets. Communications on pure and applied mathematics, 45(5):485–560, 1992.

[21] P. L. Combettes, L. Condat, J. C. Pesquet et B. C. Vu : A forward-backward view of some primal-dual optimization methods in image recovery. In Image Processing (ICIP), 2014 21st IEEE International Conference on, pages 4141–4145. IEEE, 2014.

[22] P. L. Combettes et J-C. Pesquet : Proximal splitting methods in signal processing. In Fixed-point algorithms for inverse problems in science and engineering, pages 185–212. Springer, 2011.

[23] P. L. Combettes et V. R. Wajs : Signal recovery by proximal forward-backward splitting. Multiscale Modeling & Simulation, 4(4):1168–1200, 2005.

[24] M. Cullen et W. Gangbo : A variational approach for the 2-dimensional semi-geostrophic shallow water equations. Archive for rational mechanics and analysis, 156(3):241–273, 2001.

[25] I. Daubechies : Ten lectures on wavelets, volume 61. SIAM, 1992.

[26] E. Deriaz : Ondelettes pour la Simulation des Ecoulements Fluides Incompressibles en Turbulence. Thèse de doctorat, Institut National Polytechnique de Grenoble, 2006.

[27] E. Deriaz et V. Perrier : Divergence-free and curl-free wavelets in two dimensions and three dimensions : application to turbulent flows. Journal of Turbulence, (7), 2006.

[28] B. Desjardins : A few remarks on ordinary differential equations. Communications in Partial Differential Equations, 21(11-12), 1996.

[29] E. Esser, X. Zhang et T. Chan : A general framework for a class of first order primal-dual algorithms for tv minimization. UCLA CAM Report, pages 09–67, 2009.

[30] L. C. Evans et W. Gangbo : Differential equations methods for the Monge-Kantorovich mass transfer problem, volume 653. American Mathematical Soc., 1999.

[31] L. C. Evans et R. F. Gariepy : Measure theory and fine properties of functions. CRC Press, Boca Raton, 1992.

[32] H. Federer : Geometric measure theory. Grundlehren der Mathematischen Wissenschaften, 153, 1969.

[33] N. Feyeux : Optimal transport for data assimilation from images. Thèse de doctorat, Grenoble-Alpes University, 2016.

[34] A. Figalli et N. Gigli : A new transportation distance between non-negative measures, with applications to gradients flows with Dirichlet boundary conditions. Journal de mathématiques pures et appliquées, 94(2):107–130, 2010.

[35] J. H. Fitschen, F. Laus et G. Steidl : Dynamic optimal transport with mixed boundary condition for color image processing. arXiv preprint arXiv :1501.04840, 2015.

[36] J. H. Fitschen, F. Laus et G. Steidl : Transport between rgb images motivated by dynamic optimal transport. arXiv preprint arXiv :1509.06142, 2015.

[37] W. Gangbo et R. J. McCann : The geometry of optimal transportation. Acta Mathematica, 177(2):113–161, 1996.

[38] V. Girault et P.-A. Raviart : Finite element methods for Navier-Stokes equations : theory and algorithms. Springer-Verlag, 1986.

[39] E Giusti : Minimal surfaces and functions of bounded variation. Numéro 80. Springer Science & Business Media, 1984.

[40] A. Gramfort, G. Peyré et M. Cuturi : Fast optimal transport averaging of neuroimaging data. arXiv preprint arXiv :1503.08596, 2015.

[41] K. Guittet : Contributions à la résolution numérique de problèmes de transport optimal de masse. Thèse de doctorat, Paris 6, 2003.

[42] K. Guittet : On the time-continuous mass transport problem and its approximation by augmented lagrangian techniques. SIAM Journal on Numerical Analysis, 41(1):382–399, 2003.

[43] E. Haber, T. Rehman et A. Tannenbaum : An efficient numerical method for the solution of the l_2 optimal mass transfer problem. SIAM Journal on Scientific Computing, 32(1):197–211, 2010.

[44] E. Haber, T. Rehman et A. Tannenbaum : An efficient numerical method for the solution of the l_2 optimal mass transfer problem. SIAM Journal on Scientific Computing, 32(1):197–211, 2010.

[45] B. He et X. Yuan : Convergence analysis of primal-dual algorithms for total variation image restoration. Rapport technique, Citeseer, 2010.

[46] R. Hug : Transport optimal sous contrainte, algorithmes numériques et applications à l'assimilation. Thèse de doctorat, Grenoble-Alpes University, 2016.

[47] Hug, R., Maitre, E. et Papadakis, N. : Multi-physics optimal transportation and image interpolation. ESAIM : M2AN, 49(6):1671–1692, 2015.

[48] R. Jordan, D. Kinderlehrer et F. Otto : The variational formulation of the fokker–planck equation. SIAM journal on mathematical analysis, 29(1):1–17, 1998.

[49] A Jouini et PG Lemarié-Rieusset : Analyse multi-résolution bi-orthogonale sur l'intervalle and applications. In Annales de l'IHP Analyse non linéaire, volume 10, pages 453–476, 1993.

[50] S. Kadri Harouna : Ondelettes pour la prise en compte de conditions aux limites en turbulence incompressible. Thèse de doctorat, Université de Grenoble, 2010.

[51] S. Kadri Harouna et V. Perrier : Effective construction of divergence-free wavelets on the square. Journal of Computational and Applied Mathematics, 240:74–86, 2013.

[52] S. Kadri Harouna et V. Perrier : Divergence-free wavelet projection method for incompressible viscous flow on the square. Multiscale Modeling & Simulation, 13(1):399–422, 2015.

[53] J. P. Kahane et P. G. Lemarié-Rieusset : Séries de Fourier et ondelettes. Cassini, 1998.

[54] L. V. Kantorovich : On the transfer of masses. Dokl. Akad. Nauk. SSSR, 37(7-8):227–229, 1942.

[55] L. V. Kantorovich : On the translocation of masses. Dokl. Akad. Nauk SSSR, 37:199–201, 1942.

[56] L. V. Kantorovich : On a problem of monge. English translation in Journal of Mathematical Sciences, 133(4):1383–1383, 2006.

[57] S. Kondratyev, L. Monsaingeon et D. Vorotnikov : A new optimal transport distance on the space of finite radon measures. arXiv preprint arXiv :1505.07746, 2015.

[58] P. G. Lemarie-Rieusset : Analyses multi-résolutions non orthogonales, commutation entre projecteurs et dérivation et ondelettes vecteurs à divergence nulle. Revista Matemática Iberoamericana, 8(2):221–238, 1992.

[59] D. Lombardi et E. Maitre : Eulerian models and algorithms for unbalanced optimal transport. ESAIM : Mathematical Modelling and Numerical Analysis, 49(6): 1717–1744, 2015.

[60] S. Mallat : A wavelet tour of signal processing. Academic press, 1999.

[61] R. Masson : Biorthogonal spline wavelets on the interval for the resolution of boundary problems. Mathematical Models and Methods in Applied Sciences, 6(06): 749–791, 1996.

[62] R. J. McCann : Existence and uniqueness of monotone measure-preserving maps. Duke Mathematical Journal, 80(2):309–324, 1995.

[63] R. J McCann : A convexity principle for interacting gases. Advances in mathematics, 128(1):153–179, 1997.

[64] Y. Meyer : Ondelettes and opérateurs, vol. i. Hermann, Paris, 1990.

[65] P. Monasse et V. Perrier : Orthonormal wavelet bases adapted for partial differential equations with boundary conditions. SIAM journal on mathematical analysis, 29(4):1040–1065, 1998.

[66] G. Monge : Mémoire sur la théorie des déblais et des remblais. page 666–704, 1781.

[67] J-J. Moreau : Proximité et dualité dans un espace hilbertien. Bulletin de la Société mathématique de France, 93:273–299, 1965.

[68] K. Ni, X. Bresson, T. Chan et S. Esedoglu : Local histogram based segmentation using the wasserstein distance. International Journal of Computer Vision, 84(1):97–111, 2009.

[69] N. Papadakis, G. Peyré et E. Oudet : Optimal transport with proximal splitting. SIAM Journal on Imaging Sciences, 7(1):212–238, 2014.

[70] G. Peyré, J. Fadili et J. Rabin : Wasserstein active contours. In Image Processing (ICIP), 2012 19th IEEE International Conference on, pages 2541–2544. IEEE, 2012.

[71] J. Rabin, S. Ferradans et N. Papadakis : Adaptive color transfer with relaxed optimal transport. In Image Processing (ICIP), 2014 21st IEEE International Conference on, pages 4852–4856. IEEE, 2014.

[72] J. Rabin, G. Peyré, J. Delon et M. Bernot : Wasserstein barycenter and its application to texture mixing. pages 435–446, 2012.

[73] S.T. Rachev et L. Rüschendorf : Mass Transportation Problems : Volume I : Theory, volume 1. Springer, 1998.

[74] R. T. Rockafellar : Characterization of the subdifferentials of convex functions. Pacific Journal of Mathematics, 17(3):497–510, 1966.

[75] R. T. Rockafellar : Monotone operators and the proximal point algorithm. SIAM journal on control and optimization, 14(5):877–898, 1976.

[76] R T. Rockafellar : Convex analysis, volume 28. Princeton university press, 1997.

[77] Y. Rubner, C. Tomasi et L. J. Guibas : A metric for distributions with applications to image databases. In Computer Vision, 1998. Sixth International Conference on, pages 59–66. IEEE, 1998.

[78] Y. Rubner, C. Tomasi et L. J. Guibas : The earth mover's distance as a metric for image retrieval. International journal of computer vision, 40(2):99–121, 2000.

[79] L. I. Rudin, S. Osher et E. Fatemi : Nonlinear total variation based noise removal algorithms. Physica D : Nonlinear Phenomena, 60(1):259–268, 1992.

[80] L.-P. Saumier, M. Agueh et B. Khouider : An efficient numerical algorithm for the l2 optimal transport problem with applications to image processing. arXiv preprint arXiv :1009.6039, 2010.

[81] C. S. Smith et M. Knott : Note on the optimal transportation of distributions. Journal of Optimization Theory and Applications, 52(2):323–329, 1987.

[82] T. Valkonen : A primal-dual hybrid gradient method for non-linear operators with applications to MRI. Inverse Problems, 30(5): :055012, 2014.

[83] C. Villani : Topics in optimal transportation. Numéro 58. American Mathematical Soc., 2003.