

# Conception d'algorithmes performants pour le contrôle, le transport optimal et l'accélération de la résolution d'EDP.

THÈSE D'HABILITATION À DIRIGER DES RECHERCHES

présentée et soutenue publiquement le 18 novembre 2010

pour l'obtention de l'

**Habilitation à diriger des recherches – Université Paris-Dauphine**  
(spécialité mathématiques)

par

Julien Salomon

## Composition du jury

*Rapporteurs :* Yann Brenier  
Martin Gander  
Enrique Zuazua

*Examineurs :* Antonin Chambolle  
Antoine Henrot  
Yvon Maday  
Bertrand Maury  
Gabriel Turinici

Mis en page avec la classe thloria.

## Remerciements

La lecture de ce mémoire pourrait en masquer les aspects humains. Ce dernier retrace surtout les cheminements et les résultats scientifiques auxquels j'ai pris part ces huit dernières années. Les travaux qui y sont décrits n'auraient pourtant pas vu le jour sans tous les gens dont je vais vous parler ici.

Je tiens tout d'abord à exprimer ma profonde gratitude à Yann Brenier, Martin Gander et Enrique Zuazua pour avoir eu la gentillesse d'accepter de rapporter ce mémoire.

Je remercie aussi sincèrement Antonin Chambolle, Antoine Henrot et Bertrand Maury d'avoir participé au jury de me soutenir, même si leurs domaines de recherche peuvent être un peu distants de certains de mes travaux.

Je salue également Yvon Maday et Gabriel Turinici, avec qui il fût et il est toujours très agréable de travailler. Leur confiance, leur enthousiasme et leur sympathie sont pour moi fondamentales.

Pour leur amitié et pour être là où ils sont dans mon quotidien, je rappelle à mon cher collègue de bureau Guillaume Legendre, à ma chère collègue de transport Julie Delon, à ma chère collègue de Schrödinger Karine Beauchard et à mon deuxième et non moins cher collègue de transport Andreï Sobolevskiï, qu'ils sont indispensables à mon équilibre professionnel et probablement personnel.

C'est connu, les mathématiques sont une source intarissable de tracasseries, de blocages et autres prises de tête. J'aurais longtemps tourné en rond si certaines personnes providentielles n'avaient pas été sur mon chemin aux moments opportuns. Même si les conseils qu'ils m'ont donnés étaient pour eux élémentaires, ils m'ont permis d'avancer et leurs savoir-faire m'ont largement influencé depuis. Je remercie ainsi pour leurs interventions essentielles Jérôme Bolte, Otared Kavian, Frédéric Lagoutière, Édouard Oudet et Gérard Lebourg.

Je dois également beaucoup à Guillaume Carlier. Sa gentillesse, son enthousiasme, ses préoccupations font qu'il est pour moi toujours important, intéressant et agréable de discuter avec lui.

Il va de soi que mon travail est aussi celui de mes collaborateurs. Pour ces efforts partagés, je remercie Lucie Baudouin, Mohamed Belhadj, Alfio Borzi, Claude Dion, Bernard Haasdonk, Ivan Maximov, Barbara Wohlmuth, et bien sûr Alexander Weiss.

C'est avec grand plaisir que j'ai co-encadré la thèse d'Aimé Lachapelle, que je co-encadre celle de Kamel Riahi et que j'ai encadré les stages de M2 de Mehdi Benhamouche et de Philippe Laurent. Je les remercie de leur confiance.

J'ai également une pensée pour les membres de mon laboratoire, le CEREMADE et pour Isabelle Bellier, Tatiana Blondel, Patricia Dessans, César Faivre, Jean-Paul Fourmas, Irène Dos Santos, Christine Vermont pour leur soutien pas seulement technique.

Enfin, merci à mes très chers copains Jérôme, Julien, Vincent, Stéphane, Nathalie, Marlène, Hadia, Régis, Nicolas, à mes parents, à Maud et Antoine, et à Dagmar.



# Table des matières

Introduction générale	1
-----------------------	---

Partie I Optimisation	13
-----------------------	----

<b>Chapitre 1 Algorithmes monotones pour le contrôle optimal</b>
--

1.1 Conception et analyse des algorithmes monotones . . . . .	17
1.1.1 État adjoint et factorisation . . . . .	17
1.1.2 Lien avec la poursuite de trajectoire et stratégies d'optimisation . . .	21
1.1.3 Discrétisation et implémentation . . . . .	23
1.1.4 Convergence de l'algorithme . . . . .	25
1.2 Applications . . . . .	27
1.2.1 Contrôle de l'alignement et de l'orientation de molécules . . . . .	28
1.2.2 Résonance magnétique nucléaire . . . . .	28
1.2.3 Construction de champs sélectifs pour l'identification . . . . .	29

<b>Chapitre 2 Quelques méthodes numériques pour le transport optimal</b>
--

2.1 Algorithmes pour la dimension 1 . . . . .	34
2.1.1 Coût de transport convexe sur le cercle . . . . .	34
2.1.2 Indicateurs d'appariement locaux pour les coût concaves . . . . .	36
2.2 Algorithme pour les dimensions supérieures, application aux jeux à champ moyen	40
2.2.1 Algorithme monotone adapté . . . . .	40
2.2.2 Application à la théorie des jeux à champ moyen . . . . .	43

Partie II Analyse numérique	47
-----------------------------	----

<b>Chapitre 1 Parallélisation en temps de méthodes de contrôle</b>
--

1.1 Algorithme pour les équations hyperboliques . . . . .	50
1.1.1 Le problème . . . . .	50

1.1.2	Cadre adapté à la parallélisation . . . . .	51
1.1.3	L'algorithme . . . . .	52
1.1.4	Convergence . . . . .	52
1.1.5	Résultats numériques . . . . .	53
1.2	Algorithme pour les équations paraboliques . . . . .	53
1.2.1	Le problème . . . . .	54
1.2.2	Parallélisation . . . . .	55
1.2.3	L'algorithme . . . . .	56
1.2.4	Convergence . . . . .	56
1.2.5	Résultats numériques . . . . .	57

<b>Chapitre 2 Analyse numérique et formulation co-rotationnelle</b>
---

2.1	Schéma conservatif pour la formulation co-rotationnelle . . . . .	60
2.1.1	Le modèle . . . . .	60
2.1.2	Décomposition co-rotationnelle . . . . .	61
2.1.3	Linéarisation . . . . .	62
2.1.4	Discrétisation en temps et schémas de résolution . . . . .	62
2.1.5	Discrétisation en espace et caractère bien posé de l'algorithme . . . . .	64
2.1.6	Cas de la dimension 3 . . . . .	65
2.1.7	Tests numériques . . . . .	65
2.2	Prise en compte du contact et de la friction . . . . .	66
2.2.1	Modèle . . . . .	66
2.2.2	Décomposition co-rotationnelle . . . . .	67
2.2.3	Discrétisation en temps et schémas de résolution . . . . .	69
2.2.4	Résolution des non-linéarités . . . . .	70
2.2.5	Quelques tests . . . . .	70

<b>Chapitre 3 Algorithmes d'accélération par pré-calcul</b>
---

3.1	Méthode du <i>Toolkit</i> . . . . .	74
3.1.1	Présentation de la méthode . . . . .	74
3.1.2	Analyse de la méthode . . . . .	75
3.1.3	Variantes et augmentation de l'ordre de convergence . . . . .	75
3.1.4	Tests numériques . . . . .	77
3.1.5	Couplage avec les schémas monotones . . . . .	78
3.2	Méthode de base réduite pour les inégalités variationnelles . . . . .	79
3.2.1	Problème considéré . . . . .	79
3.2.2	Trois formulations adaptées aux bases réduites . . . . .	80

---

3.2.3	Pré-calcul et résolution en ligne . . . . .	81
3.2.4	Méthodes numériques utilisées . . . . .	82
3.2.5	Test numériques . . . . .	82
<b>Partie III Perspectives</b>		<b>87</b>
<b>Bibliographie</b>		<b>91</b>



# Introduction générale

*Cette introduction a pour objectif de présenter mes contributions en langage courant, les différents cadres dans lesquels elles ont été construites ainsi que les cheminements qui ont permis de les obtenir. Les énoncés rigoureux des résultats ainsi que les remarques bibliographiques seront détaillés dans le corps du manuscrit.*

L'objectif de ce mémoire est de constituer un guide de lecture des résultats que j'ai obtenus au cours de ces huit dernières années : trois années de thèse à l'université Paris VI, une année de stage post-doctoral à l'université de Stuttgart et quatre années de recherches en tant que Maître de conférences à l'université Paris-Dauphine. Ces résultats se regroupent autour de deux thématiques principales, l'optimisation et l'analyse numérique.

Le terme *optimisation* recouvre ici plus précisément méthodes numériques d'optimisation, l'essentiel des résultats de ce mémoire consistant en la conception, l'étude théorique et l'application d'algorithmes dédiés à la résolution de problèmes d'optimisation, l'enjeu étant ici de calculer numériquement les optima de diverses fonctionnelles de coût.

Le terme *analyse numérique*, recouvre ici plus précisément la conception et l'analyse de méthodes numériques de simulation et de résolution approchées d'équations ou de systèmes d'équations aux dérivées partielles (EDP). L'enjeu principal est d'obtenir de méthodes fournissant des approximations raisonnables en des temps courts.

Avant d'entrer dans une description plus détaillée, signalons aussi que ce travail aurait également pu être présenté selon une autre structure, construite à partir des applications des méthodes. Celles-ci s'articulent en effet autour de trois domaines : le contrôle quantique, le transport optimal et la mécanique des milieux continus. Une telle description fait apparaître les cadres de production des algorithmes présentés dans ce mémoire. Le contrôle quantique fut pour moi le point d'entrée dans la recherche. Ce sujet m'a été proposé au printemps 2002 par Yvon Maday et Gabriel Turinici lors de mon stage de DEA, et a occupé l'essentiel de mes trois années de thèse. Le travail sur cette thématique trouvait sa motivation dans une collaboration avec l'équipe d'Herschel Rabitz à l'université de Princeton. Les travaux concernant la mécanique des milieux continus ont été initiés à l'automne 2005 lors de mon post-doc à l'université de Stuttgart avec Barbara Wohlmuth. Enfin, j'ai commencé mon travail sur les algorithmes pour le transport optimal au printemps 2007 à l'occasion du projet A.N.R. *Otarie* (Optimal transport : Theory and Applications to cosmological Reconstruction and Image processing), coordonné par Andreï Sobolevskiĭ.

Les travaux décrits dans cette partie ont tous donné lieu à des algorithmes permettant d'optimiser des fonctionnelles. Les méthodes se classent selon deux catégories : une première partie d'entre elles est liée à des formulations continues, éventuellement discrétisées dans un second temps (algorithmes monotones, algorithmes pour le transport optimal en dimension supérieure à 1), une seconde partie est constituée d'algorithmes discrets, de type combinatoire (algorithmes pour le transport optimal en dimension 1).

### Algorithmes monotones pour le contrôle optimal

La première partie de mes travaux en optimisation concerne le contrôle optimal et fait l'objet du chapitre 1 de la partie I du mémoire. Ces résultats ont été principalement obtenus lorsque j'étais en thèse au laboratoire Jacques-Louis Lions, en collaboration avec Lucie Baudouin, Yvon Maday et Gabriel Turinici. Ils sont inspirés par une classe d'algorithmes lié au contrôle par laser de l'équation de Schrödinger. Ces algorithmes, appelés *schémas monotones* ou encore *algorithmes monotones* furent introduits dans les années 90 par différents groupes de chimistes. Au début de ma thèse, ce type de procédure soulevait trois problèmes de nature différente. Le premier était d'ordre pratique : l'implémentation d'un algorithme mettait en évidence une grande instabilité du schéma numérique résultant. Le deuxième concernait la compréhension des procédures elles-mêmes : la méthode utilisée pour obtenir la factorisation évoquée plus haut repose exclusivement sur une série de calculs qui ne permet pas d'interpréter l'algorithme intuitivement, ou de le rapprocher d'un autre algorithme connu. Le troisième problème était d'ordre théorique : aucune preuve de convergence n'existait pour cet algorithme. Une partie de mon travail sur les algorithmes monotones traite de ces trois questions, le reste concernant les adaptations et les applications de cette méthode.

Commençons par décrire plus précisément les fondements des algorithmes monotones. La méthode fut donc proposée par des chimistes et a pour but d'optimiser une fonctionnelle  $J$  associée à un certain objectif, par exemple un état cible  $\psi_{cible}$  à atteindre :

$$J(\varepsilon) = \|\psi(T) - \psi_{cible}\|_2^2 + \alpha \int_0^T \varepsilon(t)^2 dt,$$

Ici, le contrôle  $\varepsilon$  est le champ électrique produit par un laser, l'état  $\psi$  est la fonction d'onde à contrôler. Le terme  $\alpha$  permet la pénalisation  $L^2$  du contrôle, de sorte que sa norme reste acceptable physiquement. Un état initial  $\psi_0$  étant donné, le cadre est celui du contrôle bilinéaire, puisque le terme de contrôle multiplie l'état dans l'équation de Schrödinger :

$$i\partial_t \psi = [H_0 - \mu\varepsilon(t)]\psi.$$

Le terme  $H_0$  représente l'Hamiltonien du système considéré, et  $\mu$  son moment dipolaire. L'algorithme repose avant tout sur un calcul astucieux dans lequel l'introduction d'un état adjoint  $\chi$  conduit à une factorisation de la variation de la fonctionnelle entre deux contrôles arbitraires :

$$J(\varepsilon') - J(\varepsilon) = \alpha \int_0^T (\varepsilon'(t) - \varepsilon(t)) \cdot \Delta(\psi'(t), \chi(t)) dt,$$

---

où  $\Delta(\psi'(t), \chi(t))$  est une fonction explicite dépendant de l'état contrôlé  $\psi'(t)$  par l'intermédiaire de  $\varepsilon'$  et de l'adjoint  $\chi(t)$  associé à  $\varepsilon$ . Sous cette forme, il devient facile de concevoir un algorithme itératif minimisant de manière monotone la fonctionnelle  $J$ . Dans un article récent, présenté à la section 1.1.1, j'ai donné un cadre mathématique général dans lequel s'applique cette démarche, ainsi qu'une forme générale de méthode monotone. Ce travail, effectué en collaboration avec Gabriel Turinici est présenté à la section 1.1.1 de ce mémoire et correspond à la publication [K].

Comme je l'ai déjà mentionné, l'implémentation numérique des algorithmes monotones dans le cadre quantique donnait lieu dans de nombreux tests à des instabilités numériques (qui pouvaient cependant être retardées en diminuant le pas de temps). Dans un premier travail, je me suis intéressé à la discrétisation en temps du problème. J'ai en particulier explicité l'impact du choix de discrétisation sur les critères de monotonie de l'algorithme. Ce travail a permis de proposer deux classes de schémas – implicite et explicite – assurant la stabilité de la méthode. La démarche suivie consiste à reprendre la manipulation algébrique sous-jacente aux algorithmes au niveau discret. Sous cette forme, on obtient de manière exacte un équivalent de la factorisation de la fonctionnelle, et donc des critères de monotonie de l'algorithme. Ce travail, effectué en collaboration avec Gabriel Turinici et Yvon Maday est présenté à la section 1.1.3 de ce mémoire et correspond aux publications [Proc. b] et [B].

Dans un second temps, j'ai identifié la procédure à un algorithme de poursuite de trajectoire, *Reference trajectory tracking*, en horizon fini. Autrement dit, un algorithme monotone consiste à rechercher à chaque instant entre 0 et  $T$  un contrôle permettant de réduire l'écart entre l'état courant du système et une trajectoire de référence. Dans le cas considéré, la trajectoire de référence est mise à jour itérativement. Celle-ci se trouve être celle de l'état adjoint qui, dans le cas hyperbolique, suit la même dynamique que celle de l'état direct et atteint au temps  $T$  la cible considérée. Cette interprétation, qui fut fondamentale pour la mise en place de parallélisation présentée au chapitre 1 de la partie II, a permis de concevoir un algorithme monotone stochastique, de faible coût numérique. Ce travail, effectué en collaboration avec Gabriel Turinici est présenté à la section 1.1.2 de ce mémoire et correspond à la publication [C].

La dernière partie de mon travail est sans doute la plus importante. Elle concerne la preuve de convergence de la méthode dans le cas quantique. La démonstration repose sur une utilisation particulière de l'inégalité de Łojasiewicz. Ce résultat a en fait donné lieu à deux articles distincts : l'un consacré spécifiquement aux schémas numériques, où des conditions sur le pas de temps sont obtenues, l'autre, traitant le cas de l'algorithme continu, où un travail d'analyse fonctionnelle a été nécessaire pour étendre les résultats à la dimension infinie. Dans les deux cas, des taux de convergence ont été obtenus, et vérifiés numériquement dans le cas discret. Ce travail, réalisé en partie avec Lucie Baudouin, est présenté à la section 1.1.4 de ce mémoire et correspond aux publications [D] et [F].

Parallèlement à ces recherches, plusieurs applications ont été développées et testées sur des exemples fournis par des chimistes. La première, développée en collaboration avec Claude Dion et Gabriel Turinici, concerne le problème assez largement décrit dans la littérature du contrôle de l'alignement et de la rotation de molécules. Une spécificité de cette problématique est que le contrôle y intervient de manière complexe : par rapport au cadre standard du contrôle bilinéaire, ce n'est plus le contrôle, mais une fonction de celui-ci qui multiplie l'état. Dans ce contexte, une adaptation d'un algorithme monotone a permis d'obtenir un contrôle optimal révélant un mécanisme de contrôle par *Ladder climbing* : les différentes résonnances du système apparaissent au cours du contrôle, faisant monter le système vers des états de plus en plus orientés. Il est intéressant de voir que l'algorithme retrouve à cette occasion les fréquences naturelles du système. Ce

travail est présenté à la section 1.2.1 de ce mémoire et correspond aux publications [Proc. c] et [A].

Plus récemment et en collaboration avec Ivan Maximov et Gabriel Turinici, nous avons construit une version de l'algorithme adapté à un problème de résonance magnétique nucléaire. La difficulté a ici consisté à coupler l'algorithme avec une procédure de régularisation permettant de contrôler le spectre fréquentiel du laser. Ce travail est présenté à la section 1.2.2 de ce mémoire et correspond à la publication [N].

Dans une dernière application, un algorithme monotone a été utilisé pour construire des familles de champs lasers discriminants. Ces champs s'avèrent très efficaces lorsqu'ils sont utilisés dans des procédures d'identification de systèmes quantiques. Ce travail, réalisé en collaboration avec Yvon Maday, est présenté à la section 1.2.3 de ce mémoire et correspond à la publication [Proc. f].

### *Méthodes numériques pour le transport optimal*

Un second aspect de mon travail sur l'optimisation numérique est le transport optimal. L'ensemble des travaux correspondants sont présentés au chapitre 2 de la première partie du mémoire. Peu après être arrivé au CEREMADE, j'ai commencé à m'intéresser au transport optimal, et plus particulièrement aux algorithmes de résolution. Même s'ils relèvent majoritairement d'une toute autre classe de méthodes, les algorithmes que j'ai développés pour résoudre des problèmes de transport optimal trouvent initialement leur source dans le contrôle optimal. La piste m'a été suggérée par Günter Leugering, qui me signala que les algorithmes monotones pouvaient peut-être trouver une application en transport optimal, via la formulation du problème de Monge-Kantorovitch proposée par Brenier et Benamou dans [3]. Ce point de départ se poursuit sous la forme d'un projet A.N.R. que j'ai monté en collaboration avec Andreï Sobolevskiï.

Le premier algorithme de transport que j'ai proposé est basé sur l'extension de certaines idées issues des algorithmes présentés au chapitre 1 de la partie I. Ce travail a été réalisé en collaboration avec Guillaume Carlier. La formulation précise du problème et l'algorithme de résolution font l'objet de la section 2.2.1. Partant d'un problème de contrôle optimal, lié à une équation de transport avec diffusion de type Fokker-Planck, on a construit un schéma d'optimisation monotone qui s'est avéré numériquement très efficace. Les deux principales innovations de ce travail portent d'une part sur le couplage du schéma d'optimisation avec une méthode de résolution numérique de l'équation d'évolution du système, d'autre part sur la mise en place de la stratégie d'optimisation.

Le problème considéré est celui de la minimisation de la fonctionnelle définie par :

$$J(v) = \frac{1}{2} \int_0^T \int_{\Omega} \rho(x, t) v^2(x, t) dx dt + \int_{\Omega} V(x) \rho(x, T) dx,$$

où  $\Omega \subset \mathbb{R}^N$  est borné,  $T$  le temps de contrôle,  $\rho$  est la variable d'état, contrôlée par le champ de vitesse  $v$  suivant l'équation :

$$\begin{aligned} \partial_t \rho(x, t) - \varepsilon \Delta \rho(x, t) + \operatorname{div}(v(x, t) \rho(x, t)) &= 0, \\ \rho(x, 0) &= \rho_0(x), \end{aligned}$$

où  $\rho_0$  est l'état initial et  $\varepsilon$  est un paramètre de diffusion. La fonction  $V$  représente un potentiel, associé à  $\rho$ . Ce problème peut être vu comme une version simplifiée d'un problème de transport optimal, puisque l'état final n'est pas prescrit. En revanche, cette formulation permet la prise en compte de phénomènes de diffusion, ce qui n'est généralement pas le cas dans les problèmes de transport optimal. Le modèle sous-jacent est celui d'un mouvement de foule, se déplaçant à partir d'un état initial fixé vers un état minimisant conjointement son énergie cinétique moyenne et le potentiel  $\int_{\Omega} V(x)\rho(x,T)dx$ , au temps  $T$ .

La fonctionnelle étant concave par rapport à l'état (puisque linéaire par rapport à  $\rho$ ) et l'équation d'évolution étant linéaire, les critères d'application des algorithmes monotones sont satisfaits. Quelques précautions sont cependant nécessaires pour la mise en œuvre pratique d'une telle procédure : celle-ci va en effet donner lieu à un couplage entre une méthode numérique associée à l'équation d'évolution et une stratégie d'optimisation. Dans ce cadre la linéarité de la fonctionnelle peut engendrer des instabilités numériques : une erreur numérique peut rendre l'état négatif en certains points et la routine de minimisation aura alors la possibilité d'accroître cette erreur en faisant tendre la valeur fonctionnelle discrète vers  $-\infty$ . Ceci n'a évidemment aucun sens puisqu'au niveau continu, l'équation d'évolution préserve le caractère positif de l'état. Le recours à une technique de décentrage garantissant la positivité a permis de résoudre cette difficulté et j'ai ensuite conçu un algorithme d'optimisation adapté à ce cadre. La méthode ainsi construite s'avère particulièrement efficace, puisque dans de nombreux cas seule une trentaine d'itérations est nécessaires pour obtenir un optimiseur. Ce travail est présenté à la section 2.2.1 de ce mémoire et correspond à la publication [Proc. e].

La théorie des jeux à champ moyen développée par Jean-Michel Lasry et Pierre-Louis Lions donne lieu à des problèmes d'optimisation auxquels la méthode générale qui vient d'être décrite peut aussi être appliquée. En collaboration avec Gabriel Turinici et Aimé Lachapelle, j'ai ainsi considéré le problème de jeux à champ moyen suivant :

$$J(v) = \int_Q \left( p(t)(1 - \beta z) + \frac{c_0 \cdot z}{c_1 + c_2 \rho(t, z)} + \frac{v^2(t)}{2} \right) \rho(t, z) dz,$$

où  $\beta, c_0, c_1, c_2$  sont des constantes positives, et où  $p(t)$  est une fonction positive. Cette fonctionnelle rend compte d'une situation simple, où des populations ont à choisir pour leurs habitations entre un niveau d'isolation  $z$  élevé, ou au contraire un rapprochement les unes des autres. Le premier terme rend ainsi compte du coût de chauffage au cours du temps, le second modélise le bénéfice apporté par une stratégie d'agglomération et le coût d'entretien. Le dernier terme représente le coût de changement d'état.

Ce problème rentre dans le cadre d'application de la stratégie d'optimisation par algorithme monotone : la fonctionnelle est d'une part concave par rapport à la variable d'état du système et d'autre part l'équation d'évolution est linéaire.

D'un point de vue pratique, les expériences numériques ont mis en évidence l'existence de plusieurs équilibres pour un jeu de paramètres donné. Ce travail est présenté à la section 2.2.2 de ce mémoire et correspond à la publication [Proc. e].

Une seconde série de travaux concerne des problèmes de transport optimal en dimension 1. Il est connu depuis longtemps que dans le cas d'un transport sur la droite réelle et lorsque le coût de déplacement est une fonction convexe de la distance, le plan de transport optimal entre deux mesures est donné de manière explicite par le ré-arrangement monotone. En collaboration avec Andreï Sobolevskiï et Julie Delon, j'ai considéré deux variantes de ce problème pour lesquelles il n'existait à ma connaissance pas de tel résultat : le problème du transport optimal sur le cercle

en coût convexe d'une part, et le problème du transport optimal (sur le cercle et sur la droite) en coût concave d'autre part.

Le premier travail sur le cercle est issu d'un problème proposé par Julie Delon et lié au traitement d'image. Pour comparer deux images, il est souvent nécessaire de commencer par égaliser leurs histogrammes de couleurs ou de niveaux de gris. Cette égalisation est en pratique réalisée en calculant le transport optimal entre les deux histogrammes. Or certains codages de couleurs, tels le HSV ou le HSL ont une composante périodique, ce qui fait que même dans le cas simple d'un coût de transport quadratique, il n'existe pas de méthode explicite pour calculer le plan de transport. Pour éviter ce problème, une technique (heuristique) généralement employée consiste à se ramener au cas de la droite en « coupant » le cercle en un point. Si on utilise un coût de transport convexe, le plan de transport optimal est donné par le ré-arrangement monotone et le coût correspondant peut être facilement calculé. On répète alors cette opération de coupure en plusieurs points et choisit comme solution le plan de transport associé au coût le plus faible qui a été obtenu.

Dans ce cadre, mon travail a consisté d'une part à justifier mathématiquement la validité de cette méthode et d'autre part à construire un algorithme permettant de calculer directement et rapidement un point de coupure associé à la solution optimale. La démarche suivie a consisté à construire un problème de minimisation convexe dont ce point est solution. L'algorithme qui en découle est alors une simple descente de gradient dans le cas continu ou une dichotomie dans le cas de mesures discrètes. Enfin on a analysé la complexité de l'algorithme dans ce dernier cas et montré qu'elle était linéaire par rapport au nombre de points. Avec cette méthode, le coût de calcul d'un plan de transport dans le cas du cercle est donc du même ordre de grandeur que dans le cas de la droite. Ce travail est présenté à la section 2.1.1 de ce mémoire et correspond à la publication [M].

Le second travail porte sur le transport en coût concave. Le cadre considéré ici est entièrement discret : on considère deux mesures, l'une représentant des puits, l'autre des sources, définies par deux sommes de distributions de Dirac à coefficients entiers. De manière générale, on peut facilement montrer que le transport en coût concave n'autorise pas de *croisements* tels qu'ils peuvent apparaître dans les plans de transport optimaux en coût convexe : deux trajectoires sont soit disjointes, soit incluses l'une dans l'autre. Cette propriété implique une autre, dite d'*équilibre local* : entre un puits et une source appariés dans un plan de transport optimal, il y a autant de sources que de puits. On peut alors construire des *chaînes*, c'est-à-dire des ensembles alternés de sources et de puits stables par le plan de transport optimal. Ce raisonnement permet de partitionner un problème général en le restreignant à celui des chaînes. Le travail dans ce cadre a débuté par une remarque simple : étant donnée une chaîne, la règle de non-croisement implique que celle-ci contient au moins deux points consécutifs appariés dans le plan de transport optimal. L'idée fut alors de détecter ces deux points, pour retirer itérativement deux par deux tous les points du problème considéré. En pratique, une classe hiérarchique d'indicateurs a été mise en place, ces indicateurs étant basés sur un petit nombre de calculs et permettant de garantir l'optimalité d'appariements de points successifs. Une utilisation itérative de ces fonctions débouche sur un algorithme de recherche simple et facilement parallélisable.

Par la suite, j'ai étendu le cadre d'application de ces indicateurs à des situations plus générales. Deux premières extensions sont relativement simples à obtenir : celles-ci concernent le cas du cercle et le cas où les masses sont rationnelles. La première situation se règle en limitant l'usage des indicateurs à des séries de points de longueurs bornées. La seconde peut être traitée en remarquant tout d'abord que le cas rationnel se ramène au cas entier puis en répartissant les masses

---

entières en masses unitaires sur un petit intervalle autour de la position initiale. On peut ensuite construire les chaînes associées à ce problème, puis rechercher des indicateurs négatifs. Une extension moins évidente a concerné le cas non équilibré. Une série de lemmes permet également de montrer que la démarche suivie dans le cas équilibré peut en fait être appliquée.

La dernière partie de ce travail porte sur la mise au point d'une implémentation efficace, c'est-à-dire minimisant le nombre d'appels de la fonction coût ainsi que des estimations théoriques et empiriques de la complexité de l'algorithme résultant. Ce travail est présenté à la section 2.1.2 de ce mémoire et correspond aux publications [P, Cras. b, Proc. g].

<i>Analyse numérique</i>
--------------------------

Les travaux décrits dans cette partie du mémoire concernent l'accélération de calculs liés à la résolution d'équations aux dérivées partielles. Dans les trois chapitres, une nouvelle étape de résolution est introduite pour produire une méthode plus efficace que la méthode standard. Cette étape consiste soit en un découpage de l'intervalle de temps considéré, soit en une décomposition particulière de l'inconnue et une linéarisation, soit en une série de pré-calculs.

*Parallélisation en temps et contrôle*

Une première méthode générale pour accélérer la résolution d'équations aux dérivées partielles consiste à décomposer le domaine d'espace considéré en sous-domaines sur lesquels des méthodes adaptées peuvent être appliquées de façon indépendante, donc simultanément si l'on dispose de plusieurs processeurs. L'enjeu est alors de *recoller* correctement les solutions obtenues en mettant à jour itérativement (et le moins possible) les valeurs obtenues aux interfaces. Cette démarche est étudiée intensivement depuis une vingtaine d'années et est maintenant bien comprise dans le sens où le temps de calcul d'une solution raisonnable correspond à peu près au temps de calcul sans décomposition divisé par le nombre de sous-domaines. La *full efficiency*, c'est-à-dire un algorithme donnant lieu à une division du temps de calcul par le nombre d'ordinateurs utilisés est donc pratiquement atteinte. L'étape suivante de ce travail de décomposition consiste naturellement à diviser la résolution suivant le domaine temporel. La question posée par cette approche est celle de la mise à jour des conditions initiales intermédiaires. L'algorithme dit *pararéel* introduit par Jacques-Louis Lions, Yvon Maday et Gabriel Turinici a constitué une première étape dans cette direction.

Durant ma thèse et en collaboration avec ces deux derniers, j'ai construit une nouvelle méthode de parallélisation en temps dédiée plus spécifiquement à la résolution de systèmes d'optimalité issus de problèmes de contrôle. Dans le cadre simple considéré, on cherche à atteindre en un temps  $T$  un état cible en partant d'une condition initiale fixée. Pour paralléliser la résolution, l'approche consiste à fixer des états intermédiaires, jouant tantôt le rôle de condition initiale tantôt celui de cible selon le sous-intervalle de la décomposition considéré. Des contrôles optimaux partiels peuvent alors être calculés en parallèle puis concaténés pour permettre une mise à jour des états intermédiaires. L'algorithme résultant exploite largement l'idée de poursuite de trajectoire évoquée plus haut : les états intermédiaires sont en effet construits par interpolation de la trajectoire directe et d'une trajectoire de référence conduisant à l'état cible au temps  $T$ .

Cette approche intuitive a ensuite été complétée par une formalisation qui permet d'identifier l'algorithme à une méthode de direction alternée. Dans ce cadre, j'ai prouvé la convergence de la méthode vers la solution obtenue sans parallélisation. D'un point de vue pratique, le temps de résolution a été approximativement divisé par 8 en utilisant 10 sous-intervalles, ce qui est proche de la *full efficiency*, c'est-à-dire d'une situation où le temps de calcul est divisé par le nombre de processeurs utilisés. Ce travail est présenté à la section 1.1 de ce mémoire et correspond à la publication [E].

La méthode qui vient d'être décrite s'applique à des équations hyperboliques en mettant à profit le fait que l'état adjoint suit une dynamique de même nature que l'état à contrôler : il est alors possible de calculer les points intermédiaires par simple interpolation de trajectoires. En collaboration avec Yvon Maday et Kamel Riahi, j'ai étendu cette notion de points intermédiaires au cas parabolique, où les trajectoires sont cette fois-ci de nature complètement différentes, en particulier parce qu'elles sont irréversibles. La méthode proposée repose sur la construction d'une trajectoire de référence, cette fois-ci différente de la trajectoire adjointe du problème de contrôle. La convergence a également été obtenue théoriquement. Ce travail est présenté à la section 1.2 de ce mémoire.

### *Analyse numérique et formulation co-rotationnelle*

Une autre manière de réduire le temps calcul consiste à construire des représentations rendant efficaces les linéarisations. Durant mon stage post-doctoral à l'Université de Stuttgart, j'ai travaillé en collaboration avec Barbara Wohlmuth et Alexander Weiss sur une problématique de ce type dans le cadre de la dynamique des milieux continus. Dans ce domaine, la plupart des modèles sont non-linéaires et leur simulation est rendue difficile par le coût des sous-boucles internes de résolution. Le problème devient encore plus délicat lorsque les mouvements envisagés sont rapides par rapport aux déformations élastiques. La situation de référence, nécessaire au calcul de la déformation élastique, est dans ce cas très difficile à évaluer d'un pas de temps à l'autre et aucune technique de linéarisation directe ne donne lieu à des résultats satisfaisants. Si aucune linéarisation n'est envisagée, les boucles internes convergent difficilement ou très lentement.

Pour pallier ce problème, les ingénieurs utilisent en général une décomposition du mouvement en une partie solide, sans déformation, et une partie purement élastique, sans mouvement de translation ou de rotation globale. Mais cette formulation dite *co-rotationnelle* est en général associée à des discrétisations ne préservant numériquement ni l'énergie mécanique, ni le moment cinétique du solide.

C'est ce point de départ a conduit à proposer un schéma numérique conservatif. La démarche suivie revient à étendre le schéma simple et classique de Crank-Nicholson au cas du mouvement relatif dans un référentiel en mouvement solide. La première étape de ce travail a consisté à formaliser correctement ce dernier. Si la translation solide fut facile à définir, la caractérisation de la rotation globale s'avéra en revanche délicate : les non-linéarités engendrées d'une part par le changement de référentiel et dans une moindre mesure par le modèle lui même rendent en effet un découplage des variables relativement difficile. Il peut cependant être effectué par l'introduction d'une variable auxiliaire  $s$ , la vitesse relative, et des conditions d'orthogonalité entre le mouvement solide et la déformation élastique. Dans ce cadre, la variable  $s$  peut être utilisée comme pivot dans les relations de Crank-Nicholson pour reproduire au niveau discret les calculs

---

de conservation d'énergie du cas continu. La seconde étape consiste à définir la rotation solide de telle sorte que le moment cinétique soit également préservé numériquement.

J'ai ensuite montré l'existence de solutions au système d'équations résultant et obtenu des bornes pour en faciliter la résolution. D'un point de vue pratique, il s'est avéré que l'algorithme permet une linéarisation efficace dans le cas de petites déformations et réduit le nombre de sous-itérations dans le cas non-linéaire. La dernière partie de ce travail a consisté à étendre le schéma au cas de la dimension trois, où la rotation est caractérisée non plus par un réel, mais par un vecteur. Ce travail est présenté à la section 2.1 de ce mémoire et correspond à la publication [G].

Dans un second travail, pour lequel Patrice Hauret nous a rejoint, nous avons mis en place un couplage de l'algorithme précédent avec des méthodes liées aux problèmes de contact. Ce travail technique a surtout consisté à définir des méthodes adaptées à la description co-rotationnelle du mouvement. Il a en particulier fallu construire des adaptations des algorithmes de Laursen et Chawla [16, 8] pour ce qui concerne le frottement et de la *primal-dual active set strategy* de Kunisch et Ito [15] pour la détermination des points de contact.

Bien que le cas linéaire avec frottement solide nécessite trois sous-boucles internes, une pour la détermination de la rotation, une pour les points de contact et une pour les conditions de frottement, l'approche proposée s'est avérée efficace, dans le sens où les résultats sur des benchmarks ont été reproduits avec des temps de calculs courts. Ce travail est présenté à la section 2.2 de ce mémoire et correspond à la publication [I].

### *Algorithmes d'accélération par pré-calcul*

La dernière technique générale d'accélération des calculs sur laquelle j'ai travaillé est celle du pré-calcul. L'idée est ici de profiter d'une phase préliminaire de calcul dits *offline* pour rendre les simulations plus rapides, soit en approchant les opérateurs par des approximations fines déduites rapidement du pré-calcul, soit en réduisant le nombre de variables dans les méthodes de base réduite.

Dans un premier travail réalisé en collaboration avec Lucie Baudouin et Gabriel Turinici, j'ai analysé et proposé des améliorations à une méthode de pré-calcul, dite du *Toolkit*, introduite par des chimistes pour traiter des problèmes de contrôle quantique. Dans ce domaine, de nombreuses résolutions de l'équation de Schrödinger, étaient nécessaires pour obtenir une approximation correcte du contrôle optimal. Il est donc crucial de disposer de méthodes de calcul efficaces. L'idée introduite par les chimistes consiste à améliorer la précision du calcul en considérant comme seule approximation celle du contrôle. La méthode repose ainsi sur le fait que lorsque celui-ci est constant, la discrétisation en temps conduit à une résolution exacte lorsqu'on utilise des exponentiations d'opérateurs. Puisqu'on dispose en pratique aussi bien qu'en théorie de bornes sur les valeurs du contrôle, on peut pré-calculer ces exponentielles pour un nombre grand, mais fini, de valeurs du contrôle, qui se trouve ainsi quantifié.

Une analyse a permis de montrer que cette méthode était plus efficace que les méthodes utilisées habituellement, telles que le *splitting* d'opérateur. Dans cette dernière par exemple, les constantes apparaissant dans l'estimation d'erreur dépendent de la norme du contrôle et nécessite l'utilisation de pas de temps très petits pour obtenir une bonne précision. J'ai montré que la méthode du *toolkit* n'a pas ce biais. Dans un second temps, j'ai proposé deux améliorations de l'algorithme initial. Le but est ici de montrer comment augmenter les ordres de convergence, sans accroître

de manière rédhibitoire le coût du calcul. Ce travail est présenté à la section 3.1 de ce mémoire et correspond à la publication [L].

Un second travail dans cette direction de recherche a été réalisé en collaboration avec Bernard Haasdonk et Barbara Wohlmuth. Le cadre de ce travail est celui des méthodes de base réduite. Dans cette approche, on pré-calculé un grand nombre de solutions numériques d'une équation en en faisant varier les paramètres. On extrait alors de cet ensemble une famille de petite taille qui est ensuite utilisée comme base de résolution par le biais d'une méthode de Galerkin. Cette méthode a connu ces dernières années beaucoup de développements, mais ne couvrait pas les problèmes formulés sous forme d'inégalités variationnelles.

Le travail a consisté dans un premier temps à mettre en place un cadre fonctionnel adapté à une résolution en base réduite de ces inégalités. L'intérêt majeur de la formulation proposée est que le problème initial et le problème réduit ont exactement la même structure, si bien que toutes les méthodes habituelles peuvent être directement appliquées aux inéquations réduites. Dans une seconde étape, j'ai construit plus spécifiquement un procédé d'orthogonalisation permettant de stabiliser la résolution numérique du système réduit. Ce travail est présenté à la section 3.2 de ce mémoire.

---

## Liste des publications

### Articles de journaux à comité de lecture

- A. " Optimal molecular alignment and orientation through rotational ladder climbing ",  
J. Salomon, C. Dion, G. Turinici, *J. Chem. Phys.* 123 (14), 144310 (2005).
- B. " Monotonic time-discretized schemes in quantum control ",  
Y. Maday, J. Salomon, G. Turinici, *Num. Math.* 103 (2), pp. 323-338 (2006).
- C. " On the relationship between the local tracking procedures and monotonic schemes in quantum optimal control ",  
J. Salomon, G. Turinici, *J. Chem. Phys.* 124 (7), 074102 (2006).
- D. " Convergence of the time-discretized monotonic schemes ",  
J. Salomon, *M2AN*, 41 (1), pp. 77-93 (2007).
- E. " Parareal in time control for quantum systems ",  
Y. Maday, J. Salomon, G. Turinici, *SIAM J. Num. Anal.* 45 (6), pp. 2468-2482 (2007).
- F. " Constructive solution of a bilinear control problem ",  
L. Baudouin, J. Salomon, *Syst. Cont. Lett.*, 57, pp. 453-464 (2008).
- G. " Energy conserving algorithms for a co-rotational formulation ",  
J. Salomon, A. Weiss, B. Wohlmuth, *SIAM J. Num. Anal.* 46 (4), pp. 1842-1866 (2008).
- H. " Cascadic non-linear conjugate gradient solution to finite-level quantum optimal control problems ",  
A. Borzi, J. Salomon, S. Volkwein, *J. Comp. App. Math.* 216 (1), pp. 170-197 (2008).
- I. " Energy consistent co-rotational schemes for frictional contact problems ",  
P. Hauret, J. Salomon, A. Weiss, B. Wohlmuth, *SIAM J. Sci. Comp.* 30 (5), pp. 2488-2511 (2008).
- J. " A stable toolkit method in quantum control ",  
M. Belhadj, J. Salomon, G. Turinici, *J. Phys. A* 41 (36), pp. 362001-362011 (2008).
- K. " A monotonic method for solving nonlinear optimal control problems ",  
J. Salomon, G. Turinici, *soumis (2009). Preprint HAL : hal-00335297*.
- L. " Analysis of the Toolkit method for the time-dependant Schroedinger equation ",  
L. Baudouin, J. Salomon, G. Turinici, *soumis (2009). Preprint HAL : hal-00403798* .
- M. " Fast transport optimization on the circle ",  
J. Delon, J. Salomon, A. Sobolevskii, *SIAM J. App. Math.* 70 (7), pp.2239-2258 (2010).
- N. " A smoothing monotonic convergent optimal control algorithm for NMR pulse sequence design ",  
I. I. Maximov, J. Salomon, G. Turinici, N. C. Nielsen, *J. Chem. Phys.* 132, 084107-1-084107-9 (2010).
- O. " Computation of mean field equilibria in economics ",  
A. Lachapelle, J. Salomon, G. Turinici, *M3AS*, 20 (4) pp. 567-588 (2010).
- P. " Local matching indicators for transport problems with concave costs",  
J. Delon, J. Salomon, A. Sobolevskii, *soumis (2010). Preprint HAL : hal-00525994*.

### Proceedings et actes de conférences à comité de lecture

- Proc a. " Development and calibration of a modeling tool for the analysis of clinical data in human nutrition ",  
B. Juillet, J. Salomon, D. Tomé, H. Fouillet, *ESAIM Proceedings, Vol. 14 (September 2005), pp. 124-155*.

- Proc b. " Discretely monotonically convergent algorithm in quantum control ",  
J. Salomon, G. Turinici, Y. Maday, *Proceedings of the LHMNLC03 IFAC conference, p 321, Sevilla, 3-5 April 2003.*
- Proc c. " Control of molecular orientation and alignment by monotonic schemes ",  
J. Salomon, G. Turinici, *Proceedings of the 24-th IASTED International Conference on modelling, identification and control, 457-187, pp 64-68, Innsbruck, 16-18 February 2005.*
- Proc d. " Limit points of the monotonic schemes in quantum control ",  
J. Salomon, *Proceedings of the 44th IEEE Conference on Decision and Control, Sevilla, 12-15 Decembre 2005.*
- Proc e. " A monotonic algorithm for the optimal control of the Fokker-Planck equation ",  
G. Carlier, J. Salomon, *Proceedings of the 47th IEEE Conference on Decision and Control, Cancun, 9-11 Decembre 2008.*
- Proc f. " A greedy algorithm for the identification of quantum systems ",  
Y. Maday, J. Salomon, *Proceedings of the 48th IEEE Conference on Decision and Control, Shanghai, 16-18 Decembre 2009.*
- Proc g. "Local matching indicators for transport with concave cost",  
J. Delon, J. Salomon, A. Sobolevskiĭ, *soumis (2010). Preprint HAL : hal-00437885.*

### **Notes aux Comptes-rendus de l'Académie des sciences**

- Cras a. " Constructive solution of a bilinear quantum control problem ",  
J. Salomon, L. Baudouin, *C. R. Acad. Sci. Paris, Ser. I, 342, pp. 119-124 (2006).*
- Cras b. " Local matching indicators for concave transport costs ",  
J. Delon, J. Salomon, A. Sobolevskiĭ, *C. R. Acad. Sci. Paris, Ser. I, 348, pp. 901-905 (2010). Preprint HAL : hal-00437885.*

Première partie

Optimisation



# Chapitre 1

## Algorithmes monotones pour le contrôle optimal

**Résumé** : une certaine classe de schémas d'optimisation, dit *schémas monotones* est très régulièrement utilisée en contrôle quantique. Ces méthodes itératives permettent en particulier de calculer des champs lasers favorisant des transformations moléculaires à l'échelle de l'atome ou de la molécule. Ces schémas peuvent cependant s'avérer très instables numériquement lors d'une implémentation directe.

Cette partie commence par une présentation générale de l'approche et d'un cadre général où elle s'applique [K]. Une interprétation en terme d'algorithmes dits de *tracking* [C] est ensuite donnée (les schémas monotones ne reposaient jusqu'alors que sur un certain nombre de manipulations algébriques). Dans un troisième temps, on se penche sur l'analyse numérique de la procédure et on établit des critères garantissant la stabilité ces schémas [B] (voir aussi [Proc. b]). Enfin, on esquisse la preuve de convergence de la suite de contrôles produite par la procédure discrétisée en temps [D] et l'algorithme général [F] (voir aussi [Cras. a] et pour une approche différente [Proc. d]). Cette question de convergence restait ouverte depuis l'introduction des schémas monotones.

En collaboration avec des chimistes, différentes adaptations ont été mises en place pour traiter certains problèmes d'intérêt dans cette communauté [A, N, Proc. f]. Ces contributions font l'objet de la dernière partie du chapitre.

### Introduction

Dans ce premier chapitre, le domaine d'application considéré est celui du contrôle quantique : il s'agit ici de guider à l'aide d'un laser et en un temps donné l'évolution d'un système quantique vers un objectif. Celui-ci peut par exemple être l'approche d'un état cible ou la maximisation d'une grandeur physique associée au système. Deux éléments du formalisme de la chimie quantique sont nécessaires à la compréhension de ce qui suit. D'une part, les systèmes quantiques sont représentés par des *fonctions d'ondes*, généralement notées  $\psi$ , à valeurs complexes, définies sur des espaces hilbertiens paramétrant les états du système considéré. Le carré du module de ces fonctions représente la densité de probabilité des états du système. Pour simplifier, on notera en conséquence génériquement  $L^2$  l'espace des fonctions d'ondes. D'autre part, à toute grandeur physique  $G$  d'un système mesurable expérimentalement est associé un opérateur bilinéaire  $O$ . La relation entre ces deux entités est  $G = \langle \psi, O(\psi) \rangle$ , où  $\langle \cdot, \cdot \rangle$  est le produit hermitien associé à l'ensemble de définition de  $\psi$ . Ces grandeurs sont appelées *observables*.

Le problème de contrôle optimal peut quant à lui être formalisé mathématiquement par l'introduction d'une fonctionnelle dont on va chercher à minimiser ou maximiser la valeur. Deux fonctionnelles représentatives sont :

$$\begin{aligned} J_1(\varepsilon) &= \|\psi(T) - \psi_{cible}\|_2^2 + \alpha \int_0^T \varepsilon(t)^2 dt, \\ &= 2 - \langle \psi(T), \psi_{cible} \rangle + \alpha \int_0^T \varepsilon(t)^2 dt, \end{aligned} \quad (1.1)$$

que l'on s'efforcera de minimiser, et

$$J_2(\varepsilon) = \langle \psi(T) | O | \psi(T) \rangle - \alpha \int_0^T \varepsilon(t)^2 dt, \quad (1.2)$$

que l'on cherchera au contraire à maximiser. Dans ces fonctionnelles, le contrôle  $\varepsilon \in L^2(0, T)$  est le champ électrique délivré par un laser, l'état  $\psi$  est la fonction d'onde à contrôler, en un temps  $T$ . Le coefficient  $\alpha$  permet la pénalisation du contrôle, de sorte que sa norme  $L^2(0, T)$  reste acceptable physiquement. Dans la première fonctionnelle,  $\psi_{cible}$  est un état cible. Dans la seconde,  $O$  est l'observable associée à une grandeur physique que l'on souhaite maximiser. Un état initial  $\psi_0$  étant donné, l'évolution du système est régie par l'équation de Schrödinger, qui s'écrit en unités atomiques sous la forme :

$$i\partial_t \psi = [H_0 - \mu\varepsilon]\psi. \quad (1.3)$$

Le terme  $H_0$  est appelé Hamiltonien. Il caractérise la dynamique intrinsèque du système considéré. Dans de nombreuses applications, il peut s'écrire en unité atomique sous la forme :

$$H_0 = -\Delta + V,$$

le terme  $\Delta$  étant le Laplacien, qui est l'opérateur associé à l'énergie cinétique du système, et le terme  $V = V(x) \in \mathcal{L}(L^2, L^2)$  représentant le potentiel électrostatique dans lequel évolue le système. L'opérateur  $\mu \in \mathcal{L}(L^2, L^2)$  correspond quant à lui au moment dipolaire, il caractérise l'interaction entre le système et le laser. Le terme  $\mu\varepsilon$  est en fait le résultat d'une approximation : il provient en effet de la linéarisation de modèles plus fins où l'interaction est une fonction complexe de  $\varepsilon$ . Le cadre considéré est donc celui du contrôle bilinéaire, puisque la variable de contrôle multiplie l'état dans l'équation d'évolution.

La méthode dont il est question dans cette partie a pour but d'optimiser des fonctionnelles de la forme de  $J_1$  et  $J_2$ . Elle fut introduite dans le cadre du contrôle quantique par D. Tannor [26] (en s'appuyant sur des travaux V. F. Krotov) et par W. Zhu et H. Rabitz [31] sous deux formes différentes. Leurs deux formulations furent par la suite unifiées dans un travail de Y. Maday et G. Turinici [21] qui mirent ainsi en évidence une classe plus large d'algorithmes monotones. Depuis son introduction, de nombreuses variantes ont été proposées et testées sur différents modèles issus de la chimie quantique. Pour autant, ces développements ont toujours pour finalité l'étude (numérique) de l'interaction laser matière, et très peu de travaux portent sur les aspects mathématiques de la méthode. L'objet de ma thèse était justement de combler cette lacune, en étudiant celle-ci sous un angle mathématique. Une grande partie de ce chapitre est consacré à la présentation de mes contributions dans ce domaine. Elle correspond à la section 1.1 du chapitre. Une deuxième partie, la section 1.2, concerne les variantes de la méthode que j'ai conçues en collaboration avec des chimistes.

## 1.1 Conception et analyse des algorithmes monotones

Dans cette partie, je présente les différents résultats mathématiques obtenus sur la méthode. Je commence paradoxalement par le plus récent, qui concerne le formalisme général dans lequel les algorithmes monotones s'appliquent. Dans les trois parties qui suivent, je résume les solutions que j'ai proposées à trois problèmes qui se sont posés au début de ma thèse : l'interprétation de l'algorithme, puisque celui-ci ne reposait que sur une manipulation algébrique astucieuse des variations de la fonctionnelle ; la discrétisation, puisque de nombreuses implémentations mettaient en évidence une instabilité numérique de la méthode ; la convergence enfin, qui restait à l'époque un problème théorique ouvert.

### 1.1.1 État adjoint et factorisation

*Cette partie correspond à l'article [K].*

Si elle permet d'introduire le calcul sur lequel reposent les algorithmes monotones, le but poursuivi dans cette publication était avant tout de présenter un cadre général dans lequel la méthode peut être appliquée.

#### Principe du calcul

Je commence par présenter les idées sur lesquelles reposent les algorithmes monotones sur le cas simple d'une équation différentielle ordinaire. Soit  $A, B, C$  trois matrices carrées de  $\mathcal{M}_n(\mathbb{R})$ ,  $C$  étant symétrique positive, et deux réels  $\alpha > 0$  and  $T > 0$ . On considère le problème de contrôle optimal associé à la maximisation de la fonctionnelle  $J$  définie par :

$$J(v) = y(T) \cdot Cy(T) - \alpha \int_0^T v^2(t) dt,$$

où " $\cdot$ " est le produit scalaire usuel de  $\mathbb{R}^n$ . Ici, l'état  $y : [0, T] \rightarrow \mathbb{R}^n$  et le contrôle  $v : [0, T] \rightarrow \mathbb{R}$  sont liés par l'équation :

$$\begin{cases} y'(t) = (A + v(t)B)y(t), \quad \forall t \in (0, T) \\ y(0) = y_0, \end{cases} \quad (1.4)$$

la condition initiale  $y_0$  étant fixée.

Étant donné deux contrôles  $v$  et  $\tilde{v}$  ainsi que leurs états correspondants  $y$  et  $\tilde{y}$ , on note tout d'abord que

$$\begin{aligned} J(\tilde{v}) - J(v) &= (\tilde{y}(T) - y(T)) \cdot C(\tilde{y}(T) - y(T)) + 2(\tilde{y}(T) - y(T)) \cdot Cy(T) \\ &\quad - \alpha \int_0^T (\tilde{v}(t) - v(t))(\tilde{v}(t) + v(t)) dt. \end{aligned} \quad (1.5)$$

On introduit maintenant un état auxiliaire  $z : [0, T] \rightarrow \mathbb{R}^n$ , souvent appelé état adjoint, associé à  $y$  et  $v$  par

$$\begin{cases} z'(t) &= -(A^* + v(t)B^*)z(t), \\ z(T) &= Cy(T) \end{cases}$$

où  $A^*$  et  $B^*$  sont les matrices transposées de  $A$  et  $B$ . La variable  $z$  est bien entendu le multiplicateur de Lagrange du problème d'optimisation associé à la contrainte (1.4).

On développe alors le second terme du membre de droite de (1.5) de la manière suivante :

$$(\tilde{y}(T) - y(T)) \cdot Cy(T) = \int_0^T (\tilde{v}(t) - v(t)) B\tilde{y}(t) \cdot z(t) dt.$$

Finalement, on obtient :

$$J(\tilde{v}) - J(v) = (\tilde{y}(T) - y(T)) \cdot C(\tilde{y}(T) - y(T)) + \alpha \int_0^T (\tilde{v}(t) - v(t)) \left( \frac{2}{\alpha} B\tilde{y}(t) \cdot z(t) - \tilde{v}(t) - v(t) \right) dt.$$

Une façon simple de garantir que  $\tilde{v}$  conduit à une valeur de la fonctionnelle plus élevée qu'avec  $v$  est par exemple d'imposer que :

$$(\tilde{v}(t) - v(t)) \left( \frac{2}{\alpha} B\tilde{y}(t) \cdot z(t) - \tilde{v}(t) - v(t) \right) \geq 0,$$

ce qui peut être par exemple garanti en définissant  $\tilde{v}$  par :

$$\tilde{v}(t) - v(t) = \left( \frac{2}{\alpha} B\tilde{y}(t) \cdot z(t) - \tilde{v}(t) - v(t) \right)$$

soit :

$$v^{k+1} = \frac{1}{\alpha} B\tilde{y}(t) \cdot z(t).$$

En itérant le procédé, on obtient une suite  $(v^k)_{k \in \mathbb{N}}$  définie itérativement par l'équation implicite  $v^{k+1} = \frac{1}{\alpha} B y^{k+1}(t) \cdot z^k(t)$ , où  $y^{k+1}$  et  $z^k$  correspondent aux contrôles  $v^{k+1}$  et  $v^k$  respectivement. Cette suite optimise  $J$  de manière monotone puisque

$$J(v^{k+1}) - J(v^k) = (y^{k+1}(T) - y^k(T)) \cdot C(y^{k+1}(T) - y^k(T)) + \alpha \int_0^T (v^{k+1}(t) - v^k(t))^2 dt \geq 0.$$

## Le cadre

Je commence par quelques notations : étant donné deux espaces (de Banach)  $\mathcal{A}$  et  $\mathcal{B}$ , on note  $\mathcal{L}(\mathcal{A}, \mathcal{B})$  l'espace des opérateurs linéaires continus de  $\mathcal{A}$  dans  $\mathcal{B}$ . Étant donné une fonction à valeurs réelles ou complexes  $\varphi$ , on note par  $\nabla_x \varphi$  son gradient par rapport à la variable  $x$ . Enfin les symboles  $D_x$  et  $D_{x,x}$  représentent les dérivées premières et secondes (au sens de Fréchet) de fonction vectorielles.

On se donne maintenant trois espaces de Hilbert  $E$ ,  $\mathcal{H}$  et  $\mathbf{V}$ , avec  $\mathbf{V}$  dense dans  $\mathcal{H}$ , et on représente par  $\cdot_E$  et  $\langle \cdot, \cdot \rangle_{\mathbf{V}}$  les produits scalaires associés aux espaces  $E$  et  $\mathbf{V}$ . Le problème de contrôle optimal considéré est le suivant :

$$\min_v J(v),$$

où

$$J(v) := \int_0^T F(t, v(t), X(t)) dt + G(X(T)).$$

Les fonction  $F : \mathbb{R} \times E \times \mathbf{V} \rightarrow \mathbb{R}$  et  $G : \mathbf{V} \rightarrow \mathbb{R}$  sont supposées différentiables et telles que les intégrales soient bien définies. Le système que l'on souhaite contrôler est décrit par la fonction d'état  $X(t) \in \mathbf{V}$  dont l'évolution est régie par l'équation linéaire :

$$\partial_t X(t) + A(t, v(t))X(t) = B(t, v(t)) \tag{1.6}$$

$$X(0) = X_0. \tag{1.7}$$

où  $v : [0, T] \rightarrow E$  est le terme de contrôle. L'opérateur borné  $A(t, v) : \mathbb{R} \times E \times \mathcal{H} \rightarrow \mathcal{H}$  est tel que pour presque tout  $t \in [0, T]$  le domaine de  $A(t, v)^{1/2}$  contienne  $\mathbf{V}$ ; on se place de plus dans le cas où  $B(t, v)$  est tel que pour tout  $t \in [0, T]$  et tout  $v \in E$  on ait  $B(t, v) \in \mathcal{L}(\mathcal{H}, \mathcal{H}) \cap \mathcal{L}(\mathbf{V}, \mathbf{V}^*)$ . La formulation précise des hypothèses de régularité sur  $A, B, F, G$  sera donnée au lemme 2 et au théorème 1.

On suppose par contre dès maintenant que les fonctions  $F$  et  $G$  sont concaves par rapport à l'état  $X$ , dans le sens où :

$$\forall X, X' \in \mathbf{V}, G(X') - G(X) \leq \langle \nabla_X G(X), X' - X \rangle_{\mathbf{V}}, \quad (1.8)$$

$$\forall t \in \mathbb{R}, \forall v \in E, \forall X, X' \in \mathbf{V}, F(t, v, X') - F(t, v, X) \leq \langle \nabla_X F(t, v, X), X' - X \rangle_{\mathbf{V}}. \quad (1.9)$$

Au contraire des hypothèses qui seront faites plus tard, les propriétés (1.8), (1.9) ainsi que la linéarité (1.6) sont cruciales dans la conception d'algorithmes monotones.

### Factorisation

Reprenons maintenant le raisonnement du début de cette section pour expliciter les variations de la fonctionnelle entre deux états. Par souci de clarté, on note  $X_v$  l'état associé à un contrôle  $v$  via les équations (1.6–1.7). On commence par introduire un état adjoint, associé à un contrôle  $v$  et défini par :

$$\partial_t Y_v(t) - A^*(t, v(t))Y_v(t) + \nabla_X F(t, v(t), X_v(t)) = 0 \quad (1.10)$$

$$Y_v(T) = \nabla_X G(X_v(T)). \quad (1.11)$$

On a alors le résultat suivant.

**Lemme 1.** *Pour tout  $v', v : [0, T] \rightarrow E$ , notons*

$$\begin{aligned} \Upsilon(t, X_v(t), v(t), v'(t), Y_v(t), X_{v'}(t)) &= -\langle Y_v(t), (A(t, v'(t)) - A(t, v(t)))X_{v'}(t) \rangle_{\mathbf{V}} \\ &\langle Y_v(t), B(t, v'(t)) - B(t, v(t)) \rangle_{\mathbf{V}} + F(t, v'(t), X_{v'}(t)) - F(t, v(t), X_{v'}(t)). \end{aligned}$$

Alors :

$$J(v') - J(v) \leq \int_0^T \Upsilon(t, X_v(t), v(t), v'(t), Y_v(t), X_{v'}(t)) dt. \quad (1.12)$$

Les variations de la fonctionnelle sont donc majorées par une quantité qui peut être vue comme une factorisation implicite de la fonctionnelle par  $v' - v$  :

$$\Upsilon(t, X_v(t), v(t), v'(t), Y_v(t), X_{v'}(t)) = \Delta(v, v')(t) \cdot_E (v'(t) - v(t)).$$

On a en particulier le résultat suivant :

**Lemme 2.** *Supposons que :*

- $\mathcal{A}, \mathcal{B}, F$  soient de classe  $C^2$  par rapport à  $v$  avec  $D_{vv}A, D_{vv}B$  bornés uniformément dès que  $X, Y$  sont dans un ensemble borné ;
- $\nabla_v F$  soit de classe  $C^1$  par rapport à  $X$  ;
- $D_{vv}F(t, \cdot, X)$  soit bornée, continue positive et minorée par une certaine fonction

$$X \mapsto k(\|X\|).$$

Alors il existe une fonction  $\Delta(\cdot, \cdot; t, X, Y) \in C^0(E^2, E)$  telle que, pour tout  $v, v' \in E$

$$\Delta(v', v; t, X, Y) \cdot_E (v' - v) = - \left\langle Y, \left( A(t, v') - A(t, v) \right) X + B(t, v') - B(t, v) \right\rangle_{\mathbf{V}} + F(t, v', X) - F(t, v, X).$$

De plus, si  $\mathcal{A}, \mathcal{B}, F$  est de classe  $C^1$  par rapport à  $v$ , la fonction  $\Delta(\cdot, \cdot; t, X, Y)$  peut être définie de manière explicite par l'équation :

$$\Delta(v', v; t, X, Y) = \int_0^1 -\nabla_w \left( \langle Y, A(t, w)X - B(t, w) \rangle_{\mathbf{V}} \right) \Big|_{w=v+\lambda(v'-v)} + \nabla_v F(t, v + \lambda(v' - v), X) d\lambda.$$

On remarque que  $v'$  peut toujours être choisi pour rendre l'intégrande négatif dans (1.12) (dans le pire des cas, ce dernier peut être annulé par le choix  $v'(t) = v(t)$ ).

**Remarque 1.** Ce calcul peut également être généralisé en utilisant un troisième contrôle  $\tilde{v}$  dans (1.10) pour calculer la trajectoire de l'état adjoint.

### Algorithme

L'estimation (1.12) permet de construire l'algorithme d'optimisation suivant :

**Algorithme 1.** ( Algorithme monotone )

Étant donné un contrôle  $v^0$ , on construit la suite  $(v^k)_{k \in \mathbb{N}}$  itérativement par la procédure suivante :

1. Calcul de la solution  $X_{v^k}$  de (1.6-1.7) avec  $v = v^k$ .
2. Calcul de la solution  $Y_{v^k}$  de (1.10-1.11) avec  $v = v^k$ , partant de

$$Y_{v^k}(T) := \nabla_X G(X_{v^k}(T)).$$

3. Calcul de  $v^{k+1}$  et de  $X_{v^{k+1}}$  tels que pour  $t \leq T$  :

$$\Delta(v^{k+1}, v^k)(t) \cdot_E (v^{k+1}(t) - v^k(t)) \leq 0. \quad (1.13)$$

Une façon simple de garantir (1.13), est de définir  $v^{k+1}$  par l'équation :

$$v^{k+1}(t) - v^k(t) = -\frac{1}{\theta} \Delta(v^{k+1}, v^k)(t), \quad (1.14)$$

où  $\theta$  est un nombre réel positif, qui peut éventuellement aussi dépendre de  $k$  et/ou de  $t$ . Si  $v^{k+1}$  satisfait (1.14), les variations de  $J$  entre deux étapes de l'algorithme vérifient :

$$J(v^{k+1}) - J(v^k) \leq -\theta \int_0^T (v^{k+1}(t) - v^k(t))^2 dt.$$

**Remarque 2.** Comme indiqué dans la remarque 1, une variante de cet algorithme consiste à optimiser le contrôle également pendant le calcul de la trajectoire adjointe. Ceci revient à introduire un contrôle auxiliaire  $\tilde{v}^k$ , calculé entre  $v^k$  et  $v^{k+1}$ . La factorisation doit alors être remplacée par une somme de deux termes où apparaissent une factorisation suivant  $(\tilde{v}^k - v^k)$  et suivant  $(v^{k+1} - \tilde{v}^k)$ .

Reste maintenant à énoncer le résultat principal :

**Théorème 1.** *Supposons que  $A, B, F$  satisfont les hypothèse du lemme 2. Supposons également que les opérateurs  $A, B$  soient tels que les systèmes (1.6–1.7) et (1.10–1.11) aient des solutions pour tout  $v \in L^\infty(0, T; E)$  avec  $v \mapsto X, v \mapsto Y$  localement Lipschitz. Alors*

1. *Pour tout  $v \in L^\infty(0, T; E)$ , il existe  $\theta^* > 0$  tel que pour tout  $\theta > \theta^*$ , l'équation non-linéaire*

$$\begin{aligned} \partial_t X_{v'}(t) + A(t, v')X_{v'}(t) &= B(t, v') \\ v'(t) &= \mathcal{V}_\theta(t, v(t), X_{v'}(t), Y_v(t)) \\ X_{v'}(0) &= X_0 \end{aligned}$$

où  $\mathcal{V}_\theta(t, v(t), X_{v'}(t), Y_v(t))$  est l'unique solution de

$$\Delta(v'(t), v(t); t, X_{v'}(t), Y_v(t)) = -\theta(v'(t) - v(t)),$$

a une solution. Ici, l'adjoint  $Y_v$  est défini par (1.10–1.11).

2. *Il existe une suite  $(\theta_k)_{k \in \mathbb{N}}$  tel que l'algorithme 1 mis en œuvre avec  $v^0 \in L^\infty(0, T; E)$  et  $v^{k+1}(t) = \mathcal{V}_{\theta_k}(t, v^k(t), X_{v^{k+1}}(t), Y_{v^k}(t))$  soit monotone et satisfasse*

$$J(v^{k+1}) - J(v^k) \leq -\theta_k \|v^{k+1} - v^k\|_{L^2([0, T])}^2.$$

3. *Si pour tout  $t \in [0, T]$   $v^{k+1}(t) = v^k(t)$  (i.e. l'algorithme s'arrête) alors  $v^k$  est un point critique de  $J : \nabla_v J(v^k) = 0$ .*

Ce théorème garantit le bien-fondé de l'algorithme 1, appliqué avec la stratégie (1.14).

### 1.1.2 Lien avec la poursuite de trajectoire et stratégies d'optimisation

*Cette partie correspond à l'article [C].*

Telle qu'elle est présentée à la section précédente, la conception d'un algorithme monotone repose sur l'introduction de l'état adjoint et quelques manipulations algébriques plus ou moins astucieuses. En particulier, ces techniques ne permettent pas de rapprocher la méthode d'une procédure standard d'optimisation, telle qu'un algorithme de type gradient par exemple. Au cours de tests numériques portant sur des algorithmes de contrôle par *feedback*, j'ai constaté que le critère de monotonie (1.13) est dans le cas du contrôle quantique le même que celui assurant la décroissance d'une certaine fonction de Lyapounov. Cette remarque a permis de donner une interprétation géométrique aux algorithmes monotones.

**Remarque 3.** *L'interprétation présentée dans cette partie est en fait valable dans le cadre plus général des équations hyperboliques. Je la présente dans le cas quantique car c'est dans ce domaine que j'en ai montré l'intérêt.*

### Méthodes de poursuite de trajectoire

Les méthodes dites de poursuite de trajectoire de référence, *Reference trajectory tracking*, qui vont maintenant être présentées proviennent de l'approche standard du contrôle par *feedback*. Le calcul du contrôle qu'elles proposent est en effet effectué à chaque instant en fonction de l'état courant et de telle sorte qu'une fonction de Lyapounov décroisse au cours du temps. Elles s'appliquent en général à des problèmes de contrôle en horizon infini, où l'on cherche à contrôler le

comportement asymptotique d'un système. Un objectif ainsi fréquemment considéré en contrôle quantique consiste à rapprocher la trajectoire du système d'une trajectoire de référence. Le calcul du contrôle est effectué à chaque instant, de telle sorte que la distance entre l'état courant et celui situé sur la trajectoire de référence au même moment diminue. Dans ce cas, cette distance est la fonction de Lyapounov que l'on considère. En contrôle quantique, cette trajectoire est généralement choisie parmi les états propres du système, c'est-à-dire –à un terme de phase près– parmi les vecteurs propres de l'Hamiltonien interne.

On considère maintenant un système quantique décrit par sa fonction d'onde  $\psi(t)$ , dont l'évolution est régie par l'équation de Schrödinger (1.3), où le contrôle  $\varepsilon$  reste à déterminer. Supposons maintenant que l'on souhaite qu'à un temps  $T$ , le système soit proche d'un état cible  $\psi_{cible}$ . Pour adapter la méthode au cadre du contrôle en temps fini, on considère comme trajectoire de référence celle d'un état  $\chi(t)$  partant de manière rétrograde de l'état cible  $\psi_{cible}$  au temps  $T$ . Pour avoir un intérêt du point de vue du contrôle, celle-ci doit relever d'une équation de Schrödinger : dans ce cas, il suffira au système  $\psi(t)$  d'atteindre l'un de ses points pour qu'il la suive jusqu'à l'état cible. On la définit par l'équation :

$$\begin{aligned} i\partial_t\chi &= [H_0 - \mu\varepsilon_{ref}(t)]\chi \\ \chi(T) &= \psi_{cible}. \end{aligned} \quad (1.15)$$

où  $\varepsilon_{ref}$  est un contrôle pour l'instant arbitraire et connu. En suivant la méthode de poursuite de trajectoire décrite ci-dessus, on introduit la fonction de Lyapounov  $J_{\varepsilon, \varepsilon_{ref}}^{fwd}(t)$  définie par :

$$J_{\varepsilon, \varepsilon_{ref}}^{fwd}(t) = -2\Re\langle\psi(t), \chi(t)\rangle + \alpha \int_0^t \varepsilon(t)^2 dt + \alpha \int_t^T \varepsilon_{ref}(t)^2 dt.$$

Le premier terme de cette fonction rend compte de la distance entre les deux états puisque  $\|\psi(t) - \chi(t)\|_{L^2} = 2 - 2\Re\langle\psi(t), \chi(t)\rangle$ . Le second est ajouté pour pénaliser le contrôle.

**Remarque 4.** Cette adaptation des méthodes de poursuite de trajectoire au cadre du contrôle associé à une cible et en temps fini est à la base des techniques de parallélisation en temps décrites au chapitre 1 de la partie II.

La dérivée par rapport au temps de cette fonction est donnée par :

$$\frac{d}{dt}J_{\varepsilon, \varepsilon_{ref}}^{fwd}(t) = (\varepsilon(t) - \varepsilon_{ref}(t)) \cdot (2\Im\langle\chi(t), \mu\psi(t)\rangle + \alpha(\varepsilon(t) + \varepsilon_{ref}(t))). \quad (1.16)$$

A ce stade, il ne reste plus qu'à définir  $\varepsilon(t)$  de telle sorte que cette quantité soit négative.

### Identification des deux méthodes

Oublions maintenant la méthode de poursuite de la section précédente et appliquons la démarche de conception d'un algorithme monotone décrite à la section 1.1.1 à la fonctionnelle  $J_1$  définie par (1.1). Soit donc deux contrôles,  $\varepsilon$  et  $\varepsilon'$ . Le calcul de la variation suivant la méthode des algorithmes monotones donne :

$$J(\varepsilon') - J(\varepsilon) = \int_0^T (\varepsilon'(t) - \varepsilon(t)) \cdot (2\Im\langle\chi(t), \mu\psi'(t)\rangle + \alpha(\varepsilon'(t) + \varepsilon(t))) dt,$$

où  $\psi'$  est la solution de (1.3) avec le contrôle  $\varepsilon'$  et où  $\chi$  est l'état adjoint, solution de (1.15) avec le contrôle  $\varepsilon$ . Autrement dit, le critère de monotonie (1.13) est le même que celui obtenu à l'équation (1.16) par la méthode de poursuite.

On en déduit que l'algorithme monotone repose en fait sur une stratégie de poursuite de la trajectoire de l'état adjoint, celle-ci étant mise à jour après chaque parcours de  $[0, T]$ .

**Remarque 5.** Le raisonnement précédent repose fortement sur le fait que  $J_1$  dépende linéairement de l'état  $\psi$ . Dans le cas où la fonctionnelle n'est plus linéaire mais quadratique, comme c'est le cas avec  $J_2$  (voir (1.2)), l'identification subsiste à condition de coupler la méthode de poursuite de la section précédente avec une méthode de la puissance, où la fonction  $\psi_{cible}$  est mise à jour à chaque itération par une formule du type :  $\psi_{cible} = O(\psi^k(T))$  pour se rapprocher peu à peu d'un vecteur propre de  $O$ .

## Deux premières applications pratiques

Un premier intérêt de l'identification présentée ci-dessus vient du fait que la fonction  $J_{\varepsilon^{k+1}, \varepsilon^k}^{fwd}(t)$  permet de définir une sorte de valeur courante de la fonctionnelle  $J$  pendant le calcul de  $\varepsilon^{k+1}$ . En effet, on a  $J_{\varepsilon^{k+1}, \varepsilon^k}^{fwd}(t_0) = J(\varepsilon_k^{int})$ , où  $\varepsilon_k^{int}$  est défini par :

$$\varepsilon_k^{int}(t) = \begin{cases} \varepsilon^k & \text{si } t < t_0 \\ \varepsilon^{k+1} & \text{sinon.} \end{cases}$$

Si pour une raison ou pour une autre le calcul de  $\varepsilon^{k+1}$  est interrompu, le travail qui aura été fait ne sera pas perdu, puisque  $\varepsilon_k^{int}$  conduira à une meilleure valeur de  $J$ , qui sera de plus connue, sans calcul supplémentaire.

Une deuxième application directe exploite la mesure continue du procédé d'optimisation induit par l'algorithme monotone. À partir de cette interprétation, on peut construire un algorithme monotone stochastique reposant sur le principe suivant : il est souvent observé que les algorithmes monotones ont une convergence lente en fin d'optimisation lorsqu'ils sont utilisés avec une formule du type (1.14) où la valeur de  $\theta$  est constante. Par contre, leur convergence semble être améliorée lorsque la valeur de  $\theta$  varie, au cours du temps et/ou au cours des itérations. L'idée a donc consisté à changer la valeur de  $\theta$  aléatoirement lorsque l'écart entre les deux trajectoires ne diminuait pas suffisamment entre deux pas de temps consécutifs. Cette stratégie met donc à profit l'indicateur que constitue  $J_{\varepsilon^{k+1}, \varepsilon^k}^{fwd}(t)$  pour déclencher un renouvellement des paramètres de la procédure.

### 1.1.3 Discrétisation et implémentation

*Cette partie correspond à l'article [B] et au proceeding [Proc. b].*

Les résultats qui y figurent furent chronologiquement les premiers obtenus, ce qui explique que le formalisme utilisé dans la publication diffère de celui employé dans ce qui suit. Ce travail trouve sa source dans un problème rencontré quasi-systématiquement lors de l'implémentation d'un algorithme monotone : après un certain nombre d'itérations (éventuellement très grand, si le pas de temps utilisé est petit), l'algorithme perd sa monotonie et les contrôles produits n'ont plus de sens physique.

Une manière de résoudre ce problème était de reprendre au niveau discret le calcul développé dans la preuve du lemme 1 pour obtenir un équivalent discret de l'inégalité (1.12). Plutôt que de recopier les calculs qui figurent dans l'article, j'indique les correspondances utilisées entre les cadres continu et discret dans le cas de la fonctionnelle  $J = J_1$ .

On suit donc une démarche *discretize and optimize* (aussi appelée *directe*), c'est-à-dire que l'on part d'une fonctionnelle discrétisée dont on résout le système d'optimalité. Cette approche s'oppose à la stratégie dite *optimize and discretize* (ou encore *indirecte*) qui consiste à discrétiser le système d'optimalité de la fonctionnelle continue. On se donne donc un entier  $N$  et un pas de

temps  $\Delta T$  tels que  $T = N\Delta T$ . La fonctionnelle  $J$  :

$$J(\varepsilon) = 2 - 2\Re\langle\psi_{cible}|\psi(T)\rangle + \alpha \int_0^T \varepsilon(t)^2 dt$$

est remplacée par la version discrète :

$$J_{\Delta T}(\varepsilon) = 2 - 2\Re\langle\psi_{cible}|\psi_N\rangle + \alpha\Delta T \sum_{j=0}^N \varepsilon_j^2,$$

tandis que l'équation de Schrödinger :

$$i\partial_t\psi = [H_0 - \mu\varepsilon]\psi,$$

devient

$$\psi_{j+1} = e^{-iH_0\frac{\Delta T}{2}} e^{-i(V-\mu\varepsilon_j)\Delta T} e^{-iH_0\frac{\Delta T}{2}} \psi_j,$$

où  $j$  est l'index correspondant au pas de temps. On choisit donc ici d'utiliser un schéma numérique de type splitting d'opérateur de Strang. Ce choix était en fait celui qui prévalait dans les publications à l'époque du travail. Pour autant, tout ce qui suit peut être adapté à d'autres schémas numériques (voir par exemple [H], où l'on part d'un schéma de Crank-Nicholson). Dans le cas continu, la factorisation (1.12) s'écrit :

$$J(\varepsilon') - J(\varepsilon) = \alpha \int_0^T (\varepsilon' - \varepsilon) \cdot (\varepsilon' + \varepsilon + \frac{2}{\mathfrak{S}} \alpha \langle \chi, \mu\psi' \rangle) dt$$

qui devient après discrétisation :

$$J_{\Delta T}(\varepsilon') - J_{\Delta T}(\varepsilon) = \alpha\Delta T \sum_{j=0}^{N-1} (\varepsilon'_j - \varepsilon_j) \cdot (\varepsilon'_j + \varepsilon_j + \frac{2}{\alpha} \mathfrak{S} \langle \hat{\chi}_j, \mu^*(\varepsilon_j - \varepsilon'_j) \hat{\psi}'_j \rangle), \quad (1.17)$$

où l'on a introduit (et c'est l'une des innovations apportée par ce travail) l'approximation de  $\mu$  suivante :

$$\mu^*(x) = \frac{e^{-i\mu x \Delta T} - Id}{-ix\Delta T},$$

et les états intermédiaires :

$$\tilde{\psi}_j = e^{iH_0\frac{\Delta T}{2}} \psi_j, \quad \hat{\chi}_j = e^{-iH_0\frac{\Delta T}{2}} \chi_j.$$

Par rapport à la méthode qui donne lieu à (1.12), on a mis en facteur le coefficient  $\alpha$  par commodité. Dans le cas continu, l'algorithme monotone consiste en la résolution itérative du système

$$\begin{cases} i\frac{\partial}{\partial t}\psi^k(t) = (H - \varepsilon^k(t)\mu)\psi^k(t) \\ \psi^k(t=0) = \psi_0 \\ \varepsilon^k(t) = (1 - \frac{2}{1+\theta})\varepsilon^{k-1} + \frac{2}{1+\theta} (-\frac{1}{\alpha}\mathfrak{S}\langle\chi^{k-1}(t)|\mu|\psi^k(t)\rangle) \\ i\frac{\partial}{\partial t}\chi^k(t) = (H - \varepsilon^k(t)\mu)\chi^k(t) \\ \chi^k(t=T) = \psi_{cible}, \end{cases}$$

qui vérifie :

$$J(\varepsilon^{k+1}) - J(\varepsilon^k) = -\alpha\|\varepsilon^{k+1} - \varepsilon^k\|_2^2. \quad (1.18)$$

Dans le cas discret, ce système devient :

$$\begin{cases} \psi_{j+1}^k = e^{-iH_0 \frac{\Delta T}{2}} e^{-i(V - \mu \varepsilon_j^k) \Delta T} e^{-iH_0 \frac{\Delta T}{2}} \psi_j^k \\ \psi_0^k = \psi_0 \\ \varepsilon_j^k = \left(1 - \frac{2}{1+\theta}\right) \varepsilon_j^{k-1} + \frac{2}{1+\theta} \left(-\frac{1}{\alpha} \Im \langle \hat{\chi}_j^{k-1} | \mu^* (\varepsilon_j^k - \varepsilon_j^{k-1}) | \hat{\psi}_j^k \rangle\right) \\ \begin{cases} \chi_j^k = e^{iH_0 \frac{\Delta T}{2}} e^{i(V - \mu \varepsilon_j^k) \Delta T} e^{iH_0 \frac{\Delta T}{2}} \chi_{j+1}^k \\ \chi_N^k = \psi_{cible}, \end{cases} \end{cases}$$

qui vérifie :

$$J_{\Delta T}(\varepsilon^{k+1}) - J_{\Delta T}(\varepsilon^k) = -\alpha \|\varepsilon^{k+1} - \varepsilon^k\|_2^2.$$

Comme dans le cadre abstrait avec le lemme 2, ce schéma nécessite la résolution d'une équation implicite pour déterminer la valeur de  $\varepsilon_j^k$ . L'existence d'une solution est garantie dans ce cas par le résultat suivant, qui est un cas particulier du théorème 1.

**Théorème 2.** *Supposons  $\mu$  borné dans  $\mathcal{L}(L^2, L^2)$ , alors l'équation :*

$$\varepsilon_j^k = f_{j,k}(\varepsilon_j^k), \tag{1.19}$$

avec

$$f_{j,k}(x) = \left(1 - \frac{2}{1+\theta}\right) \varepsilon_j^{k-1} + \frac{2}{1+\theta} \left(-\frac{1}{\alpha} \Im \langle \hat{\chi}_j^{k-1} | \mu^* (x - \varepsilon_j^{k-1}) | \hat{\psi}_j^k \rangle\right),$$

admet une solution.

Pour résoudre (1.19), on peut utiliser une procédure de point fixe sur  $f_{j,k}$ . On a par exemple obtenu le résultat suivant :

**Théorème 3.** *Si  $\frac{2\|\mu\|_{\mathcal{L}(L^2, L^2)}^2 \Delta T}{\alpha(1+\theta)} < 1$ , la procédure de point fixe  $u_{\ell+1} = f_{j,k}(u_\ell)$ , initiée avec  $u_0 = 0$  converge vers la solution de (1.19), qui est unique dans ce cas.*

Ce travail fut ensuite complété par la conception d'un algorithme explicite, basé sur une itération de Newton appliquée à l'intégrande (discret) de (1.17). Une analyse de cette méthode est présentée dans [D], où je prouve en particulier le résultat (voir l'annexe, lemme 5.4) suivant.

**Lemme 3.** *Supposons que  $\gamma = \alpha - 3\Delta T \|\mu\|_{\mathcal{L}(L^2, L^2)}^2$  soit positif, alors :*

$$J_{\Delta T}(\varepsilon^{k+1}) - J_{\Delta T}(\varepsilon^k) \leq -\gamma \|\varepsilon^{k+1} - \varepsilon^k\|_2^2.$$

Ceci implique que les versions explicite et implicite possèdent les mêmes propriétés de monotonie.

#### 1.1.4 Convergence de l'algorithme

*Cette partie correspond aux articles [F] (qui fait suite à la note [Cras. a]) et [D].*

Cette partie peut être considérée comme la plus importante de mes contributions sur les algorithmes monotones, puisqu'elle fournit une preuve de convergence de la méthode. Une preuve plus simple, basée sur un tout autre raisonnement a été également obtenue au prix d'une hypothèse plus forte ( $\alpha$  grand). On la trouvera dans [Proc. d], qui ne sera pas évoqué davantage dans ce manuscrit.

## Principe de la preuve

La preuve de la convergence repose essentiellement sur trois ingrédients :

- la compacité des points critiques de la fonctionnelle,
- une inégalité due à Łojasiewicz,
- une estimation du gradient.

La compacité des points critiques peut être obtenue dans le cas quantique de deux manières : soit via le Lemme d'Aubin [1], soit par en reproduisant le raisonnement de Ball et Slemrod dans [2].

L'inégalité de Łojasiewicz stipule quant à elle qu'étant donné un point critique  $a$ , une fonctionnelle  $\Gamma$  analytique et définie sur un espace de dimension finie vérifie :

$$\|\nabla\Gamma(x)\| \geq |\Gamma(x)|^{1-\nu},$$

pour un certain réel  $\nu \in ]0, 1/2]$ , dès que  $x$  est suffisamment proche de  $a$ . On a ici supposé  $\Gamma(a) = 0$ , ce qui n'est pas restrictif. Par compacité de l'ensemble des points critiques, on peut en fait étendre pour un même  $\nu$  cette inégalité à un voisinage des points critiques. Par adaptation des travaux de L. Simon [25], cette inégalité reste de plus valable en dimension infinie lorsque la fonctionnelle est de Fredholm, i.e. de Hessienne compacte. Ces propriétés sont en fait bien vérifiées dans le cas des fonctionnelles du contrôle quantique. Les preuves sont assez techniques et je ne les développerai pas ici.

Enfin, l'estimation du gradient requise dans la preuve est la suivante :

$$\|\nabla J(\varepsilon^k)\| \leq \lambda \|\varepsilon^{k+1} - \varepsilon^k\|,$$

où  $\lambda$  est un réel positif. Celle-ci s'obtient assez facilement en utilisant la formule

$$\nabla J(\varepsilon) = 2\alpha\varepsilon + \Im\langle \chi|\mu|\psi\rangle$$

et des estimations de Gronwall.

Je me contenterai ici de donner les grandes lignes de la preuve. De la monotonie de l'algorithme et du fait que la fonctionnelle  $J$  est bornée inférieurement, on déduit que la suite de réels  $(J(\varepsilon^k))_{k \in \mathbb{N}}$  converge vers un réel  $\ell_{\varepsilon_0}$ . Si cette suite atteint sa limite en un nombre fini d'itérations, alors d'après (1.18)  $(\varepsilon^k)_{k \in \mathbb{N}}$  devient constante à partir de ce même rang, et donc converge. On peut donc supposer que  $\tilde{J}(\varepsilon) = J(\varepsilon) - \ell_{\varepsilon_0}$  ne s'annule jamais sur les points de la suite  $(\varepsilon^k)_{k \in \mathbb{N}}$ . La preuve de convergence repose sur une série d'inégalités. On a en effet successivement :

$$(\tilde{J}(\varepsilon^k))^\nu - (\tilde{J}(\varepsilon^{k+1}))^\nu \geq \frac{\nu}{(\tilde{J}(\varepsilon^{k+1}))^{1-\nu}} (J(\varepsilon^k) - J(\varepsilon^{k+1})) \quad (1.20)$$

$$\geq \frac{\alpha\nu\theta}{(\tilde{J}(\varepsilon^{k+1}))^{1-\nu}} \|\varepsilon^{k+1} - \varepsilon^k\|_2^2 \quad (1.21)$$

$$\geq \frac{\alpha\nu\theta}{\|\nabla J(\varepsilon^{k+1})\|} \|\varepsilon^{k+1} - \varepsilon^k\|_2^2 \quad (1.22)$$

$$\geq \frac{\alpha\nu\theta}{\lambda} \|\varepsilon^{k+1} - \varepsilon^k\|_2, \quad (1.23)$$

où (1.20) est une simple conséquence de la concavité de  $x \mapsto x^\theta$ , (1.21) provient de la propriété fondamentale de l'algorithme déjà évoquée par l'équation (1.18), (1.22) est obtenue en utilisant l'inégalité de Łojasiewicz et (1.23) suit l'inégalité obtenue sur le gradient.

Soit  $p \in \mathbb{N}$ , l'inégalité (1.23) conduit à :

$$\begin{aligned} (\tilde{J}(\varepsilon^k))^\nu - (\tilde{J}(\varepsilon^{k+p}))^\nu &\geq \frac{\alpha\nu\theta}{\lambda} \sum_{l=k}^{k+p-1} \|\varepsilon^{l+1} - \varepsilon^l\|_2 \\ &\geq \frac{\alpha\nu\theta}{\lambda} \|\varepsilon^{k+p} - \varepsilon^k\|_2, \end{aligned}$$

ce qui démontre, puisque  $(J(\varepsilon^k))_{k \in \mathbb{N}}$  converge, que  $(\varepsilon^k)_{k \in \mathbb{N}}$  est de Cauchy, et donc converge.

### Vitesse de convergence

La vitesse de convergence s'obtient en sommant les inégalités (1.23) entre  $k$  et  $+\infty$  puis en utilisant une nouvelle fois l'inégalité de Łojasiewicz et celle sur le gradient. On obtient le résultat suivant :

**Théorème 4.** *Soit  $\varepsilon^\infty$ , la limite de  $(\varepsilon^k)_{k \in \mathbb{N}}$ . Si  $\nu < \frac{1}{2}$ , il existe  $k_0$  et  $c > 0$  tel que :*

$$\forall k > k_0, \|\varepsilon^k - \varepsilon^\infty\|_2 \leq ck^{-\frac{\nu}{1-2\nu}}.$$

Si  $\nu = \frac{1}{2}$ , il existe  $k'_0$ ,  $c'$  et  $\tau$  tels que :

$$\forall k > k'_0, \|\varepsilon^k - \varepsilon^\infty\|_2 \leq c'e^{-\tau k}.$$

Le cas  $\nu = \frac{1}{2}$  correspond en fait au cas où la Hessienne de  $J$  est inversible, ce qui arrive en particulier lorsque  $\alpha > 2T\|\mu\|_{\mathcal{L}(L^2, L^2)}^2$  (la fonctionnelle est alors convexe). En pratique, on observe numériquement seulement une convergence exponentielle, même lorsque  $\alpha$  est petit, ce qui laisse à penser que l'analyse peut encore être affinée. La courbe 1.1 montre ainsi une telle convergence, alors que les paramètres sont tels que  $\alpha < 10^{-5} \cdot 2T\|\mu\|_{\mathcal{L}(L^2, L^2)}^2$ .

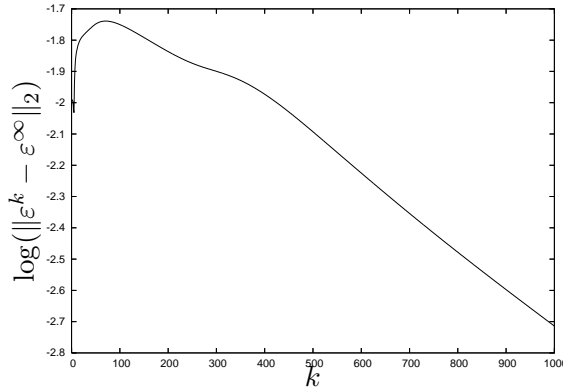


FIGURE 1.1 – Une convergence exponentielle est observée numériquement.

## 1.2 Applications

Cette partie est consacrée aux résultats obtenus en collaboration ou suite à des discussions avec des chimistes. S'ils sont le fruit d'adaptations de la procédure précédente à des problèmes souvent considérés dans les domaines d'application, ils peuvent également être vus comme des travaux de prospective, soulevant de nouvelles questions mathématiques.

### 1.2.1 Contrôle de l’alignement et de l’orientation de molécules

Je résume ici les résultats présentés dans [A]. Ils constituent la prolongation de ceux introduits dans [Proc. c].

Le système physique considéré ici est la molécule de cyanure HCN. L’intérêt porté à ce système provient de son état fondamental qui correspond à une configuration linéaire. La molécule reste dans cet état sur une plage de fréquences susceptible d’être utilisée pour contrôler sa position angulaire. Dans ce qui suit, la molécule est donc identifiée à un rotateur rigide et sa fonction d’onde associée est décrite au moyen de coordonnées sphériques. Ce modèle constitue un exemple simple sur lequel différentes méthodes de contrôle de l’orientation et de l’alignement peuvent être testées.

L’Hamiltonien de ce système est donné par :

$$H = B\hat{J}^2 - \mu_0\varepsilon(t) - \frac{\varepsilon(t)^2}{2}(\Delta\alpha \cos^2\theta + \alpha_\perp),$$

où  $B = \frac{\hbar}{4\pi I}$ , avec  $I$ , moment d’inertie de la molécule, est la *constante rotationnelle* du système,  $\hat{J}^2$  est l’opérateur de Laplace-Beltrami, correspondant au Laplacien,  $\mu_0$  est le moment dipolaire permanent de la molécule et  $\Delta\alpha$  et  $\alpha_\perp$  respectivement le moment dipolaire et le moment induit. Du point technique, on voit que l’Hamiltonien ne contient plus seulement des termes bilinéaires, mais aussi des puissances de  $\varepsilon$ . Ce modèle rentre cependant dans la classe des problèmes auxquels les algorithmes monotones s’appliquent. Dans la littérature, d’autres stratégies d’optimisation ont été testées numériquement. Des algorithmes stochastiques ont en particulier été utilisés pour trouver des contrôles efficaces dans des familles de contrôles paramétrés par un petit nombre de coefficients (pulses sinusoïdaux, paramétrés par leurs fréquences, durées, amplitudes par exemple). Ici, la stratégie permet au contraire de chercher des contrôles optimaux dans un espace de grande dimension.

L’apport principal de ce travail fut d’obtenir des champs laser mettant en évidence des processus de contrôle facilement interprétables : un mécanisme de *ladder climbing*, consistant à faire monter peu à peu le système dans des états d’orientation d’énergies de plus en plus élevés est révélé par les contrôles optimaux. Après analyse par transformée de Fourier à fenêtre glissante, on observe en effet que les fréquences propres du système apparaissent successivement dans le contrôle au cours du temps. La figure 1.2 illustre ce résultat. Il est connu que de telles fréquences permettent de faire transiter le système entre états propres correspondants. L’algorithme retrouve donc sans contrainte les fréquences propres du système.

### 1.2.2 Résonance magnétique nucléaire

Cette partie correspond à la publication [N].

Dans un second travail basé sur un modèle de résonance magnétique nucléaire et mené en collaboration avec des physiciens, j’ai construit une stratégie de couplage entre un algorithme monotone et un filtre fréquentiel. L’idée est d’ajouter une sous-étape d’interpolation à chaque fois qu’un nouveau contrôle est obtenu. L’interpolation est réalisée entre le contrôle obtenu par l’algorithme standard  $\bar{v}^{k+1}$  et sa version filtrée  $\mathcal{F}(\bar{v}^{k+1})$  :

$$v^{k+1} = (1 - \alpha^k)\bar{v}^{k+1} + \alpha^k\mathcal{F}(\bar{v}^{k+1}),$$

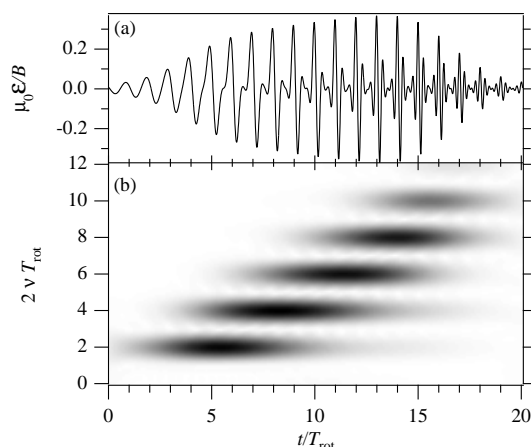


FIGURE 1.2 – Les fréquences propres du système sont « allumées » les unes après les autres.

où le coefficient  $\alpha^k$  est choisi comme étant le plus grand réel tel que l'algorithme reste monotone. Cette procédure de filtrage est en effet nécessaire d'un point de vue physique, les champs magnétiques réalisables techniquement ne contenant pas de hautes fréquences.

Une autre difficulté technique est introduit par ce modèle : le contrôle est ici vectoriel ; dans l'exemple considéré il a 4 composantes, qui sont des coefficients intervenant linéairement dans l'Hamiltonien. De ce fait, il faut optimiser à chaque pas de temps les coefficients de l'argument d'une exponentielle de matrice, qui ne possède pas de propriété de commutation particulière. Une formule de la même forme que (1.12) peut cependant être obtenue : elle permet d'utiliser une procédure d'optimisation standard (en l'occurrence une routine Matlab en dimension 4) pour déterminer les coefficients à chaque pas de temps.

### 1.2.3 Construction de champs sélectifs pour l'identification

Sur une suggestion d'Herschel Rabbitz et en collaboration avec Yvon Maday, j'ai ensuite conçu une procédure de calcul de champs discriminants dans le but de résoudre un problème inverse d'identification de moment dipolaire. Cette méthode met à profit les algorithmes monotones pour calculer des champs lasers tels que les observations d'un système subissant leur excitation soient très sensibles aux variations du moment dipolaire. Le second ingrédient de l'algorithme est une approche gloutonne : à l'étape  $k$  de construction d'une famille de champs, on commence par chercher des paramètres dipolaires auxquels la famille courante n'est pas sensible.

De manière plus précise, on note  $\varphi(\mu, \varepsilon)$  une observation simulée à partir d'un système de moment dipolaire  $\mu$  éclairé avec un laser  $\varepsilon$ . On note également  $\mu^*$  le vrai moment dipolaire. Le problème peut être formulé par la recherche d'une valeur de  $\mu$  solution du problème d'optimisation :

$$\inf_{\mu \in \mathcal{L}(L^2; L^2)} \sup_{\varepsilon \in L^2(0, T)} |\varphi(\mu, \varepsilon) - \varphi(\mu^*, \varepsilon)|^2.$$

Une hypothèse de contrôlabilité du système permet de garantir l'unicité de la solution de ce problème. L'objectif est maintenant de construire itérativement une famille de  $L$  champs sélectifs, où  $L$  est le nombre de coefficients de  $\mu^*$  à identifier. L'algorithme proposé est le suivant.

**Algorithme 2.** (Algorithme glouton de calcul de champs laser sélectifs) Soit  $\varepsilon^1$  un champ laser solution du problème :

$$\sup_{\varepsilon \in L^2(0,T)} |\varphi(\mu_1, \varepsilon)|^2.$$

Supposons qu'à l'étape  $k$ , avec  $1 < k \leq L$ , un champ  $\varepsilon^{k-1}$  ait été calculé. Le calcul de  $\varepsilon^k$  est effectué suivant la procédure suivante :

1. Problème de minimisation : trouver  $(\alpha_j^k)_{j=1, \dots, k-1}$ , solution du problème

$$\begin{cases} \varphi(\sum_{j=1}^{k-1} \alpha_j^k \mu^j, \varepsilon^1) & = \varphi(\mu^k, \varepsilon^1) \\ & \vdots \\ \varphi(\sum_{j=1}^{k-1} \alpha_j^k \mu^j, \varepsilon^m) & = \varphi(\mu^k, \varepsilon^m) \\ & \vdots \\ \varphi(\sum_{j=1}^{k-1} \alpha_j^k \mu^j, \varepsilon^{k-1}) & = \varphi(\mu^k, \varepsilon^{k-1}), \end{cases}$$

au sens des moindres carrés.

2. Problème de maximisation : trouver  $\varepsilon^k$ , solution du problème :

$$\varepsilon^k = \operatorname{argmax}_{\varepsilon \in L^2(0,T)} |\varphi(\mu^k, \varepsilon) - \varphi(\sum_{j=1}^{k-1} \alpha_j^k \mu^j, \varepsilon)|^2.$$

Ici, la famille  $(\mu^j)_j$  est une base de l'ensemble des opérateurs  $\mu$ , tirée aléatoirement. Le problème de minimisation se résout par une procédure standard appliquée à :

$$K^k(\alpha) = \sum_{m=1}^{k-1} |\varphi(\mu^k, \varepsilon^m) - \varphi(\sum_{j=1}^{k-1} \alpha_j \mu^j, \varepsilon^m)|^2.$$

Le problème de maximisation se résout à l'aide d'un algorithme monotone appliqué à :

$$J(\varepsilon) = \langle \tilde{\psi}(T) - \hat{\psi}(T) | O_{\psi_1} | \tilde{\psi}(T) - \hat{\psi}(T) \rangle - \beta \int_0^T \varepsilon^2(t) dt,$$

où  $O_{\psi_1} = \psi_1 \cdot \psi_1^T$  avec  $\psi_1$  un état fixé arbitrairement. Une fois les champs sélectifs calculés, ils sont utilisés dans une procédure standard de moindres carrés appliquée au problème inverse concernant  $\mu^*$  :

$$\begin{cases} \varphi(\sum_{j=1}^L \alpha_j \mu^j, \varepsilon^1) & = \varphi(\mu^*, \varepsilon^1) \\ & \vdots \\ \varphi(\sum_{j=1}^L \alpha_j \mu^j, \varepsilon^k) & = \varphi(\mu^*, \varepsilon^k) \\ & \vdots \\ \varphi(\sum_{j=1}^L \alpha_j \mu^j, \varepsilon^L) & = \varphi(\mu^*, \varepsilon^L), \end{cases} \quad (1.24)$$

où les valeurs  $(\varphi(\mu^*, \varepsilon^k))_{k=1, \dots, L}$  sont obtenues expérimentalement. La procédure a été testée sur l'exemple donné dans [17]. Le moment dipolaire a été retrouvé avec une précision relative de  $10^{-4}$ , en 10 min CPU (sur un processeur Intel Core2 Duo 2.6 GHz). Il est intéressant de voir différentes allures de la fonctionnelle de coût associée à la procédure de moindres carrés associée à (1.24) au voisinage de la solution. Celles-ci n'apparaissent pas dans le proceeding, et sont représentées sur la figure 1.2.3 sur deux exemples. La surface apparaît donc comme étant beaucoup plus facile à optimiser dans le cas où l'on utilise les champs produits par l'algorithme.

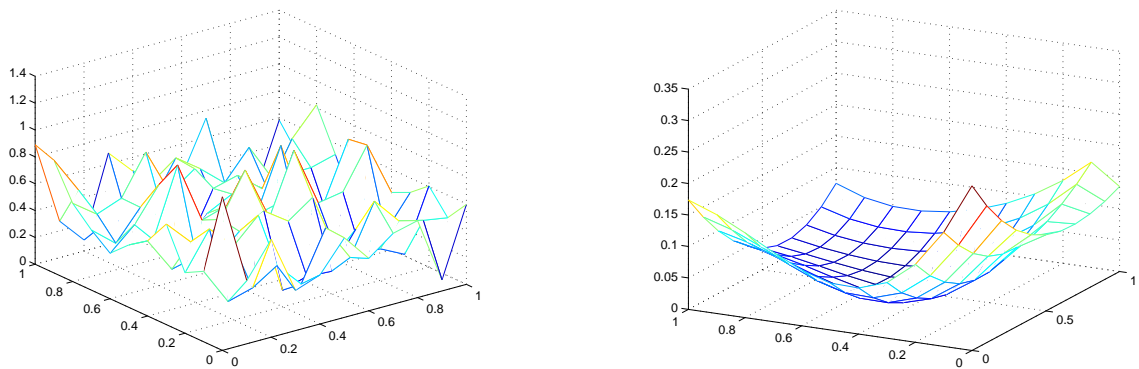


FIGURE 1.3 – Surface des moindres carrés associée au système (1.24) autour de la solution ( $x_0 = .5$ ,  $y_0 = .5$ ), avec des champs de type  $\cos(\omega t)$  (à gauche) et avec les champs obtenus par l'algorithme 2 (à droite).



## Chapitre 2

# Quelques méthodes numériques pour le transport optimal

**Résumé** : on propose ici trois algorithmes dédiés à la résolution de problèmes de transport. Les deux premiers, qui concernent la dimensions 1 ont pour objectif de combler le vide laissé par l'absence de solutions explicites dans le cas du transport en coût convexe sur le cercle et dans le cas du transport en coût concave sur la droite. Dans le premier cas, l'algorithme permet de trouver un point de section du cercle permettant de se ramener à la situation rencontrée sur  $\mathbb{R}$ . Dans le cas du transport en coût concave, on construit une classe d'indicateurs permettant de détecter des appariements entre points consécutifs. Le troisième algorithme concerne un problème de transport simplifié, où l'état final n'est pas fixé précisément. Il donne lieu à une résolution numérique très rapide. Dans un dernier travail, il est appliqué à un problème de jeu à champ moyen.

### Introduction

Cette partie de mon travail concerne la conception d'algorithmes efficaces pour le transport optimal. Le but est ici de calculer rapidement des plans de transport optimaux entre deux mesures, c'est-à-dire des mesures  $\gamma$  solutions du problème de Monge-Kantorovich :

$$\min \left\{ \int_{\Omega \times \Omega} c(x, y) \gamma(dx, dy), \gamma \in \Pi(\mu, \nu) \right\},$$

où  $\Omega$  est le domaine considéré,  $c(x, y)$  est le coût de transport entre deux points  $x$  et  $y$  et  $\Pi(\mu, \nu)$  est l'ensemble des plans de mesures marginales fixées  $\mu$  et  $\nu$ . On se référera à [28] pour une revue des résultats mathématiques concernant ce type de problème.

Trois types de méthodes sont généralement considérées pour traiter ce problème. Le premier type consiste en une approche combinatoire, où l'on aborde la question sous l'angle d'un appariement, et donc de la programmation linéaire. C'est une démarche souvent suivie en logistique et dans de nombreuses applications liées aux mathématiques de la décision. Une revue de l'ensemble des algorithmes correspondant est fournie par [7]. Une deuxième classe de méthodes utilise des formulations du problème sous forme de système d'équations aux dérivées partielles traduisant l'optimalité du déplacement d'une densité de masse. Cette approche fut développée par exemple par Brenier et Benamou [3], ou encore par Loeper et Rapetti [19]. La dernière approche est géométrique : les masses à déplacer sont discrétisées sous forme de polyèdres dont on simule l'évolution à l'aide d'équations différentielles ordinaires. Cette approche fut par exemple développée

par Cullen et Purser dans le cadre de l'étude des équations semi-géostrophiques [9].

Les algorithmes présentés dans cette partie relèvent plutôt des deux premières approches. La première partie de ce chapitre est consacrée à deux algorithmes pour la dimension 1, l'un concernant le calcul de plans de transport en coût convexe sur le cercle, l'autre concernant le problème du transport en coût concave. Ici, le travail se situe plutôt dans le domaine de la combinatoire et de l'optimisation discrète. La deuxième partie de ce chapitre traite au contraire d'un algorithme obtenu par une approche utilisant une formulation par EDP d'un problème de transport.

## 2.1 Algorithmes pour la dimension 1

Les résultats décrits dans cette partie ont été obtenus en collaboration avec Julie Delon et Andreï Sobolevskiï. Sur la droite réelle, il est connu que le ré-arrangement monotone est la solution explicite des problèmes de transport optimal associés à des coûts de transport convexes par rapport à la distance. Deux problèmes restaient cependant ouverts dans ce cadre simple de la dimension 1 : le cas du cercle en coût convexe d'une part et la cas des coûts concaves (sur la droite réelle et sur le cercle) d'autre part. Ces deux problèmes font l'objet de cette section.

### 2.1.1 Coût de transport convexe sur le cercle

*Cette section décrit les résultats obtenus dans [M].*

On rencontre parfois des problèmes de transport optimal dans le domaine du traitement d'image, où il peut être utile d'apparier deux histogrammes de mesures angulaires. N'ayant dans ce contexte pas de solution explicite, même en considérant des fonctions de coût convexes, une méthode consiste à sectionner le cercle en un certain nombre de points et à calculer les coûts résultants du ré-arrangement monotone sur les segments obtenus. Dans un premier travail, on a démontré que cette stratégie était fondée, dans le sens où il existe effectivement dans le cas des fonctions de coûts convexes un point de section pour lequel le plan de transport optimal sur le cercle et sur le segment résultant de la section coïncident.

Décrivons de manière plus précise la démarche suivie. Étant donné  $\hat{\mu}_0, \hat{\mu}_1$  deux mesures sur le cercle  $\mathbb{T} \times \mathbb{T}$ , on définit un coût de transport  $\hat{c}(\cdot, \cdot)$  sur  $\mathbb{T} \times \mathbb{T}$  tel que  $\hat{c}(\hat{x}, \hat{y}) = \inf c(x, y)$ , où  $c(\cdot, \cdot)$  est un coût défini sur  $\mathbb{R} \times \mathbb{R}$  vérifiant  $c(x + 1, y + 1) = c(x, y)$ . L'infimum est défini sur l'ensemble des  $x, y$  choisis respectivement dans les classes de  $\hat{x}$  et de  $\hat{y}$ . On relève ensuite les mesures  $\hat{\mu}_0$  et  $\hat{\mu}_1$  to  $\mathbb{R}$ , pour obtenir des mesures  $\mu_0, \mu_1$  périodiques et localement finies. Ceci fait, on remplace le problème de transport optimal sur le cercle par celui de la minimisation de l'intégrale

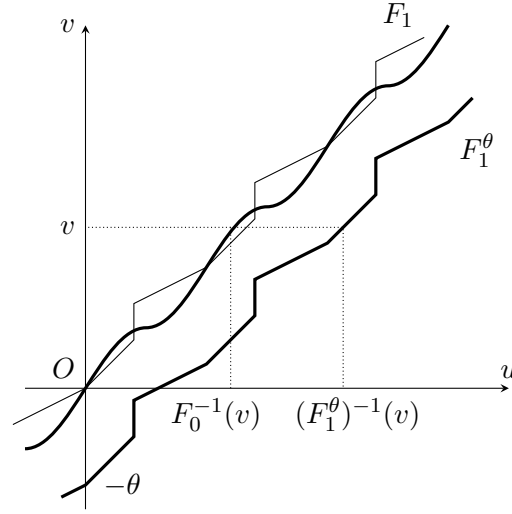
$$I(\gamma) = \iint_{\mathbb{R} \times \mathbb{R}} c(x, y) \gamma(dx \times dy).$$

Dans cette formule, on garde la notation  $\gamma$  pour la mesure associée au transport de  $\mu_0$  sur  $\mu_1$  sur  $\mathbb{R} \times \mathbb{R}$ . Bien sûr, cette intégrale est infinie mais on peut tout de même définir les plans de transport *localement optimaux* de ce problème comme étant ceux vérifiant  $I(\gamma + \delta) - I(\gamma) \geq 0$ , où  $\delta$  est une mesure signée à support compact, de masse nulle et de variation totale finie.

On suppose de plus que la fonction  $c(x, y)$  satisfait la condition dite *de Monge* :

$$c(x_1, y_1) + c(x_2, y_2) < c(x_1, y_2) + c(x_2, y_1)$$

pour tout  $x_1 < x_2$  et  $y_1 < y_2$ . Cette propriété, légèrement plus générale que celle de convexité


 FIGURE 2.1 – Construction d'un plan de transport localement optimal  $\gamma_\theta$ .

du coût<sup>1</sup>, implique que si l'ordre de deux éléments est inversé par le transport, le coût peut être réduit en échangeant les deux appariements. Elle est donc à la base du ré-arrangement monotone et il s'en suit que les plans de transport localement optimaux sont ceux qui préservent l'ordre spatial.

On note alors  $F_0$  et  $F_1$  les fonctions cumulatives de  $\mu_0$  et  $\mu_1$  s'annulant en 0, que l'on considère comme étant continues, quitte à les rendre multivaluées dans le cas où  $\mu_0$ , et/ou  $\mu_1$  contiendraient des atomes. Leurs graphes respectifs permettent de définir des fonctions réciproques  $F_0^{-1}$  et  $F_1^{-1}$ , qui induisent une correspondance entre la mesure de Lebesgue et  $\mu_0$  et  $\mu_1$  respectivement. Soit maintenant  $F_1^\theta(u) = F_1(u) - \theta$ . La fonction  $(F_1^\theta)^{-1}$  est la composée d'une translation de  $\theta$  suivant l'axe des ordonnées et  $F_1^{-1}$ . Un plan de transport  $\gamma_\theta$  qui envoie un élément de masse  $F_0^{-1}(v)$  sur  $(F_1^\theta)^{-1}(v)$  couple de manière monotone  $\mu_0$  et  $\mu_1$ . On peut en déduire qu'il est donc localement optimal. Réciproquement, on prouve que tout plan de transport localement optimal peut en fait être construit de cette manière, en choisissant correctement le paramètre  $\theta$ . La figure 2.1 illustre cette construction. Enfin, on définit le coût moyen  $C_{[F_0, F_1]}(\theta)$  d'un plan  $\gamma_\theta$  sur une période par :

$$C_{[F_0, F_1]}(\theta) = \int_0^1 c(F_0^{-1}(v), (F_1^\theta)^{-1}(v)) dv.$$

On montre alors que la condition de Monge implique la convexité de  $C_{[F_0, F_1]}(\theta)$  et que le minimum global de cette fonction coïncide avec la valeur minimale du coût de transport sur le cercle.

Ce résultat est complété par la construction d'un algorithme dans le cas où  $\mu_0, \mu_1$  sont purement atomiques, c'est-à-dire dans le cas discret. Dans ce cadre, la fonction  $C$  est affine par morceaux, et on peut calculer son minimum à l'aide d'une recherche par dichotomie. Étant donné une borne  $L$  des valeurs de  $C'_{[F_0, F_1]}$ , l'algorithme est le suivant :

**Algorithme 3.** Choisir deux valeurs initiales  $\underline{\theta} \leq 0$  et  $1 \leq \bar{\theta}$ , ainsi qu'une valeur de tolérance  $\varepsilon$ .

1. Poser  $\theta := \frac{1}{2}(\underline{\theta} + \bar{\theta})$ .
2. Calculer  $C'_{[F_0, F_1]}(\theta - 0)$ ,  $C'_{[F_0, F_1]}(\theta + 0)$ .
3. Si  $C'_{[F_0, F_1]}(\theta - 0) \leq 0 \leq C'_{[F_0, F_1]}(\theta + 0)$ , alors  $\theta$  est le minimum requis. Arrêter l'algorithme.

1. Elle est en effet vérifiée pour les coûts de la forme  $c(x, y) = g(|y - x|)$ , avec  $g$  convexe.

4. Si  $\bar{\theta} - \underline{\theta} < \varepsilon/L$  :

(a) calculer  $C_{[F_0, F_1]}(\underline{\theta})$ ,  $C_{[F_0, F_1]}(\bar{\theta})$  ;

(b) redéfinir  $\theta$  comme la solution de

$$C_{[F_0, F_1]}(\underline{\theta}) + C'_{[F_0, F_1]}(\underline{\theta} + 0)(\theta - \underline{\theta}) = C_{[F_0, F_1]}(\bar{\theta}) + C'_{[F_0, F_1]}(\bar{\theta} - 0)(\theta - \bar{\theta}); \quad (2.1)$$

(c) stop.

5. Sinon, poser  $\underline{\theta} := \theta$  si  $C'_{[F_0, F_1]}(\theta + 0) < 0$ , ou  $\bar{\theta} := \theta$  si  $C'_{[F_0, F_1]}(\theta - 0) > 0$ .

6. Aller à l'étape (1).

La borne  $L$  peut être obtenue à l'aide de formules explicites, je renvoie à l'article pour plus de détails à ce propos.

La complexité de cet algorithme est  $O((n_0 + n_1) \log(1/\varepsilon))$ , où  $\varepsilon$  est la précision avec laquelle on souhaite obtenir la valeur du coût de transport optimal. Dans le cas où les masses considérées sont rationnelles, on obtient la solution exacte en  $O((n_0 + n_1) \log M)$  calculs, où  $M$  est le PGCD des masses.

### 2.1.2 Indicateurs d'appariement locaux pour les coût concaves

Cette section décrit les résultats obtenus dans [P, Cras. b, Proc. g].

Si les coûts de transport convexes par rapport à la distance donnent lieu à des solutions facilement calculables via les ré-arrangements monotones, les fonctions de coût concaves s'avèrent plus adaptées à de nombreux modèles économiques. Il n'existe hélas dans ce cas pas de solution explicite au problème de transport optimal. Pour combler cette lacune, on va construire une classe de d'indicateurs permettant de détecter des appariements entre points consécutifs. Commençons par présenter quelques propriétés propres au transport en coût concave.

#### Règle de non-croisement et chaînes

Un exemple de différence entre le cas concave et le cas convexe est représenté sur la figure 2.2. On constate que dans le cas où la fonction de coût est concave, les trajectoires sont soit disjointes,



FIGURE 2.2 – Solutions associées à un coût concave (à gauche) et à un coût convexe (à droite). Les sources sont représentées par des points et les puits par des croix.

soit incluses les unes dans les autres. C'est en fait une règle générale, habituellement appelée règle de *non-croisement*.

Cette propriété implique une autre, que l'on appelle parfois propriété *d'équilibre local* : entre un puits et une source appariés dans un plan de transport optimal, il y a autant de sources que de puits. Dans ce cadre, on peut construire des *chaînes*, c'est-à-dire des ensembles alternés de sources et de puits dont deux points consécutifs vérifie l'équilibre local. On voit facilement que ces ensembles sont stables par le plan transport optimal. Un exemple de partition en chaînes est

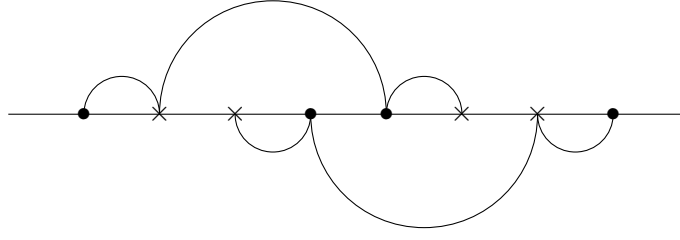


FIGURE 2.3 – Exemple de partition d'un problème en deux chaînes.

montré sur la figure 2.3. Ce raisonnement permet de partitionner un problème et de le restreindre aux problèmes associés à ses chaînes. Mon travail dans ce cadre a débuté par une remarque simple : étant donné une chaîne, la règle de non-croisement implique que celle-ci contient au moins deux points consécutifs appariés dans le plan de transport optimal. L'idée de départ était donc de détecter de tels couples de points, pour retirer itérativement deux par deux tous les points du problème considéré.

### Indicateurs d'appariements locaux

On peut maintenant présenter une classe d'indicateurs d'appariements locaux permettant de détecter des points consécutifs appariés dans la solution optimale. Étant donné un entier  $N$ , on considère deux ensembles de points de  $\mathbf{R}$ ,  $P = (p_i)_{i=1,\dots,N}$  et  $Q = (q_i)_{i=1,\dots,N}$ , représentant respectivement des sources et des puits tels que :

$$p_1 < q_1 < \dots < p_i < q_i < p_{i+1} < q_{i+1} < \dots < p_N < q_N.$$

On considère donc

$$C(\sigma) = \sum_{i,j} c(p_i, q_{\sigma(i)}),$$

où  $\sigma$  est une permutation de  $\{1, \dots, N\}$  et on s'intéresse au problème :

$$\min_{\sigma} C(\sigma).$$

Les indicateurs sont alors définis comme suit.

**Définition 1.** (*Indicateurs d'appariements locaux d'ordre  $k$* )

On pose :

$$I_k^p(i) = c(p_i, q_{i+k}) + \sum_{\ell=0}^{k-1} c(p_{i+\ell+1}, q_{i+\ell}) - \sum_{\ell=0}^k c(p_{i+\ell}, q_{i+\ell}),$$

où  $k$  et  $i$  sont tels que  $1 \leq k \leq N-1$  et  $1 \leq i \leq N-k$ , et

$$I_k^q(i) = c(p_{i+k+1}, q_i) + \sum_{\ell=1}^k c(p_{i+\ell}, q_{i+\ell}) - \sum_{\ell=0}^k c(p_{i+\ell+1}, q_{i+\ell}),$$

où  $k$  et  $i$  sont tels que  $1 \leq k \leq N-2$  and  $1 \leq i \leq N-k-1$ .

La figure 2.4 représente schématiquement un indicateur d'ordre 2. L'intérêt de ces fonctions réside dans le résultat suivant :



FIGURE 2.4 – Représentation schématique d'un indicateur (ici d'ordre 2).

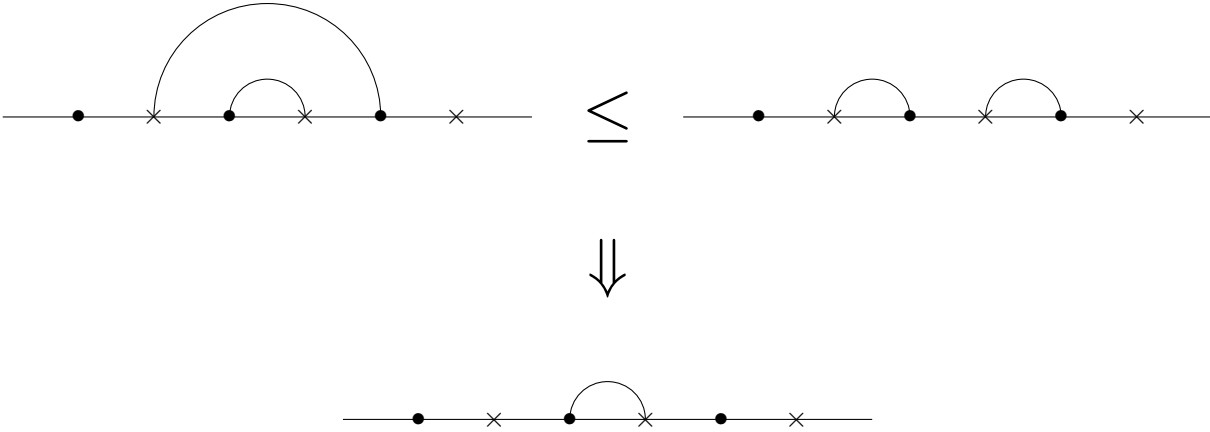


FIGURE 2.5 – Représentation schématique du théorème 5 dans le cas où  $k = 1$ .

**Théorème 5.** Soit  $k_0 \in \mathbf{N}$  tel que  $1 \leq k_0 \leq N - 1$  et  $i_0 \in \mathbf{N}$  (resp.  $i'_0 \in \mathbf{N}$ ), tel que  $1 \leq i_0 \leq N - k_0$  (resp.  $1 \leq i'_0 \leq N - k_0 - 1$ ).

Supposons que

1.  $I_k^p(i) \geq 0$  pour  $k = 1, \dots, k_0 - 1$ ,  $1 \leq i \leq N - k$ ,
2.  $I_k^q(i') \geq 0$  pour  $k = 1, \dots, k_0 - 1$ ,  $1 \leq i' \leq N - k - 1$ ,
3.  $I_{k_0}^p(i_0) < 0$  (resp.  $I_{k_0}^q(i'_0) < 0$ ).

Alors, la permutation  $\sigma^*$  associée au plan de transport optimal satisfait  $\sigma^*(i) = i - 1$  pour  $i = i_0 + 1, \dots, i_0 + k_0$  (resp.  $\sigma^*(i) = i$  pour  $i = i_0 + 1, \dots, i_0 + k_0$ ).

Il est donc possible, en utilisant les indicateurs de la définition 1, de trouver des appariements du plan optimal par un calcul uniquement local. La figure 2.5 illustre le résultat dans le cas  $k = 1$ .

### Algorithme

L'utilisation hiérarchique de ces indicateurs permet ainsi de construire en un nombre fini d'itérations le plan de transport optimal. L'algorithme précis qui découle de cette idée est le suivant :

**Algorithme 4.** Soit  $\mathcal{P} = \{p_1, \dots, p_N, q_1, \dots, q_N\}$ ,  $\ell^p = (1, \dots, N)$ ,  $\ell^q = (1, \dots, N)$ , et  $k = 1$ . Tant que  $\mathcal{P} \neq \emptyset$  et  $k \leq N - 1$  faire :

1. Calculer  $I_k^p(i)$  et  $I_k^q(i')$  pour  $i = 1, \dots, N - k$  et  $i' = 1, \dots, N - k - 1$ .

2. Définir

$$\mathcal{I}_k^p = \{i_0, 1 \leq i_0 \leq N - k, I_k^p(i_0) < 0\},$$

$$\mathcal{I}_k^q = \{i_0, 1 \leq i_0 \leq N - k - 1, I_k^q(i_0) < 0\},$$

et faire :

(a) si  $\mathcal{I}_k^p = \emptyset$  et  $\mathcal{I}_k^q = \emptyset$ , incrémenter  $k = k + 1$  ;

(b) sinon, faire :

- pour tout  $i_0$  dans  $\mathcal{I}_k^p$  et pour  $i = i_0 + 1, \dots, i_0 + k$ , faire :
  - Définir  $\sigma^*(\ell_i^p) = \ell_{i-1}^q$ ,
  - Retirer  $\{p_{\ell_i^p}, q_{\ell_{i-1}^q}\}$  de  $\mathcal{P}$ ,
  - Retirer  $\ell_i^p$  et  $\ell_i^q$  de  $\ell^p$  et  $\ell^q$  respectivement.
- Pour tout  $i'_0$  dans  $\mathcal{I}_k^q$  et pour  $i = i'_0 + 1, \dots, i'_0 + k$ , faire :
  - Définir  $\sigma^*(\ell_i^q) = \ell_i^p$ ,
  - Retirer  $\{p_i, q_i\}$  de  $\mathcal{P}$ ,
  - Retirer  $\ell_i^p$  et  $\ell_i^q$  de  $\ell^p$  et  $\ell^q$  respectivement.
- Poser  $N = \frac{1}{2}\text{Card}(\mathcal{P})$ , et renommer les points de  $\mathcal{P}$  de telle sorte que

$$\mathcal{P} = \{p_1, \dots, p_N, q_1, \dots, q_N\},$$

$$p_1 < q_1 < \dots < p_i < q_i < p_{i+1} < q_{i+1} < \dots < p_N < q_N.$$

- Ré-indexer en conséquence les entiers de  $\ell^p$  et de  $\ell^q$ .
- Poser  $k = 1$ .

Si  $k = N - 1$ , pour  $i = 1, \dots, N$  poser  $\sigma^*(\ell_i^p) = \ell_i^q$ .

On peut montrer que cet algorithme a une complexité comprise entre  $\mathcal{O}(N)$  et  $\mathcal{O}(N^2)$ .

**Théorème 6.** Soit  $C^+(N)$  le nombre d'additions requis pour calculer un plan de transport optimal entre  $N$  couples par l'algorithme 4. On a la borne :

$$C^+(N) \leq 3N^2 - 6N.$$

Le pire des cas est celui où l'algorithme ne trouve que des indicateurs positifs et le cas le plus simple est celui où tout les points sont appariés grâce à des indicateurs négatifs d'ordre 1.

### Adaptations et extensions

Le résultat obtenu au théorème 5 et l'algorithme 4 peuvent être étendus et appliqués à des situations plus générales. On a ainsi adapté les indicateurs au cas du cercle, ainsi que sur des problèmes où le nombre de sources diffère du nombre de puits. L'adaptation au cercle est relativement simple : il suffit de ne pas autoriser l'utilisation d'indicateurs tels que le segment considéré lors du calcul ne dépasse pas la demi circonférence. L'adaptation au cas déséquilibré consiste en fait à montrer que la démarche suivie reste valable, depuis la construction des chaînes jusqu'au théorème 5. Un certain nombre de résultats techniques supplémentaires permettent d'obtenir ce résultat.

Enfin, on a construit une méthode pour traiter le cas où les masses ne sont plus unitaires, mais entières : dans cette situation, le plan de transport optimal peut être obtenu en répartissant chaque masse entière  $m$  considérée en  $m$  masses unitaires réparties dans un intervalle de taille  $\varepsilon$  autour du point où est situé  $m$ . On montre alors (facilement) que pour  $\varepsilon$  suffisamment petit, le plan de transport obtenu est optimal.

## 2.2 Algorithme pour les dimensions supérieures, application aux jeux à champ moyen

En dimension supérieure et dans le cas de coûts de transport convexes, plusieurs algorithmes ont été proposés. Un algorithme de type combinatoire bien adapté aux problèmes discrets souvent utilisé est l'algorithme de Bertsekas [4]. Plusieurs algorithmes basés sur des formulations EDP ont également été proposés. L'algorithme de Brenier et Benamou [3] en est un représentant désormais classique. Citons également l'algorithme de Loeper et Rapetti [19], basé sur une méthode de Newton et plus récemment l'algorithme de Haber, Rehman et Tannenbaum basé sur une formulation variationnelle du problème et une discrétisation assurant la conservation de la masse [14].

Ces algorithmes ne s'appliquent qu'au cas convexe et aux situations purement hyperboliques, i.e. sans diffusion. Pour palier ce problème et proposer une méthode plus rapide, j'ai construit un algorithme reprenant les idées des schémas monotones présentés au chapitre 1 de la partie I. Le problème considéré est celui d'un déplacement de foule en présence de bruit, dont le mouvement est gouverné par une équation de type Fokker-Planck. Une simplification a été introduite puisqu'on ne prescrit plus l'état final, mais on cherche à minimiser un potentiel. La convergence numérique de ce schéma s'avère extrêmement rapide.

### 2.2.1 Algorithme monotone adapté

*Cette section correspond à l'acte de conférence [Proc. e].*

Le problème considéré ici est celui de la minimisation de la fonctionnelle définie par :

$$E(v) = \frac{1}{2} \int_0^T \int_{\Omega} \rho(x, t) v^2(x, t) dx dt + \int_{\Omega} V(x) \rho(x, T) dx,$$

où  $\Omega \subset \mathbb{R}^N$  est borné,  $T$  le temps de contrôle,  $\rho$  est la variable d'état, contrôlée par le champ de vitesse  $v$  suivant l'équation de Fokker-Planck :

$$\begin{aligned} \partial_t \rho(x, t) - \varepsilon^2 \Delta \rho(x, t) + \operatorname{div}(v(x, t) \rho(x, t)) &= 0, \\ \rho(x, 0) &= \rho_0(x), \end{aligned} \tag{2.2}$$

où  $\rho_0$  est l'état initial et  $\varepsilon$  est un paramètre de diffusion. La fonction  $V$  représente un potentiel. Ce problème peut être vu comme une version simplifiée d'un problème de transport optimal, puisque l'état final n'est pas prescrit. Il modélise un déplacement de foule composée d'agents rationnels, cherchant à minimiser le coût de leur déplacement, tout en ayant un objectif à optimiser au temps  $T$ . Le modèle prend aussi en compte le bruit dans le déplacement, en faisant intervenir un terme lié à la diffusion correspondant à une partie brownienne dans le mouvement des agents. Cet aspect n'est généralement pas pris en compte dans les problèmes de transport optimal.

### Discrétisation

Une partie importante de ce travail concerne la discrétisation, qui fut le point de blocage qui résista le plus longtemps lors du développement de la méthode.

Soit  $M, N$  deux entiers et un réel positif  $L$ . On se place pour simplifier sur un intervalle  $[0, L]$ , mais tout ce qui suit s'adapte sans problème aux dimensions supérieures et aux domaines adaptés aux schémas aux différences finies. Soit le pas de temps  $\Delta t = \frac{1}{N}$  et le pas d'espace  $\Delta x = \frac{L}{M}$ .

Pour  $j = 0, \dots, M$ ,  $i = 0, \dots, N$ , on note  $\rho_j^i$  l'approximation numérique de  $\rho(i.\Delta t, j.\Delta x)$ . La vitesse  $v_{j+1/2}^i$  est définie aux points  $(i.\Delta t, (j+1/2).\Delta x)$  et est notée  $v_{j+1/2}^i$ . On considère des conditions de Neumann pour  $\rho$  et on impose  $v_{1/2}^i = v_{M-1/2}^i = 0$  pour tout  $i = 0 \dots N-1$ , de sorte que le schéma aux différences finies

$$\begin{aligned} \rho_j^{i+1} &= \rho_j^i + \varepsilon^2 \frac{\Delta t}{\Delta x^2} (\rho_{j+1}^i - 2\rho_j^i + \rho_{j-1}^i) \\ &\quad - \frac{\Delta t}{\Delta x} (\rho_{j+1/2}^i v_{j+1/2}^i - \rho_{j-1/2}^i v_{j-1/2}^i). \end{aligned} \quad (2.3)$$

où

$$\rho_{j+1/2}^i = \begin{cases} \rho_{j+1}^i & \text{if } v_{j+1/2}^i < 0 \\ \rho_j^i & \text{if } v_{j+1/2}^i \geq 0, \end{cases} \quad (2.4)$$

présERVE la masse numériquement. Comme le montre ces équations, on utilise un schéma aux différences finies comme méthode numérique pour résoudre (2.2). Si dans cette approche la partie parabolique est discrétisée par une approximation centrée pour le Laplacien, on utilise un décentrage (parfois appelé *up-winding*) pour la partie hyperbolique, comme le reflète (2.4). Cette technique garantit la positivité de  $\rho$  au niveau discret, sous réserve que la condition CFL

$$\forall j = 1 \dots M-1, |v_{j+1/2}^i| \leq \lambda := \frac{\Delta x}{2\Delta t} - \varepsilon^2 \frac{1}{\Delta x}. \quad (2.5)$$

soit vérifiée. Ceci constitue en fait une propriété cruciale puisque ce schéma va être couplé avec une procédure de minimisation, qui pourrait tirer parti d'erreurs numériques pour accentuer les valeurs négatives éventuelles de  $\rho$  et faire tendre  $E(v)$  vers  $-\infty$ .

Pour simplifier les notations, on écrit (2.3) sous la forme matricielle

$$\rho^{i+1} = (A + B(v^i))\rho^i, \quad (2.6)$$

où  $A$  correspond à la première ligne de (2.3) et  $B$  à la deuxième, la partie hyperbolique de l'équation de Fokker-Planck (2.2). On garde également les notations  $v$  et  $V$  pour désigner les vecteurs  $(v_{j+1/2}^i)_{i,j}$  et  $V_j = V(j.\Delta x)$  respectivement.

La fonctionnelle à optimiser est discrétisée de la manière suivante.

$$\begin{aligned} E_{\Delta t, \Delta x}(v) &:= \Delta t \cdot \Delta x \sum_{i=0}^{N-1} \sum_{j=1}^{M-1} \frac{1}{2} q_j(v^i) \rho_j^i + \Delta x \sum_{j=1}^{M-1} V_j \rho_j^N \\ &= \Delta t \sum_{i=0}^{N-1} \frac{1}{2} \langle \rho^i, q(v^i) \rangle + \langle \rho^N, V \rangle \end{aligned}$$

où  $\langle \cdot, \cdot \rangle$  est le produit scalaire sur  $\mathbb{R}^{M-1}$  défini par :

$$\langle u, v \rangle = \Delta x \sum_{j=1}^{M-1} u_j v_j.$$

Le vecteur  $q(v^i) = (q_j(v^i))_{j=1 \dots M-1}$  est défini par :

$$q_j(v^i) = \frac{(v_{j-1/2}^i)^2 + (v_{j+1/2}^i)^2}{2}.$$

### Procédure d'optimisation

On reprend maintenant la démarche suivie habituellement dans la conception d'un schéma monotone. Soit donc deux contrôles  $v$  et  $v'$  ainsi que les états  $(\rho^i)_{i=0\dots N}$  et  $(\rho'_i)_{i=0\dots N}$  associés, solutions de (2.6). Soit également  $\phi = (\phi_i)_{i=0\dots N}$  l'état adjoint correspondant à  $v$  défini par l'itération rétrograde :

$$\begin{aligned}\phi^N &= V, \\ \phi^i &= (A^* + B^*(v^i))\phi^{i+1} + \frac{\Delta t}{2}q(v^i).\end{aligned}\tag{2.7}$$

La démarche présentée au chapitre 1 de la partie I débouche sur

$$E_{\Delta t, \Delta x}(v') - E_{\Delta t, \Delta x}(v) = \Delta t \cdot \Delta x \sum_{i=0}^{N-1} \sum_{j=1}^{M-2} \Delta_j^i(v', v),$$

où

$$\begin{aligned}\Delta_j^i(v', v) &= \frac{\rho_j^i + \rho_{j+1}^i}{2} \left( \frac{(v_{j+1/2}^i)^2 - (v_{j+1/2}^i)^2}{2} \right) \\ &+ \left( \rho_{j+1/2}^i v_{j+1/2}^i - \tilde{\rho}_{j+1/2}^i v_{j+1/2}^i \right) \left( \frac{\phi_{j+1}^{i+1} - \phi_j^{i+1}}{\Delta x} \right).\end{aligned}\tag{2.8}$$

Ici, on a introduit :

$$\tilde{\rho}_{j+1/2}^i = \begin{cases} \rho_{j+1}^i & \text{if } v_{j+1/2}^i < 0 \\ \rho_j^i & \text{if } v_{j+1/2}^i \geq 0. \end{cases}$$

Si  $v$  est fixé, cette formule est quasi-explicite par rapport à  $v_j^i$  : le seul aspect implicite concerne  $\rho_{j+1/2}^i$  dont la valeur dépend du signe de  $v_j^i$ . La fonction  $v_j^i \mapsto \Delta_j^i(v', v)$  est donc une fonction continue, polynomiale de degré 2 par morceaux.

On se donne maintenant un réel  $\theta$ , et on définit  $v_j^i$  comme la solution de

$$\Delta_j^i(v', v) = -\theta \frac{\rho_j^i + \rho_{j+1}^i}{2} (v_{j+1/2}^i - v_{j+1/2}^i)^2.\tag{2.9}$$

D'après les remarques précédentes, cette équation a entre une et quatre solutions, incluant la solution évidente  $v_j^i = v_j^i$ . On choisit de définir  $v_j^i$  comme la racine de (2.9) la plus proche de  $v_j^i$  lorsqu'il y a plus d'une solution et par  $v_j^i = v_j^i$  dans le cas contraire. La monotonie de l'algorithme est de toute façon garantie. Une modification doit cependant être faite pour que  $v'$  vérifie la CFL (2.5), mais on peut facilement faire en sorte que celle-ci soit compatible avec la monotonie.

### Algorithme

Donnons maintenant précisément l'algorithme.

**Algorithme 5.** *Supposons  $v^k$  déjà calculé. Le calcul de  $v^{k+1}$  se fait de la manière suivante.*

- Calculer  $\phi^k$  par (2.7) avec  $v = v^k$ .
- Poser  $\rho^0 = \rho_0$  et calculer itérativement  $\rho^i$  à partir  $\rho^{i-1}$  par :
  - pour tout  $j = 1, \dots, M-1$ , calculer la racine  $v_{j+1/2}^i$  de (2.8) (avec  $\phi = \phi^k$ ,  $v = v^k$ ) la plus proche de  $(v^k)_j^{i-1}$  si elle existe. Si elle n'existe pas, poser  $(v^{k+1})_j^{i-1} = (v^k)_j^{i-1}$ , si elle existe poser  $(v^{k+1})_j^{i-1} = \text{sign}(v_{j+1/2}^i) \min(\lambda, |v_{j+1/2}^i|)$ .

– calculer  $(\rho^{k+1})^i$  par (2.3) avec  $v^{i-1} = (v^{k+1})^i$ .

Un critère d'arrêt possible consiste en la vérification des conditions d'optimalité discrètes. Étant donné un seuil  $\text{Tol} > 0$ , ce critère s'écrit alors :

$$\sup_{1 \leq i \leq N-1, 1 \leq j \leq M-1} \left( \left| \frac{(\rho^k)_j^i + (\rho^k)_{j+1}^i}{2} (v^k)_{j+1/2}^i + (\rho^k)_{j+1/2}^i \frac{(\phi^k)_{j+1}^{i+1} - (\phi^k)_j^{i+1}}{\Delta x} \right| \right) \leq \text{Tol}.$$

### Exemples de résultats numériques

Cet algorithme a été appliqué à un exemple bidimensionnel avec  $\Omega = [0, 1] \times [0, 1]$ . On considère la densité initiale :

$$\rho(0, x, y) = e^{-10(x-0.2)^2} + e^{-10(y-0.2)^2},$$

et le potentiel (représenté sur la figure 2.6)

$$V(x, y) = 40(1 + e^{-10(y-0.8)^2} - e^{-10(x-0.8)^2} + e^{-10(y-0.5)^2} - e^{-10(x-0.5)^2}).$$

L'évolution de la densité  $\rho$  au cours du temps est représentée sur la figure 2.7. La convergence

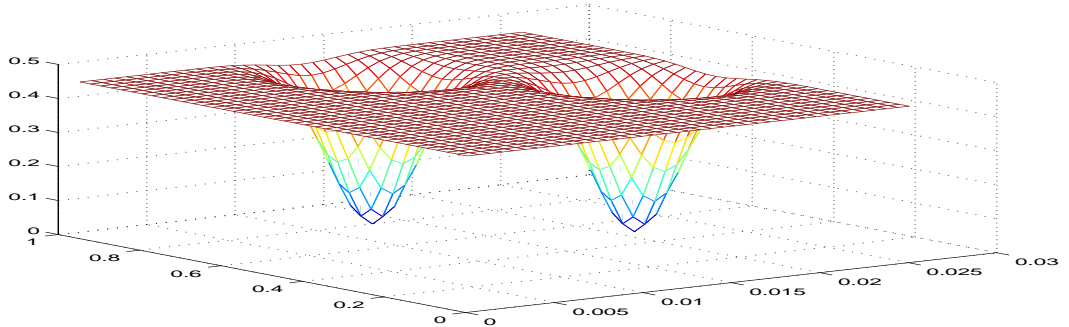


FIGURE 2.6 – Représentation du potentiel  $V(x, y)$ .

numérique est obtenue en une centaine d'itérations.

### 2.2.2 Application à la théorie des jeux à champ moyen

Cette section correspond aux résultats obtenus dans [O].

Dans un second temps, cet algorithme a été adapté de manière à proposer un premier schéma numérique pour résoudre des problèmes issus de la théorie des jeux à champ moyen. Cette théorie, développée par Jean-Michel Lasry et Pierre-Louis Lions donne également lieu à des problèmes d'optimisation auxquels peut être adaptée la méthode générale qui vient d'être décrite. En collaboration avec Gabriel Turinici et Aimé Lachapelle, j'ai ainsi considéré le problème de jeux à champ moyen suivant :

$$J(v) = \int_Q \left( p(t)(1 - \beta z) + \frac{c_0 \cdot z}{c_1 + c_2 \rho(t, z)} + \frac{v^2(t)}{2} \right) \rho(t, z) dz,$$

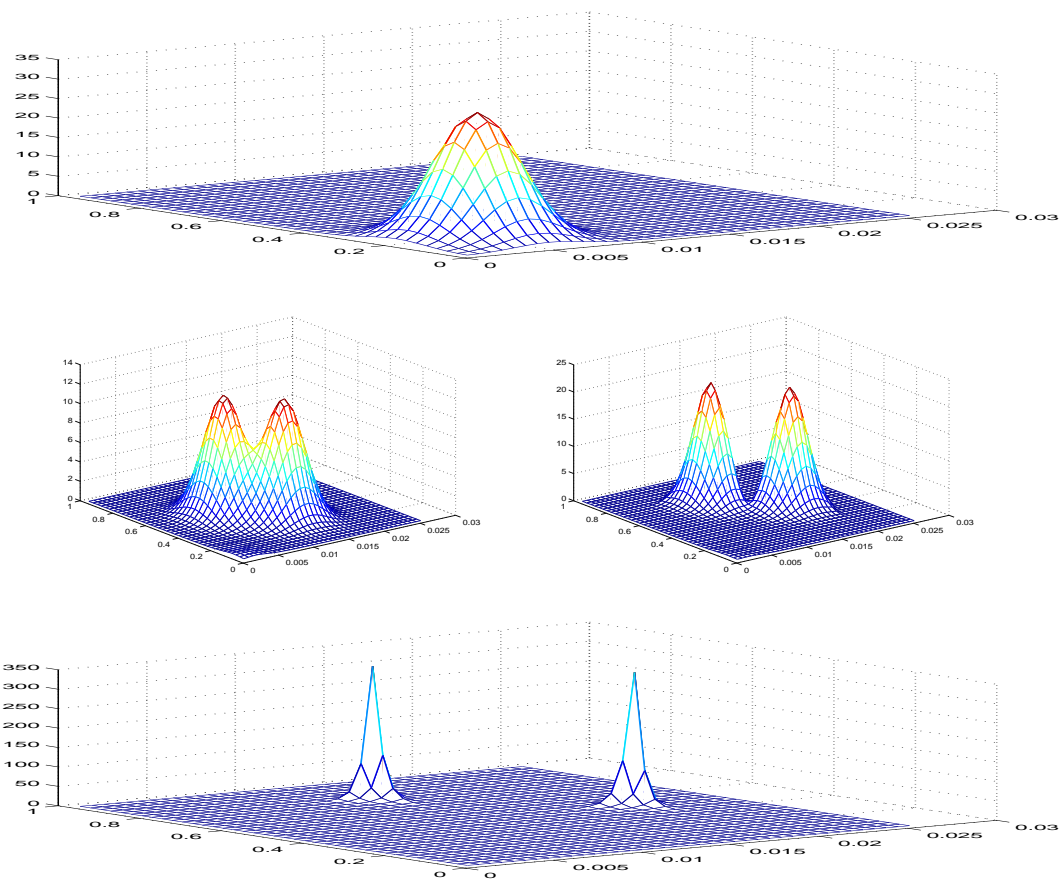


FIGURE 2.7 – Évolution de  $\rho$ . Haut : densité initiale. Milieu : densité aux temps  $t = 0.5$  et  $t = 0.75$ . Bas : densité finale.

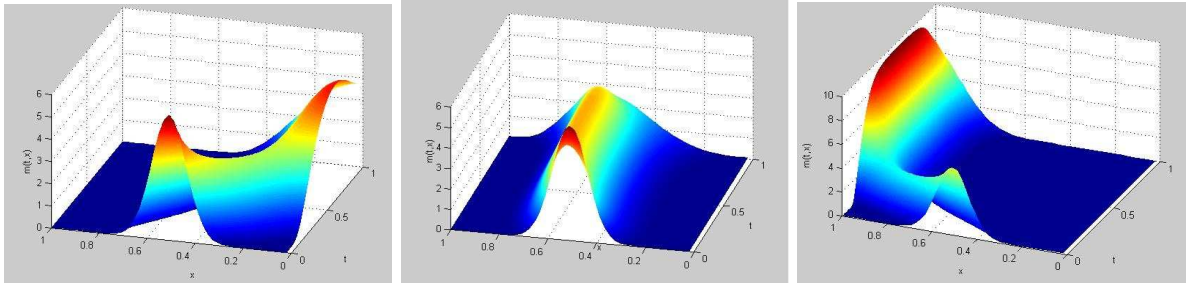


FIGURE 2.8 – Évolution des choix sociaux lorsque le coût de chauffage est nul (figure de gauche), moyen (figure du milieu) et élevé (figure de droite). L’abscisse correspond au niveau d’isolation, l’ordonnée au temps et la cote à la densité.

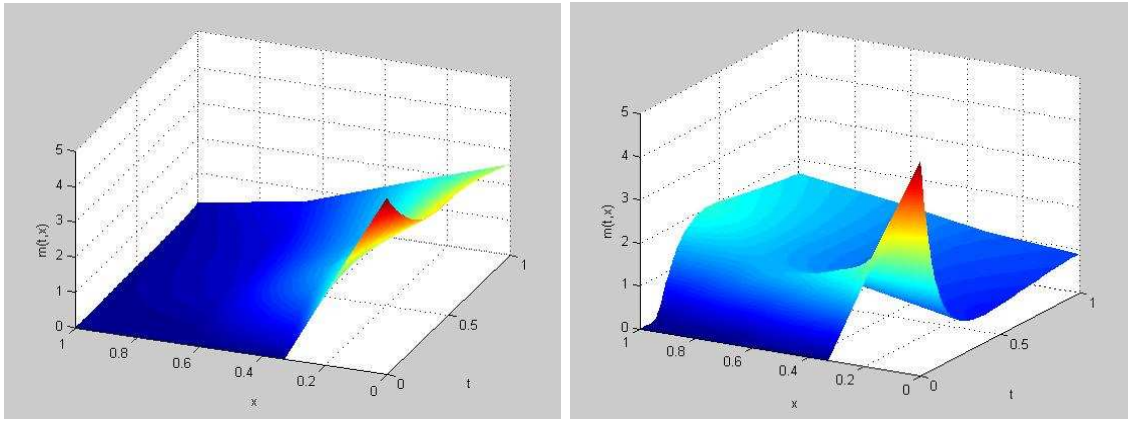


FIGURE 2.9 – Deux équilibre différents sont obtenus pour un même jeu de paramètres.

où  $\beta, c_0, c_1, c_2$  sont des constantes positives, et où  $p(t)$  est une fonction positive. Cette fonctionnelle traduit une situation de choix social simple, où des populations ont à choisir entre un niveau d’isolation  $z$  élevé, ou au contraire un rapprochement les unes des autres. Le premier terme rend ainsi compte du coût de chauffage, le second modélise le bénéfice apporté par une agglomération et le coût d’entretien. Le dernier terme représente le coût de changement d’état. L’équation d’évolution de l’état  $\rho$  est l’équation de Fokker-Planck (2.2). Puisque la fonctionnelle  $J$  est concave par rapport à l’état  $\rho$  et la dynamique est linéaire, on est dans le cadre d’application de la procédure précédente. Je renvoie à l’article [O] pour une description plus complète du modèle et des détails techniques. Je me contente ici de présenter quelques résultats numériques.

La figure 2.8 montre les choix effectués par les joueurs lorsqu’ils font face à trois coûts de chauffage différents. Dans les trois cas, la solution correspond à l’intuition : les joueurs décident soit de s’agglomérer lorsque le coût de chauffage est élevé, soit de rester proche de leur configuration initiale lorsque le coût est moyen, soit de s’isoler lorsque le coût est nul. Dans chacun des cas, l’un des trois termes de la fonctionnelle est prépondérant. L’étude du modèle menée par l’intermédiaire de l’algorithme a également mis en évidence l’existence de plusieurs équilibres pour un jeu de paramètres donné. Ces différents équilibres peuvent être obtenus en changeant le contrôle  $v^0$  utilisé pour initialiser l’algorithme. La figure 2.9 montre deux de ces équilibres. Ces deux situations peuvent s’interpréter en terme de croyance, révélée par la condition initiale : selon qu’un agent pense que les autres vont choisir un niveau d’isolation élevé ou non, il aura tendance à effectuer le même choix. Ce phénomène est communément appelé *herding*.



Deuxième partie

Analyse numérique



# Chapitre 1

## Parallélisation en temps de méthodes de contrôle

**Résumé** : je présente dans ce chapitre deux algorithmes de parallélisation en temps permettant d'accélérer la résolution de problèmes de contrôle optimal. Ces algorithmes reposent sur un découpage de l'intervalle de temps sur lequel le système est contrôlé ainsi que sur un système spécifique de cibles intermédiaires. La méthode développée dans les deux cas permet de transformer le problème de contrôle initial en  $N$  sous-problèmes de contrôle indépendants, pouvant donc être résolus en parallèle. Des théorèmes garantissent alors la convergence de la suite de contrôles obtenus par concaténation vers un point critique de la fonctionnelle initiale, ce qui rend transparente la parallélisation en temps. Ces procédures, testées sur des exemples standards, permettent de gagner au moins un ordre de grandeur dans le coût computationnel du contrôle optimal.

### Introduction

La décomposition de domaine est un thème faisant l'objet de nombreux travaux depuis une vingtaine d'années. Le but est d'accélérer les calculs en les répartissant sur plusieurs processeurs. Si de nombreuses techniques de parallélisation par une décomposition en espace sont maintenant maîtrisées, la question de la décomposition en temps est une thématique qui est l'objet de diverses recherches ces dernières années. Longtemps laissée de côté, elle trouve son origine dans les techniques dites de *multi-shooting* [6] développées initialement pour traiter des problèmes de contrôle d'équations différentielles ordinaires.

Plus tard le découpage en temps fut considéré pour traiter la résolution d'équations différentielles et aux dérivées partielles. Un algorithme appartenant à cette classe est l'algorithme parallèle introduit par Jacques-Louis Lions, Yvon Maday et Gabriel Turinici [18]. Il fut développé et appliqué à de nombreux domaines. Une analyse mathématique de cet algorithme a été présentée dans [12]. Dans toutes ces méthodes, l'intervalle de temps considéré est partitionné et des points intermédiaires sont définis de manière itérative pour pouvoir calculer l'évolution sur chaque sous-intervalle de manière indépendante. Mon travail sur cette problématique a consisté à construire une méthode dédiée spécifiquement à la résolution de systèmes d'optimalité associés à des problèmes de contrôle. Le point commun aux algorithmes qui sont présentés ici est la définition de cibles intermédiaires, rendant cohérents le problème initial et les problèmes parallélisés.

## 1.1 Algorithme pour les équations hyperboliques

Cette section correspond aux résultats obtenus dans [E].

Durant ma thèse et en collaboration avec Yvon Maday et Gabriel Turinici, j'ai construit une méthode de parallélisation en temps pour le contrôle optimal d'équations hyperboliques. L'idée initiale était de coupler le schéma pararéel décrit dans [18] avec un algorithme monotone du type de ceux décrit au chapitre 1 de la partie I. Finalement, même si la stratégie d'optimisation proposée partage avec le schéma pararéel l'idée d'utiliser une trajectoire grossière, les deux algorithmes s'avèrent de nature totalement différente. Les contrôles virtuels sont en particulier mis à jour selon une formule complètement différente.

Dans le cadre considéré, on cherche à atteindre en un temps  $T$  un état cible en partant d'une condition initiale fixée. Pour paralléliser la résolution, on fixe des états intermédiaires jouant tantôt le rôle de condition initiale, tantôt celui de cible selon le sous-intervalle de la décomposition que l'on considère. Des contrôles optimaux partiels peuvent alors être calculés en parallèle puis concaténés pour permettre une mise à jour des états intermédiaires. L'algorithme qui en résulte utilise fortement l'idée de poursuite de trajectoire évoquée plus haut : les états intermédiaires sont en effet construits par interpolation de la trajectoire directe et d'une trajectoire de référence conduisant à l'état cible au temps  $T$ .

Ce qui suit décrit la procédure et ses propriétés dans le cas du contrôle de l'équation de Schrödinger. Pour autant, la démarche et la plupart des résultats restent valables dans le cas plus général des équations linéaires hyperboliques. J'entends ici par *équation linéaire hyperbolique* une équation de la forme :

$$\partial_t y = A(c)y + B(c),$$

où  $y \in \mathcal{H}$  est l'état du système (avec  $\mathcal{H}$  un espace hilbertien) et  $A$  est un opérateur anti-symétrique ou anti-hermitien.

### 1.1.1 Le problème

On considère le problème de contrôle optimal, déjà présenté au chapitre 1 de la partie I associé à la minimisation de la fonctionnelle :

$$\begin{aligned} J(\varepsilon) &= \|\psi(T) - \psi_{cible}\|_2^2 + \alpha \int_0^T \varepsilon(t)^2 dt, \\ &= 2 - \Re\langle \psi(T), \psi_{cible} \rangle + \alpha \int_0^T \varepsilon(t)^2 dt. \end{aligned} \quad (1.1)$$

On rappelle que le contrôle  $\varepsilon$  est le champ électrique délivré par un laser, l'état  $\psi$  est la fonction d'onde à contrôler, en un temps  $T$ . Le paramètre  $\alpha$  permet la pénalisation du contrôle, et  $\psi_{cible}$  est un état cible. L'évolution du système est régie par l'équation de Schrödinger qui lie le contrôle et l'état selon

$$\begin{aligned} i\partial_t \psi^\varepsilon &= (H - \varepsilon\mu)\psi^\varepsilon \\ \psi^\varepsilon(t=0) &= \psi_{init}. \end{aligned} \quad (1.2)$$

La fonction  $\psi_{init}$  représente l'état initial du système. Je renvoie à l'introduction du chapitre 1 de la partie I pour plus de détails sur ce problème de contrôle ainsi que sur les opérateurs  $H$  et  $\mu$ ,

dont on n'utilisera ici seulement la symétrie<sup>2</sup>. Dans la suite, il sera souvent appelé à l'état adjoint  $\chi^\varepsilon$  défini par l'équation :

$$\begin{aligned} i\partial_t \chi^\varepsilon &= (H - \varepsilon\mu)\chi^\varepsilon \\ \chi^\varepsilon(t = T) &= \psi_{cible}. \end{aligned} \quad (1.3)$$

### 1.1.2 Cadre adapté à la parallélisation

Pour  $N \geq 1$ , on introduit maintenant une partition de l'intervalle de contrôle  $[0, T]$

$$[0, T] = \cup_{\ell=0}^{N-1} [T_\ell, T_{\ell+1}],$$

avec  $T_0 = 0$  et  $T_N = T$  ainsi qu'une suite d'états  $\Lambda = (\lambda_\ell)_{\ell=1, \dots, N-1}$ , avec  $\lambda_0 = \psi_{init}$  et  $\lambda_N = \psi_{cible}$ .

Plutôt que d'attaquer directement la résolution du problème initial associé à (1.1), on considère le problème de contrôle consistant en la minimisation de la fonctionnelle

$$J_{\parallel}(\varepsilon, \Lambda) = \sum_{\ell=0}^{N-1} \beta_\ell \|\psi_{\ell, n_{\ell+1}}^{\varepsilon_\ell} - \lambda_{\ell+1}\|_2^2 + \alpha \sum_{\ell=0}^{N-1} \int_{T_\ell}^{T_{\ell+1}} \varepsilon_\ell^2(t) dt,$$

avec  $\beta_\ell = \frac{T}{T_{\ell+1} - T_\ell}$  et où l'état  $\psi_\ell^{\varepsilon_\ell}$  est la solution de

$$\begin{cases} i\partial_t \psi_\ell^{\varepsilon_\ell} &= (H - \varepsilon_\ell \mu) \psi_\ell^{\varepsilon_\ell} \\ \psi_\ell^{\varepsilon_\ell}(t = T_\ell) &= \lambda_\ell, \quad \ell = 0, \dots, N-1, \end{cases}$$

où  $\varepsilon_\ell$  est la restriction de  $\varepsilon$  à  $[T_\ell, T_{\ell+1}]$  (avec  $\ell = 0, \dots, N-1$ ). Par cette équation,  $\psi_\ell^{\varepsilon_\ell}$  dépend donc aussi de  $\lambda_\ell$  ; j'omettrai cette dépendance par simplicité dans la suite. La fonctionnelle  $J_{\parallel}$  peut être décomposée de la manière suivante :

$$J_{\parallel}(\varepsilon, \Lambda) = \sum_{\ell=0}^{N-1} \beta_\ell J_\ell(\varepsilon_\ell, \lambda_\ell, \lambda_{\ell+1}),$$

où  $J_\ell$  sont des fonctionnelles partielles définies par :

$$J_\ell(\varepsilon_\ell, \lambda_\ell, \lambda_{\ell+1}) = \|\psi_\ell^{\varepsilon_\ell}(T_{\ell+1}) - \lambda_{\ell+1}\|_2^2 + \alpha'_\ell \int_{T_\ell}^{T_{\ell+1}} \varepsilon_\ell^2(t) dt,$$

avec :

$$\alpha'_\ell = \frac{\alpha}{\beta_\ell}.$$

Une propriété intéressante de la fonctionnelle  $J_{\parallel}$  est qu'à contrôle  $\varepsilon$  fixé, le minimum par rapport à  $\Lambda$  est calculable explicitement et vérifie certaines propriétés décrites dans le théorème suivant.

**Théorème 7.** *On pose  $\Lambda^\varepsilon = (\lambda_\ell^\varepsilon)_{\ell=1, \dots, N-1}$  avec :*

$$\lambda_\ell^\varepsilon = (1 - \gamma_\ell) \psi^\varepsilon(T_\ell) + \gamma_\ell \chi^\varepsilon(T_\ell),$$

*où l'adjoint  $\chi^\varepsilon$  est défini par (1.3) et où  $\gamma_\ell = \frac{T_{\ell+1} - T_\ell}{T}$ . Alors :*

$$\Lambda^\varepsilon = \operatorname{argmin}_\Lambda (J_{\parallel}(\varepsilon, \Lambda)). \quad (1.4)$$

*De plus :*

$$J_{\parallel}(\varepsilon, \Lambda^\varepsilon) = J(\varepsilon).$$

La trajectoire  $\Lambda^\varepsilon$  est donc construite par interpolation des trajectoires de l'état  $\psi^\varepsilon$  et de l'état adjoint  $\chi^\varepsilon$ .

2. plus précisément : on utilisera l'antisymétrie de  $iH$  et  $i\mu$ .

### 1.1.3 L'algorithme

L'idée est donc d'appliquer une méthode de descente par directions alternées, en optimisant  $J_{\parallel}$  tour à tour par rapport à  $\varepsilon$  (en parallèle!) et par rapport à  $\Lambda$  (cette étape étant explicite, d'après le théorème 7). Ceci conduit à l'algorithme suivant :

**Algorithme 6.** *Étant donné un contrôle initial  $\varepsilon^0$ , un réel  $\nu > 0$  et le critère d'arrêt*

$$c(\varepsilon) = J(\varepsilon) + \nu \sum_{\ell=0}^{N-2} |\varepsilon_{\ell}(T_{\ell+1}) - \varepsilon_{\ell+1}(T_{\ell+1})|^2,$$

*supposons qu'à l'étape  $k$  on dispose d'un contrôle  $\varepsilon^k$  et d'une variable  $Tol \geq 0$ . Le calcul de  $\varepsilon^{k+1}$  est effectué selon :*

1. *Si  $c(\varepsilon^k) \leq Tol$ , arrêter le calcul. Sinon passer à l'étape 2.*
2. *Calculer  $\psi^k = \psi^{\varepsilon^k}$ , la solution de (1.2) avec  $\varepsilon = \varepsilon^k$ .*
3. *Calculer  $\chi^k = \chi^{\varepsilon^k}$ , la solution de (1.3) avec  $\varepsilon = \varepsilon^k$ .*
4. *Calculer  $\Lambda^k = \Lambda^{\varepsilon^k}$  à l'aide de  $\psi^k$  et  $\chi^k$ , selon (1.4).*
5. *Sur chaque intervalle  $[T_{\ell}, T_{\ell+1}]$ , calculer en parallèle un contrôle  $\varepsilon_{\ell}^{k+1}$  solution du problème*

$$\min_{\varepsilon_{\ell}} J_{\ell}(\varepsilon_{\ell}, \lambda_{\ell}^k, \lambda_{\ell+1}^k).$$

6. *Définir  $\varepsilon^{k+1}$  comme la concaténation des contrôles  $\varepsilon_{\ell}^{k+1}$  ainsi obtenus.*
7. *Assigner  $k \leftarrow k + 1$ . Retourner à l'étape 1.*

Une solution simple pour effectuer l'étape 5 consiste à appliquer quelques itérations d'un algorithme monotone, décrit au chapitre 1 de la partie I.

L'algorithme 6 est décrit schématiquement sur la figure 1.1.

### 1.1.4 Convergence

L'atout majeur de l'algorithme précédent est qu'il converge vers un point critique de la fonctionnelle  $J$  (c.f. (1.1)) initialement considérée.

**Théorème 8.** *Supposons  $\alpha > 0$ . Étant donné un contrôle initial  $\varepsilon^0$ , soit la suite  $(\varepsilon^k)_{k \in \mathbb{N}}$  obtenue par l'algorithme 6, où l'étape 5 est effectuée par un nombre fini non nul d'itérations d'un algorithme monotone. Alors, la suite  $(\varepsilon^k)_{k \in \mathbb{N}}$  converge vers un point critique de  $J$ .*

Ce théorème est en fait valable dans des cadres plus larges : j'utilise un algorithme monotone à l'étape 5 car c'est à ma connaissance la méthode la plus adaptée à ce problème. Le résultat reste cependant valable si on utilise un autre algorithme convergent pour résoudre les  $N$  sous-problèmes de l'étape 5. Il reste également valable, si, au lieu de résoudre à chaque itération précisément les équations de Schrödinger aux étapes 2 et 3, on les résout de manière approchée, mais de plus en plus finement. Cette dernière démarche permet de gagner du temps lors du calcul effectif. Dans tous les cas, le calcul du contrôle est effectué uniquement en parallèle.

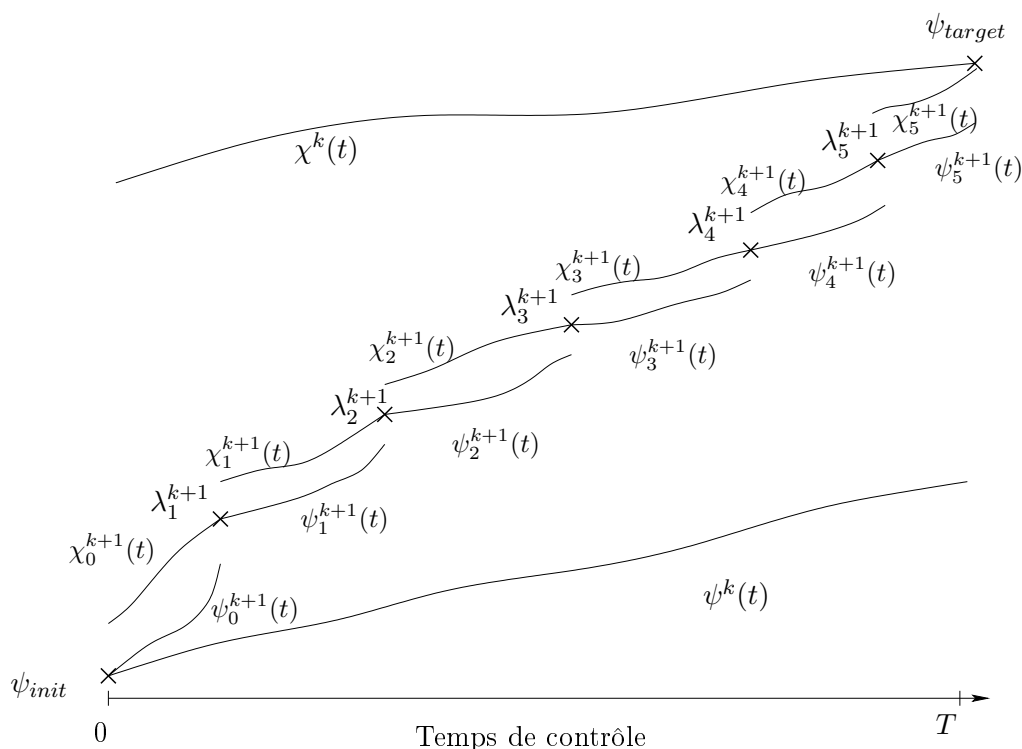


FIGURE 1.1 – Représentation schématique de l'algorithme. L'optimisation est réalisée en parallèle sur chaque intervalle  $[T_\ell, T_{\ell+1}]$ . La suite d'états  $\lambda_\ell^k$  est mise à jour à chaque itération.

### 1.1.5 Résultats numériques

Toutes les expériences qui suivent ont été réalisées sur le problème du contrôle de l'orientation et de l'alignement de la molécule *HCN* décrit plus précisément à la section 1.2.1 du chapitre 1 de la partie I et surtout dans l'article [A]. La figure 1.2 montre l'évolution du contrôle au cours des itérations de l'algorithme 6 ainsi que le contrôle optimal, obtenu sans parallélisation.

On voit que les discontinuités disparaissent peu à peu au cours des itérations. Le nombre de sous-intervalles joue un rôle important dans la procédure d'optimisation, comme le montre la figure 1.3.

Des tests numériques simples montrent que l'algorithme permet de diminuer d'à peu près un ordre de grandeur le temps de calcul d'un contrôle optimal.

## 1.2 Algorithme pour les équations paraboliques

Dans un travail en cours et en collaboration avec Yvon Maday et Kamel Riahi, j'ai proposé un algorithme pour un problème standard de contrôle de l'équation de la chaleur. Ce travail a avant tout pour but d'étendre l'approche des cibles intermédiaires à des cadres ne présentant pas les particularités de symétries propres aux équations hyperboliques. Je présente ici une première méthode, qui reprend certaines idées de la section précédente.

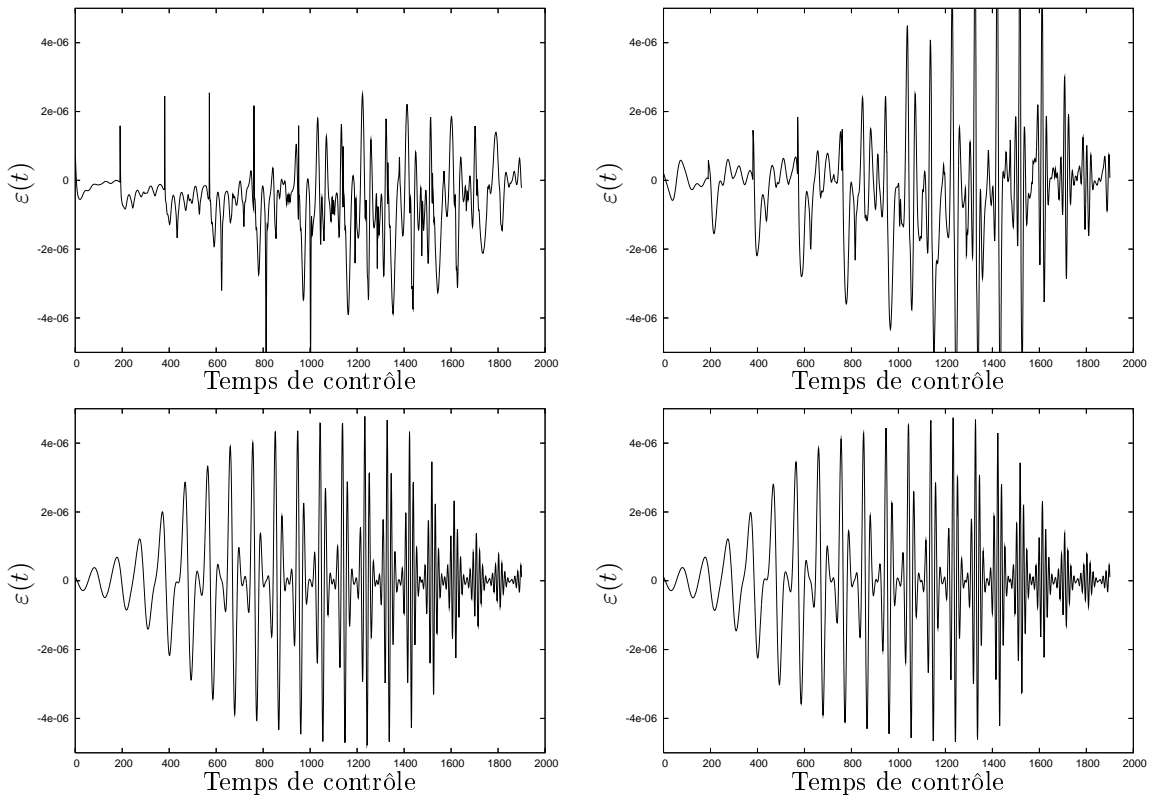


FIGURE 1.2 – En haut : contrôle calculé en parallèle après une (à gauche) et 10 itérations (à droite) par l’algorithme 6. En bas : contrôle calculer en parallèle après 250 itérations par l’algorithme 6 et par un algorithme monotone ( $N = 1$ ), également après 250 itérations.

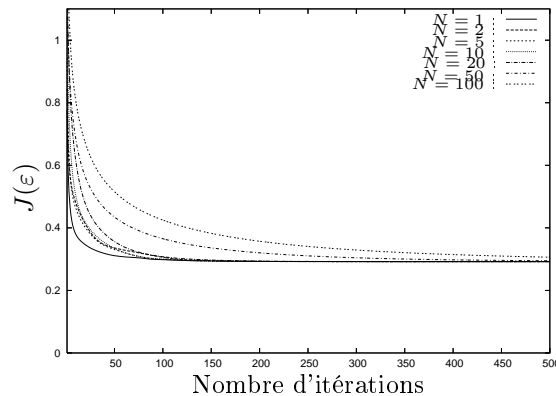


FIGURE 1.3 – Évolution des valeurs de la fonctionnelle de coût au cours de 500 itérations, avec différents nombres de sous-intervalles.

### 1.2.1 Le problème

On se donne une nouvelle fois un temps de contrôle  $T > 0$  et un paramètre de pénalisation  $\alpha > 0$ . Le problème de contrôle que l’on considère consiste à résoudre le problème d’optimisation :

$$\min_{v \in L([0,T]; L^2(\Omega_c))} J(v),$$

où  $J$  est la fonctionnelle :

$$J(v) = \frac{1}{2} \|y(T) - y_{target}\|_2^2 + \frac{\alpha}{2} \int_0^T \|v(t)\|_{c,2}^2 dt.$$

Le domaine  $\Omega$  est une partie bornée et connexe de  $\mathbb{R}^2$ . L'état  $y$  évolue à partir de la condition initiale  $y_0$  sur  $[0, T]$  selon l'équation

$$\partial_t y - \nu \Delta y = Bv.$$

Dans cette dernière,  $\Delta$  est le Laplacien,  $v$  est le terme de contrôle, qui consiste en une source de chaleur agissant sur une partie  $\Omega_c$  de  $\Omega$ . L'opérateur  $B$  est l'injection canonique de  $\Omega_c$  dans  $\Omega$ . Le système d'optimalité de ce problème peut s'écrire comme suit.

$$\begin{cases} \partial_t y - \nu \Delta y &= Bv & \text{on } [0, T] \times \Omega \\ y(0) &= y_0, \end{cases} \quad (1.5)$$

$$\begin{cases} \partial_t p + \nu \Delta p &= 0 & \text{on } [0, T] \times \Omega \\ p(T) &= y(T) - y_{target}, \end{cases} \quad (1.6)$$

$$\alpha v + B^* p = 0, \quad (1.7)$$

où  $B^*$  est l'opérateur adjoint de  $B$ .

Ce problème est bien posé puisque la fonctionnelle est strictement convexe (à cause du terme  $\alpha > 0$ ). Le système (1.5–1.7) a par conséquent une unique solution, notée dans la suite  $v^*$ . On note de plus  $y^*$ ,  $p^*$  l'état et l'état adjoint associé.

### 1.2.2 Parallélisation

Présentons maintenant un cadre adapté à la résolution en parallèle de ce problème. On considère  $N \geq 1$  et une partition de  $[0, T]$

$$[0, T] = \cup_{n=0}^{N-1} [t_\ell, t_{\ell+1}].$$

Étant donné un contrôle  $v$  et les état et état adjoint associés  $y, p$ , on définit une trajectoire cible, arrivant au temps  $T$  sur l'état désiré :

$$\begin{cases} \chi &= y - p & \text{sur } [0, T] \times \Omega \\ \chi(T) &= y_{target}. \end{cases} \quad (1.8)$$

Celle-ci est bien sûr l'analogue de la trajectoire interpolée  $\Lambda^\varepsilon$  qui a été introduite au théorème 7 dans le cas des équations hyperboliques. Elle n'est pas non plus la solution d'une équation particulière mais va nous servir à définir un ensemble de cibles utilisées dans les sous-problèmes traités en parallèle :

$$\min_{v_\ell \in L^2([t_\ell, t_{\ell+1}]; L^2(\Omega_c))} J_\ell(v_\ell),$$

avec

$$J_\ell(v_\ell) = \frac{1}{2} \|y_\ell(t_{\ell+1}) - \chi(t_{\ell+1})\|_2^2 + \frac{\alpha}{2} \int_{T_\ell}^{T_{\ell+1}} \|v_\ell(t)\|_{c,2}^2 dt, \quad (1.9)$$

où  $\|\cdot\|_{c,2}$  désigne la norme  $L^2$  associée au domaine  $\Omega_c$  et où la fonction  $y_\ell$  est définie par

$$\begin{cases} \partial_t y_\ell - \nu \Delta y_\ell &= Bv_\ell & \text{sur } [t_\ell, t_{\ell+1}] \times \Omega \\ y_\ell(t_\ell) &= y(t_\ell). \end{cases} \quad (1.10)$$

Ce sous problème de contrôle optimal est donc paramétré par  $v$ , via les états  $y$  et  $p$ . Sa structure est la même que le problème de contrôle original et en particulier également strictement convexe. Son système d'optimalité est donné par (1.10) et les équations :

$$\begin{cases} \partial_t p_\ell + \nu \Delta p_\ell &= 0 & \text{sur } [t_\ell, t_{\ell+1}] \times \Omega \\ p_\ell(t_{\ell+1}) &= y(t_{\ell+1}) - \chi(t_{\ell+1}), \end{cases} \quad (1.11)$$

$$\alpha v_\ell + B^* p_\ell = 0, \quad (1.12)$$

On note  $v_\ell^*$  sa solution.

L'intérêt de toute cette construction réside dans le résultat suivant.

**Lemme 4.** *Notons  $\chi^*$  la trajectoire cible définie par (1.8) avec  $y = y^*$ ,  $p = p^*$  et  $y_\ell^*, p_\ell^*, v_\ell^*$  les solutions de (1.10–1.12) avec  $y = y^*$  et  $p = p^*$ . On a :*

$$v_\ell^* = v_{|[t_\ell, t_{\ell+1}]}$$

Ce lemme peut être vu comme un résultat de consistance apportant une garantie en cas de convergence d'une méthode de résolution du problème parallèle (1.5–1.7).

### 1.2.3 L'algorithme

Présentons maintenant une telle méthode.

**Algorithme 7.** *Étant donné un contrôle initial  $v^0$ , un réel  $\nu > 0$  et le critère d'arrêt*

$$c(v) = J(v) + \nu \sum_{\ell=0}^{N-2} \|v_\ell(T_{\ell+1}) - v_{\ell+1}(T_{\ell+1})\|_{c,2}^2,$$

*on suppose qu'à une étape  $k$ , le contrôle  $v^k$  ait été obtenu. Le calcul du contrôle  $v^{k+1}$  est effectué comme suit.*

1. *Calculer  $y^k$ ,  $p^k$  et la trajectoire cible associée  $\chi^k$  selon (1.5), (1.6) et (1.8) respectivement.*
2. *Résoudre en parallèle les  $N$  sous-problèmes (1.9). Pour  $\ell = 1, \dots, N$ , soit  $\tilde{v}_\ell^{k+1}$  leurs solutions.*
3. *Définir  $v^{k+1}$  comme la concaténation des contrôles  $(\tilde{v}_\ell^{k+1})_{\ell=0, \dots, N-1}$ .*

Je n'ai pas détaillé l'étape de résolution en parallèle (étape 2) de manière à proposer une méthode générale de parallélisation. Les sous-problèmes étant strictement convexes, on peut par exemple effectuer quelques itérations d'une méthode de gradient conjugué.

### 1.2.4 Convergence

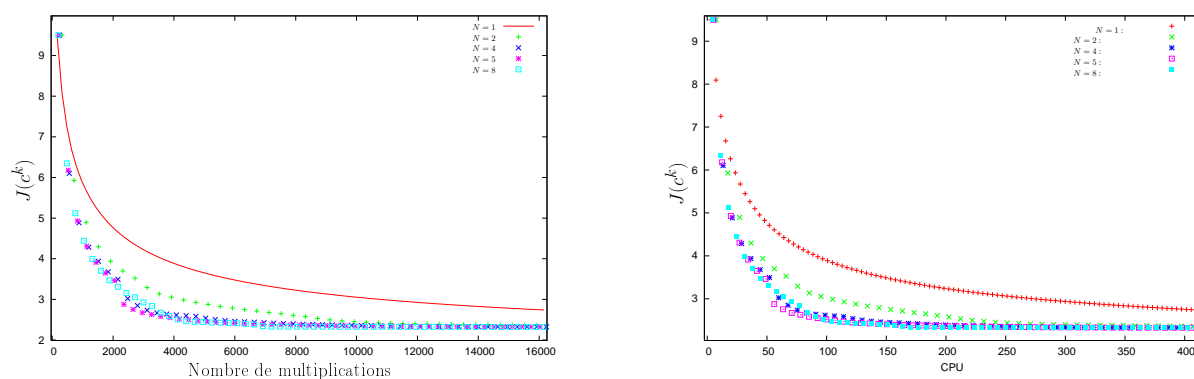
La convergence de la procédure précédente est garantie par le théorème suivant.

**Théorème 9.** *Étant donné un contrôle initial  $v^0$ , la suite  $(v^k)_{k \in \mathbb{N}}$  obtenue par l'algorithme 7, où l'étape 2 est effectuée par un algorithme convergent, converge vers  $v^*$ .*

De même que dans le cas hyperbolique, ce résultat peut donner lieu à de nombreuses variations. L'usage d'une méthode de résolution grossière dans les premières itérations de l'algorithme peut par exemple en constituer une.

### 1.2.5 Résultats numériques

Je termine cette section par quelques résultats numériques, obtenus sur l'exemple d'un domaine carré de  $\mathbb{R}^2$  à l'aide du logiciel freefem<sup>3</sup>. Quel que soit le critère de mesure du temps de calcul utilisé, on observe, par rapport à une méthode standard, une accélération importante de la résolution par l'algorithme 7. Dans ces deux exemples, on a considéré comme critère le nombre



de multiplications matrice-vecteur effectué et le temps CPU effectif lors d'une implémentation parallèle basée sur MPI<sup>4</sup>. L'accélération observé est de l'ordre de 10 dans les deux cas.

3. <http://www.freefem.org/>

4. <http://www.lam-mpi.org/>



## Chapitre 2

# Analyse numérique et formulation co-rotationnelle

**Résumé** : les schémas numériques de simulation en élastodynamique présentent de nombreuses faiblesses lorsqu'on considère de grandes vitesses de rotation. Sauf à utiliser de très petits pas de temps, les modèles non-linéaires engendrent des boucles internes de résolutions qui convergent difficilement et leurs versions linéarisées donnent des simulations médiocres. Pour résoudre ces problèmes, on utilise souvent une formulation, dite *co-rotationnelle*, dans laquelle le mouvement est décomposé en une partie solide et une partie élastique de petite amplitude [G]. Mon travail dans ce cadre a consisté à construire des discrétisations conservant l'énergie ainsi que le moment cinétique et à en étudier les propriétés. Des simulations démontrent leur efficacité par rapport aux méthodes classiques.

Par la suite, j'ai étudié le couplage de l'approche co-rotationnelle avec des algorithmes permettant de prendre en compte des phénomènes de contact avec et sans friction [I] : à chaque pas de temps, trois boucles sont en fait mises en œuvre pour calculer l'angle, les points de contact et leur situation vis-à-vis du cône de Coulomb. La stratégie proposée repose sur une combinaison stable de différents algorithmes. Ce travail est complété par des tests numériques.

### Introduction

La discrétisation en temps des problèmes d'élastodynamique est connue pour être un problème compliqué : des lois non-linéaires sont souvent nécessaires à la description correcte du mouvement et des systèmes complexes d'équations doivent être résolus à chaque pas de temps. La question devient encore plus difficile lorsqu'on souhaite simuler des mouvements de rotation rapide. Dans ce cas, l'usage d'un pas de temps petit est nécessaire pour préserver la convergence des méthodes itératives utilisées dans les boucles internes. Des artefacts numériques apparaissent aussi parfois, en se manifestant sous forme d'oscillations. Une linéarisation des équations s'avère de plus inefficace : en pratique celle-ci se traduit par une contribution exagérée de la rotation dans la déformation élastique.

Pour résoudre ces problèmes, une démarche consiste à décomposer le mouvement en une partie solide et une partie relevant de la déformation élastique. Cette approche fut initialement introduite par Maurice Biot et Jacques Romain dans [5] pour traiter des problèmes issus de l'aéronautique. L'idée fut reprise et formalisée dans un cadre abstrait par Fraejis de Veubeke dans [27] où il proposa plusieurs critères pour garantir l'unicité de la décomposition. De nos jours, de telles décompositions sont souvent utilisées sous la dénomination *formulation co-rotationnelle*.

En collaboration avec Barbara Wohlmuth et Alexander Weiss, mon travail sur ce thème a consisté à concevoir un schéma numérique associé à une telle formulation. De nombreux algorithmes avaient, bien entendu, déjà été proposés (voir à ce propos la présentation très complète de Felippa dans [10]) mais à ma connaissance aucun n'assurait la préservation de l'énergie et du moment cinétique au niveau discret ou ne bénéficiait d'une analyse de stabilité. C'est sous cette contrainte qu'ont été conçus les algorithmes de ce chapitre. Il s'est avéré par la suite que les schémas permettaient de travailler avec des pas de temps beaucoup plus grands que ceux utilisés dans les méthodes usuelles.

## 2.1 Schéma conservatif pour la formulation co-rotationnelle

*Cette section décrit les résultats obtenus dans [G].*

À part dans la dernière section, le problème est abordé dans le cadre de la dimension 2.

### 2.1.1 Le modèle

On considère un domaine borné  $\Omega \subset \mathbb{R}^2$  de frontière  $\Gamma := \partial\Omega$  suffisamment régulière par morceaux ainsi qu'un solide élastique de densité  $\rho(x)$ . Quitte à opérer une translation, on peut supposer que le centre de gravité est situé en  $(0, 0)$ , c'est-à-dire que  $\int_{\Omega} \rho x = 0$ . Ce solide est soumis à des forces volumiques  $f(x, t)$  et de bord  $g(x, t)$ . On note  $\varphi : \Omega \times [0, T] \rightarrow \mathbb{R}^2$  la fonction décrivant sa déformation. Dans un cadre complètement non-linéaire, l'évolution de  $\varphi$  est décrite sous la forme faible suivante :

$$\int_{\Omega} \rho \ddot{\varphi} \cdot \eta + \int_{\Omega} \frac{\partial}{\partial E} W(E(d)) : F^{\top} \nabla \eta = \int_{\Omega} f \cdot \eta + \int_{\Gamma} g \cdot \eta, \quad (2.1)$$

où  $\eta$  est une fonction de  $V := [H^1(\Omega)]^2$ ,  $d := \varphi - x$  et

$$F := \nabla \varphi = I + \nabla d \quad (2.2)$$

est le gradient de la déformation. Le tenseur de Green-Lagrange  $E$  est défini par

$$E(d) := \frac{1}{2}(\nabla d + \nabla d^{\top} + \nabla d^{\top} \nabla d).$$

On considère de plus que le solide suit le modèle de Saint Venant–Kirchhoff où l'énergie emmagasinée  $W(E(d))$  est calculée par

$$W(E(d)) := \frac{\lambda}{2} (\text{tr}(E(d)))^2 + \mu \text{tr}(E(d)^2),$$

où  $\lambda$  et  $\mu$  sont les paramètres de Lamé. Pour ce type de matériaux linéaires le second tenseur de contrainte de Piola-Kirchhoff  $\Sigma(d) := \frac{\partial}{\partial E} W(E(d))$  s'écrit

$$\Sigma(d) = 2\mu E(d) + \lambda \text{tr} E(d) Id, \quad (2.3)$$

avec  $Id$  la matrice identité de  $\mathbb{R}^3$ .

### 2.1.2 Décomposition co-rotationnelle

L'idée est donc maintenant de décrire le mouvement en terme de déplacement solide et de déformation purement élastique. Une telle décomposition s'écrit sous la forme générale

$$\varphi(x, t) = \tau(t) + R_{\theta(t)}(x + u(x, t)). \quad (2.4)$$

Dans cette équation  $\tau(t) \in \mathbb{R}^2$  et  $R_{\theta}$  correspondent respectivement à la translation et à la rotation solide par lesquelles on décrit le mouvement global. Une propriété intéressante du paramétrage (2.4) est qu'il préserve le tenseur  $E$  :

$$E(d) = E(u),$$

ce qui montre que le modèle non-linéaire élimine naturellement la partie rigide du mouvement apparaissant dans  $\varphi$ . Pour garantir l'unicité de la décomposition, on impose en plus les relations

$$\int_{\Omega} \rho u = 0, \quad \int_{\Omega} \rho u \cdot \Pi x = 0,$$

avec  $\Pi$  la rotation de  $\pi/2$ . Ces relations traduisent l'orthogonalité de la déformation  $u$  par rapport aux translations et aux rotations et garantissent donc, en ce sens, le caractère « pur » de la déformation élastique. Le paramètre  $u$  est donc recherché dans l'espace  $X := X^{\text{trans}} \cap X^{\text{rot}}$  défini par

$$X^{\text{trans}} := \left\{ u \in V; \int_{\Omega} \rho u = 0 \right\}, \quad X^{\text{rot}} := \left\{ u \in V; \int_{\Omega} \rho u \cdot \Pi x = 0 \right\}.$$

En posant  $v = R_{\theta}^{\top} \eta$ , ce changement de variable transforme (2.1) en l'équation :

$$\int_{\Omega} \rho (\dot{s} + \dot{\theta} \Pi s) \cdot v + \int_{\Omega} \Sigma(u) : \hat{F}^{\top} \nabla v = B_{\theta}(v) \quad \forall v \in X, \quad (2.5)$$

avec  $\hat{F} := I + \nabla u$  et

$$B_{\theta}(v) := \int_{\Omega} f \cdot R_{\theta} v + \int_{\Gamma} g \cdot R_{\theta} v$$

et où la vitesse relative  $s$  est définie par

$$s(x, t) := R_{\theta(t)}^{\top} (\dot{\varphi}(x, t) - \dot{\tau}(t)) = \dot{u}(x, t) + \dot{\theta}(t) \Pi(x + u(x, t)). \quad (2.6)$$

L'accélération est quant à elle définie par

$$\ddot{\varphi}(x, t) = \ddot{\tau}(t) + R_{\theta(t)} \left( \dot{s}(x, t) + \dot{\theta}(t) \Pi s(x, t) \right). \quad (2.7)$$

L'équation (2.5) n'est qu'une implication de (2.1), puisque son inconnue est dans un espace plus petit. Pour obtenir une description complète, il faut ajouter les relations de conservation de la quantité de mouvement et du moment cinétique (qui sont également des conséquences de (2.1)) :

$$\mathcal{M} \ddot{\tau} = \int_{\Omega} f + \int_{\Gamma} g, \quad (2.8)$$

et

$$\dot{\mathcal{J}} = B_{\theta}(\Pi(x + u)), \quad (2.9)$$

où  $\mathcal{M}$  et  $\mathcal{J}$  sont définis par

$$\mathcal{M} := \int_{\Omega} \rho, \quad \mathcal{J} := \int_{\Omega} \rho s \cdot \Pi(x + u).$$

### 2.1.3 Linéarisation

Pour accélérer le calcul et dans le cadre de petites déformations, on peut considérer une version où le terme élastique de (2.5) est linéarisé. Celle-ci s'écrit :

$$\int_{\Omega} \rho \left( \dot{s} + \dot{\theta} \Pi s \right) \cdot v + \int_{\Omega} \sigma(u) : \nabla v = B_{\theta}(v), \quad (2.10)$$

où  $\sigma(u) := 2\mu\varepsilon(u) + \lambda \text{tr}\varepsilon(u) Id$ . Deux approximations ont été introduites. L'une sur le tenseur  $E$  :

$$E(u) \approx \varepsilon(u) := \frac{1}{2}(\nabla u + \nabla u^{\top}),$$

l'autre sur le gradient de la déformation  $\hat{F} \approx Id$ . Pour ce modèle, l'existence et l'unicité d'une solution au système (2.8–2.10) a été obtenue par Céline Grandmont, Yvon Maday et Paul Metier dans [13].

### 2.1.4 Discrétisation en temps et schémas de résolution

Les discrétisations de (2.8–2.10) que j'ai conçues dans ce cadre conservent l'énergie au niveau discret. Expliquons brièvement la démarche suivie lors de leur construction. En utilisant (2.6) et (2.7), on peut écrire

$$R_{\theta}^{\top}(\ddot{\varphi} - \ddot{\tau}) = \left( \frac{d}{dt} + \dot{\theta} \Pi \right) s \quad (2.11)$$

$$\begin{aligned} &= \left( \frac{d}{dt} + \dot{\theta} \Pi \right)^2 (x + u) \\ &= \ddot{u} + \Pi \left( \frac{d}{dt}(\dot{\theta} u) + \dot{\theta} \dot{u} \right) - (\dot{\theta})^2 (x + u). \end{aligned} \quad (2.12)$$

Le premier schéma proposé repose sur une discrétisation de chaque terme de (2.12). Dans le second schéma, on discrétise tout d'abord  $s = \left( \frac{d}{dt} + \dot{\theta} \Pi \right) (x + u)$  (voir (2.21)) puis  $R_{\theta}^{\top}(\ddot{\varphi} - \ddot{\tau})$  via (2.11) (voir le premier terme de (2.20)). Des deux approches, seule la deuxième garantit la préservation du moment cinétique.

Un ingrédient essentiel dans ce qui suit est la discrétisation au point milieu qui vérifie :

$$\dot{a}_{n+1/2} b_{n+1/2} + a_{n+1/2} \dot{b}_{n+1/2} = [\dot{ab}]_{n+1/2},$$

où les notations suivantes ont été utilisées :

$$\star_{n+1/2} := \frac{\star_{n+1} + \star_n}{2}, \quad \dot{\star}_{n+1/2} := \frac{\star_{n+1} - \star_n}{\Delta t}.$$

Ici,  $\Delta t$  est le pas de temps considéré et  $\star$  est une quantité générique.

Avant de présenter les algorithmes, il reste à définir l'énergie et le moment cinétique discret. A un pas de temps  $t_n$ , on les calculera par

$$\mathcal{E}_n := \frac{1}{2} \mathcal{M} \dot{\tau}_n^2 + \frac{1}{2} \int_{\Omega} \rho s_n^2 + \frac{1}{2} \int_{\Omega} \sigma(u_n) : \varepsilon(u_n), \quad (2.13)$$

et

$$\mathcal{J}_n := \int_{\Omega} \rho s_n \cdot \Pi(x + u_n).$$

Ici  $s_n$  est une discrétisation de (2.6), qui n'est pas la même dans les deux schémas, schémas que l'on peut (enfin!) expliciter.

**Algorithme 8.** Soit  $\tau_0 \in \mathbb{R}^2$ ,  $\dot{\tau}_0 \in \mathbb{R}^2$ ,  $\theta_0 \in \mathbb{R}$ ,  $\dot{\theta}_0 \in \mathbb{R}$ ,  $u_0 \in X$ , et des forces extérieures  $(f_{n+1/2})_{n \in \mathbb{N}}$  et  $(g_{n+1/2})_{n \in \mathbb{N}}$  données. On suppose que  $\tau_n, \dot{\tau}_n, \theta_n, \dot{\theta}_n$  et  $u_n, \dot{u}_n \in X$  ont déjà été calculés. Le calcul de  $\tau_{n+1}, \dot{\tau}_{n+1}, \theta_{n+1}, \dot{\theta}_{n+1}$  et  $u_{n+1}, \dot{u}_{n+1} \in X$  est effectué en résolvant

$$\mathcal{M}\ddot{\tau}_{n+1/2} = \int_{\Omega} f_{n+1/2} + \int_{\Gamma} g_{n+1/2}, \quad (2.14)$$

$$\begin{aligned} \frac{1}{\Delta t} \left( \int_{\Omega} \rho \dot{\theta}_{n+1} (x + u_{n+1})^2 - \int_{\Omega} \rho \dot{\theta}_n (x + u_n)^2 \right) + \int_{\Omega} \rho \Pi \frac{\dot{\theta}_{n+1} u_{n+1} + \dot{\theta}_n u_n}{2\dot{\theta}_{n+1/2}} \cdot \ddot{u}_{n+1/2} \\ = B_{n+1/2}(\Pi(x + u_{n+1/2})), \end{aligned} \quad (2.15)$$

$$\begin{aligned} \int_{\Omega} \rho \left( \ddot{u}_{n+1/2} + \Pi \frac{\dot{\theta}_{n+1} u_{n+1} - \dot{\theta}_n u_n}{\Delta t} + \Pi \dot{\theta}_{n+1/2} \dot{u}_{n+1/2} - \dot{\theta}_n \dot{\theta}_{n+1} (x + u_{n+1/2}) \right) \cdot v \\ + \int_{\Omega} \sigma(u_{n+1/2}) : \varepsilon(v) = B_{n+1/2}(v) \quad \forall v \in X, \end{aligned} \quad (2.16)$$

où

$$B_{n+1/2}(v) := \int_{\Omega} f_{n+1/2} \cdot v + \int_{\Omega} g_{n+1/2} \cdot v.$$

Dans ce schéma, la discrétisation de la vitesse relative est effectuée selon

$$s_n := \dot{u}_n + \dot{\theta}_n \Pi(x + u_n). \quad (2.17)$$

Le moment cinétique discret n'est conservé qu'à l'ordre  $\Delta t^2$  par l'algorithme. En revanche, le schéma suivant le conserve exactement.

**Algorithme 9.** Soit  $\tau_0 \in \mathbb{R}^2$ ,  $\dot{\tau}_0 \in \mathbb{R}^2$ ,  $\theta_0 \in \mathbb{R}$ ,  $u_0 \in X$ ,  $s_0 \in [H^1(\Omega)]^2$  et des forces extérieures  $(f_{n+1/2})_{n \in \mathbb{N}}$ , et  $(g_{n+1/2})_{n \in \mathbb{N}}$  données. On suppose que  $\tau_n, \dot{\tau}_n, \theta_n, \dot{\theta}_n$  et  $u_n, \dot{u}_n \in X$  ont déjà été calculés. Le calcul de  $\tau_{n+1}, \dot{\tau}_{n+1}, \theta_{n+1}, \dot{\theta}_{n+1}$  et  $u_{n+1}, \dot{u}_{n+1} \in X$  est effectué en résolvant

$$\mathcal{M}\ddot{\tau}_{n+1/2} = \int_{\Omega} f_{n+1/2} + \int_{\Gamma} g_{n+1/2}, \quad (2.18)$$

$$\dot{\mathcal{J}}_{n+1/2} = B_{n+1/2}(\Pi(x + u_{n+1/2})), \quad (2.19)$$

$$G(\dot{s}_{n+1/2}, s_{n+1/2}, \dot{\theta}_{n+1/2}, u_{n+1/2}; v) = B_{n+1/2}(v) \quad \forall v \in X, \quad (2.20)$$

avec

$$G(\dot{s}, s, \dot{\theta}, u; v) := \int_{\Omega} \rho(\dot{s} + \dot{\theta} \Pi s) \cdot v + \int_{\Omega} \sigma(u) : \varepsilon(v)$$

et

$$s_{n+1/2} := \dot{u}_{n+1/2} + \dot{\theta}_{n+1/2} \Pi(x + u_{n+1/2}). \quad (2.21)$$

C'est en fait l'équation (2.19) qui garantit la préservation du moment cinétique.

Comme annoncé précédemment, ces deux algorithmes préservent l'énergie au sens du théorème suivant.

**Théorème 10.** Les algorithmes 8 et 9 définis par (2.14–2.16) et (2.18–2.20), respectivement, vérifient

$$\mathcal{E}_{n+1} - \mathcal{E}_n = \Delta t \left( \int_{\Omega} f_{n+1/2} \cdot \dot{\varphi}_{n+1/2} + \int_{\Gamma} g_{n+1/2} \cdot \dot{\varphi}_{n+1/2} \right).$$

Ici,  $\mathcal{E}_n$  est l'énergie totale du système définie par (2.13) où  $s$  est définie par (2.17) dans l'algorithme 8 et par (2.21) dans l'algorithme 9. La discrétisation en temps de  $\dot{\varphi}$  est donnée par

$$\dot{\varphi}_{n+1/2} := \dot{\tau}_{n+1/2} + R_{\theta_{n+1/2}} s_{n+1/2}.$$

### 2.1.5 Discrétisation en espace et caractère bien posé de l'algorithme

On se restreint dans la suite à l'algorithme 9. On discrétise maintenant les déformations  $u_n$  suivant un espace d'éléments finis  $V_h$ , en gardant les mêmes notations que précédemment pour simplifier la présentation. Pour garantir que la discrétisation complète de  $u_n$  appartienne à  $X$ , il est nécessaire d'introduire deux multiplicateurs de Lagrange  $\alpha_{n+1/2} \in \mathbb{R}$  et  $\beta_{n+1/2} \in \mathbb{R}^2$  associés respectivement aux contraintes d'orthogonalité  $u_{n+1/2} \in X^{\text{trans}}$  et  $u_{n+1/2} \in X^{\text{rot}}$ . Le système complètement discrétisé qui découle de (2.20) est alors le suivant :

$$\begin{aligned} G(\dot{s}_{n+1/2}, s_{n+1/2}, \dot{\theta}_{n+1/2}, u_{n+1/2}; v) + \alpha_{n+1/2} \int_{\Omega} \rho v \cdot \Pi x + \int_{\Omega} \rho \beta_{n+1/2} \cdot v &= B_{n+1/2}(v) \quad \forall v \in V_h, \\ \int_{\Omega} \rho u_{n+1/2} \cdot \Pi x &= 0, \\ \int_{\Omega} \rho u_{n+1/2} &= 0. \end{aligned} \quad (2.22)$$

S'il est facile de calculer directement  $\beta_{n+1/2}$ , par

$$\beta_{n+1/2} = \frac{1}{\mathcal{M}} \left( \int_{\Omega} R_{\theta_{n+1/2}}^{\top} f_{n+1/2} + \int_{\Gamma} R_{\theta_{n+1/2}}^{\top} g_{n+1/2} \right) = R_{\theta_{n+1/2}}^{\top} \ddot{\tau}_{n+1/2},$$

le calcul de  $\alpha_{n+1/2}$ , paramètre associé à la rotation doit faire l'objet d'une sous-boucle. Les deux premières équations du système (2.22) sont alors réécrites sous la forme :

$$\left( \begin{array}{c|c} \tilde{G}(\theta_{n+1} - \theta_n, \Delta t) & M \Pi x \\ \hline (M \Pi x)^{\top} & 0 \end{array} \right) \cdot \left( \begin{array}{c} u_{n+1/2} \\ \alpha_{n+1/2} \end{array} \right) = \left( \begin{array}{c} \tilde{c}_{n+1/2}(\theta_{n+1} - \theta_n, \Delta t) \\ 0 \end{array} \right), \quad (2.23)$$

où

- $\tilde{G}(\delta, \Delta t) := M(2I + \delta \Pi)^2 + \Delta t^2 S$ ,
- $\tilde{c}_{n+1/2}(\delta, \Delta t) := M(2I + \delta \Pi)(2u_n - \delta \Pi x) + 2M \Delta t s_n + \Delta t^2 \tilde{B}_{n+1/2}$ ,
- $\tilde{B}_{n+1/2} = B_{n+1/2} - \int_{\Omega} \rho R_{\theta_{n+1/2}}^{\top} \ddot{\tau}_{n+1/2} \cdot v$ .

Dans ce cadre, l'équation (2.19) s'écrit

$$h(\theta_{n+1} - \theta_n, \Delta t) = 0, \quad (2.24)$$

avec

$$\begin{aligned} h(\delta, \Delta t) : &= 4\delta(x + u_{n+1/2}) \cdot M(x + u_{n+1/2}) \\ &+ \left( -\delta M(x + u_n) + 2\Delta t M \Pi s_n + \Delta t^2 \Pi R_{\theta_n + \delta/2}^{\top} \tilde{B}_{n+1/2} \right) \cdot (x + u_{n+1/2}), \end{aligned}$$

où  $u_{n+1/2}$  est la solution de (2.23) qui correspond à  $\theta_{n+1} - \theta_n = \delta$ . On a alors le résultat suivant :

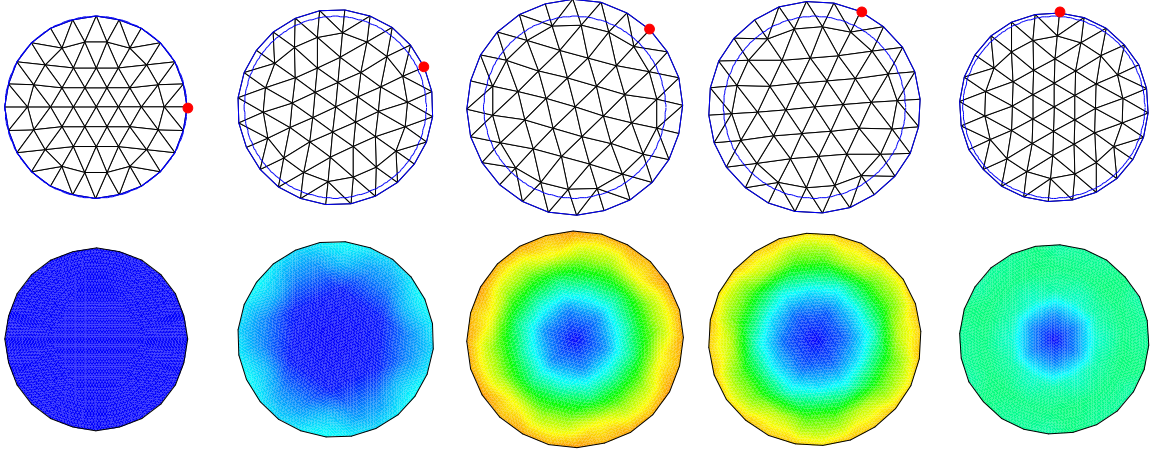


FIGURE 2.1 – Déformation du maillage et contrainte élastique à différents pas de temps, simulée à l'aide du schéma co-rotationnel.

**Théorème 11.** *Étant donnés  $\mathcal{E}_{max} > 0$ ,  $B_{max} > 0$ ,  $\nu > 0$ , supposons qu'à un temps  $t_n$  on ait  $\mathcal{E}_n \leq \mathcal{E}_{max}$ ,  $\|\tilde{B}_{n+1/2}\| \leq B_{max}$ , où  $\|\cdot\|$  est une norme sur  $V_h$ , et*

$$\nu \leq x \cdot M(x + u_n). \quad (2.25)$$

*Alors, il existe  $\Delta t^* > 0$  dépendant de  $\mathcal{E}_{max}$ ,  $B_{max}$ , et  $\nu$  tel que, étant donnés  $\Delta t \leq \Delta t^*$ ,  $\tau_n$ ,  $\dot{\tau}_n$ ,  $\theta_n$ ,  $u_n$ ,  $s_n$ , et les forces extérieures  $f_{n+1/2}$  et  $g_{n+1/2}$ , il existe  $\theta_{n+1} \in \mathbb{R}$  satisfaisant (2.24) (avec  $u_{n+1/2}$  la solution correspondante de (2.23)).*

*De plus, il existe  $\delta^+(\Delta t)$ ,  $\delta^-(\Delta t)$  tels que*

- $\lim_{\Delta t \rightarrow 0} \delta^+(\Delta t) = 2$ ,
- $\lim_{\Delta t \rightarrow 0} \delta^-(\Delta t) = -2$ ,
- $\theta_{n+1} \in [\theta_n + \delta^-(\Delta t), \theta_n + \delta^+(\Delta t)]$ .

Notons que la condition (2.25) avait déjà été obtenue par un autre biais dans [13].

### 2.1.6 Cas de la dimension 3

L'algorithme 9 a ensuite été adapté au cas de la dimension 3. Dans ce cadre, le moment cinétique ne peut plus être préservé strictement. Son module l'était cependant. Je ne détaille pas plus ici l'algorithme, qui bien que plus technique, reprend les idées de la dimension 2, et je renvoie à la publication [G] pour une description complète.

### 2.1.7 Tests numériques

Les algorithmes précédents ont été testés sur de nombreux exemples. Je n'en reproduis qu'un seul ici, en renvoyant une nouvelle fois à la publication elle-même pour d'autres résultats numériques. On considère un disque élastique, lancé en rotation sans déformation initiale. La solution numérique reproduit donc une oscillation entre l'énergie de rotation et l'énergie élastique emmagasinée. La suite de figures 2.1 montre le résultat de la simulation obtenue par l'algorithme 9. L'évolution des différentes énergies et de la vitesse angulaire est représentée sur la figure 2.2. On peut aussi comparer sur cet exemple simple différents schémas. Plusieurs qualités des schémas ont été mis en évidence par les tests numériques. Le principal avantage réside dans la possibilité de travailler avec des pas de temps beaucoup plus grands (10 fois plus grands dans certains exemples) que dans les approches standard.

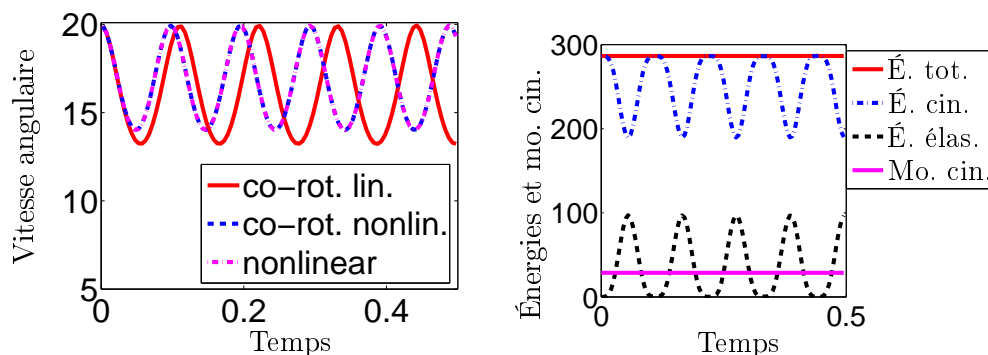


FIGURE 2.2 – À gauche : vitesse angulaire obtenue avec les schémas standard, co-rotationnel et co-rotationnel non linéaire. À droite : évolution des différentes énergies et du moment cinétique dans le cas du schéma co-rotationnel linéarisé.

## 2.2 Prise en compte du contact et de la friction

Cette section décrit les résultats obtenus dans [1].

Cette partie concerne un second travail, réalisé en collaboration avec Barbara Wohlmuth, Alexander Weiss et Patrice Hauret, où l'algorithme 9 a été adapté pour pouvoir traiter des problèmes de contact.

### 2.2.1 Modèle

On ajoute maintenant au modèle présenté à la section 2.1.1 la prise en compte de contact avec friction. On garde les notations précédentes. Pour être modélisé, le contact nécessite l'introduction d'une fonction  $\gamma_C : \Gamma_C \rightarrow \mathbb{R}$  qui modélise la distance entre la partie  $\Gamma_C \in \partial\Omega$  susceptible de rentrer en contact avec la frontière  $\Gamma_r$  d'un obstacle  $\Omega_r$  suivant le vecteur normal  $\nu$  par la formule :

$$\gamma_C := (\pi_\nu \varphi - \varphi) \cdot \nu,$$

où  $\pi_\nu$  est la projection le long du vecteur normal<sup>5</sup>  $\nu$ , comme l'illustre la figure 2.3. On introduit

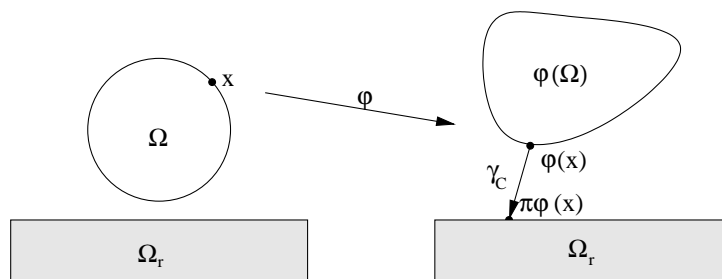


FIGURE 2.3 – Définition de  $\gamma_C$

également le tenseur des contraintes de Cauchy  $\sigma$  défini par

$$\sigma = \frac{1}{\det F} F^\top \Sigma F,$$

5. Cette fonction n'est pas la projection sur l'obstacle.

où  $\Sigma$  et  $F$  sont respectivement définis par (2.3) et par (2.2). Pour exprimer les conditions de non-pénétration et de friction, il est utile de définir la pression de contact et la force tangentielle exercée par le solide par

$$\sigma_\nu := (\sigma\nu) \cdot \nu, \quad \sigma_\tau := \sigma\nu - \sigma_\nu \nu.$$

Il sera également fait appel aux déplacements normaux et tangentiels :

$$d_\nu := d \cdot \nu, \quad d_\tau := d - d_\nu \nu.$$

On rappelle que le déplacement  $d$  est défini par  $d(x) := \varphi(x) - x$ . La condition de non-pénétration de  $\Gamma_C$  dans  $\Omega_r$  peut être exprimée via les conditions de Karush-Kuhn-Tucker :

$$\gamma_C \geq 0, \quad \sigma_\nu \leq 0, \quad \sigma_\nu \gamma_C = 0. \quad (2.26)$$

La première relation correspond à la condition de non-pénétration. La deuxième correspond à la réaction du support. En présence d'un phénomène de friction modélisé par la loi de Coulomb de coefficient  $\mathfrak{F}$ , une force tangentielle résistante s'applique au solide selon

$$|\sigma_\tau| - \mathfrak{F}|\sigma_\nu| \leq 0, \quad \dot{d}_\tau + \beta^2 \sigma_\tau = 0, \quad \dot{d}_\tau (|\sigma_\tau| - \mathfrak{F}|\sigma_\nu|) = 0, \quad (2.27)$$

où  $\beta^2$  est le coefficient de frottement dynamique.

Résoudre le problème de dynamique associé à ce contexte revient alors à trouver  $(d, \tilde{\lambda})$  tels que pour presque tout  $t \in [0, T]$ ,  $d(t) \in \mathcal{V}$ ,  $\tilde{\lambda}(t) \in \mathcal{M}(\tilde{\lambda})$  et

$$\begin{aligned} m(\ddot{d}, \eta) + a_{nl}(d, \eta) + b(\eta, \tilde{\lambda}) &= F(\eta), & \eta \in \mathcal{V}, \\ b_\tau(\dot{d}, \mu - \tilde{\lambda}) &\leq \langle \gamma_C, \mu_\nu - \tilde{\lambda}_\nu \rangle, & \mu \in \mathcal{M}(\tilde{\lambda}), \end{aligned} \quad (2.28)$$

où  $\mathcal{V}$  représente l'espace des déplacements,  $\mathcal{M}$  est l'espace de multiplicateurs de Lagrange, espace dual de la trace  $\mathcal{W}$  de  $\mathcal{V}$  restreint à  $\Gamma_C$  et où

$$\mathcal{M}(\tilde{\lambda}) := \{\mu \in \mathcal{M} : \langle \mu, \xi \rangle \leq \mathfrak{F} \langle \lambda_\nu, |\xi_\tau| \rangle, \text{ pour tout } \xi \in \mathcal{W} \text{ tel que } \xi_\nu \leq 0\}$$

et

$$\begin{aligned} m(\ddot{d}, \eta) &:= \int_\Omega \rho \ddot{d} \cdot \eta, \, dx, & a_{nl}(d, \eta) &:= \int_\Omega \frac{\partial W}{\partial E}(E(d)) : (F^\top \nabla \eta) \, dx \\ b(\eta, \mu) &:= b_\nu(\eta, \mu) + b_\tau(\eta, \mu) := \langle \eta_\nu, \mu_\nu \rangle + \langle \eta_\tau, \mu_\tau \rangle := \int_{\Gamma_C} \eta_\nu \mu_\nu \, ds + \int_{\Gamma_C} \eta_\tau \mu_\tau \, ds. \end{aligned}$$

### 2.2.2 Décomposition co-rotationnelle

En appliquant les mêmes raisonnements qu'à la section 2.1.2 pour introduire les variables de la formulation co-rotationnelle, on obtient les correspondances suivantes :

$$m(\ddot{d}, \eta) = m(\dot{s} + \dot{\theta} \Pi s, R_\theta^\top \eta).$$

$$a_{nl}(d, \eta) = \int_\Omega \frac{\partial W}{\partial E}(E(u)) : \left( (\hat{F}^\top R_\theta^\top \nabla \eta) \right) \, dx = a_{nl}(u, R_\theta^\top \eta) \approx a(u, R_\theta^\top \eta),$$

où, toujours comme à la section 2.1.2, on a linéarisé le terme associé à l'élasticité par  $a(u, \eta) := \int_\Omega \sigma(u) : \varepsilon(\eta)$ .

Le reste du travail de reformulation concerne les termes associés au contact et à la friction. On utilise les approximations suivantes :

$$\nu \approx R_\theta \nu_0, \quad \tau \approx R_\theta \tau_0,$$

où  $\nu_0$  et  $\tau_0$  sont respectivement les vecteurs normaux et tangentiels de la configuration initiale, c'est-à-dire de  $\Omega$ . Ceci conduit à

$$\begin{aligned} \gamma_C(x) &\approx (\pi_{R_\theta \nu_0} R_\theta(x+u) - R_\theta(x+u)) \cdot R_\theta \nu_0 \\ &= (R_\theta^\top \pi_{R_\theta \nu_0} R_\theta(x+u) - (x+u)) \cdot \nu_0 \\ &= (R_\theta^\top \pi_{R_\theta \nu_0} R_\theta(x+u) - x) \cdot \nu_0 - u_{\nu_0}, \end{aligned}$$

où  $u_{\nu_0} := u \cdot \nu_0$ . Pour continuer à simplifier l'écriture du modèle, on remarque qu'appliquer une rotation au solide puis appliquer une projection revient au même que d'appliquer une projection suivant  $\nu_0$  sur l'image de l'obstacle par la rotation inverse (voir la figure 2.4) :

$$R_\theta^\top \pi_{R_\theta \nu_0} R_\theta = \pi_{\nu_0}^\theta,$$

où l'on a noté  $\pi_{\nu_0}^\theta$  la projection sur  $R_\theta^\top(\Omega_r)$ .

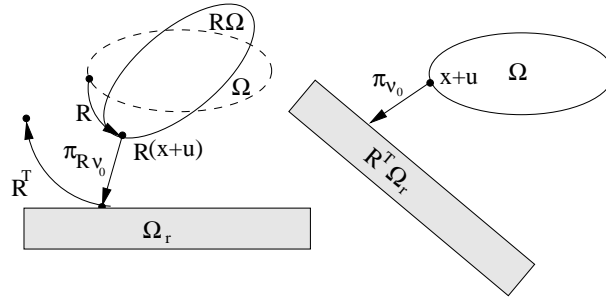


FIGURE 2.4 – Composition de la projection avec la rotation. À gauche : opérateur  $R_\theta^\top \pi_{R_\theta \nu_0} R_\theta$ , à droite : opérateur  $\pi_{\nu_0}^\theta$ .

En utilisant l'approximation  $x+u \approx x$ , on déduit :

$$\gamma_C(x) \approx \gamma_{C,\theta} - u_{\nu_0},$$

où la fonction  $\gamma_{C,\theta}$  est définie par  $\gamma_{C,\theta} := (\pi_{\nu_0}^\theta x - x) \cdot \nu_0$ . La condition de non-pénétration (2.26) s'écrit alors

$$u_{\nu_0} \leq \gamma_{C,\theta}, \quad \tilde{\sigma}_{\nu_0}(u) \leq 0, \quad \tilde{\sigma}_{\nu_0}(u)(u_{\nu_0} - \gamma_{C,\theta}) = 0, \quad (2.29)$$

où  $\tilde{\sigma} = R_\theta^\top \sigma R_\theta$ , tandis que la condition de friction (2.27) devient :

$$|\sigma_{\tau_0}(u)| - \mathfrak{F}|\sigma_{\nu_0}(u)| \leq 0, \quad s_{\tau_0} + \beta^2 \sigma_{\tau_0}(u) = 0, \quad s_{\tau_0} (|\sigma_{\tau_0}(u)| - \mathfrak{F}|\sigma_{\nu_0}(u)|) = 0,$$

où l'on a posé  $s_{\tau_0} := s - (s \cdot \nu_0)\nu_0$ . En introduisant  $\lambda := R_\theta^\top \tilde{\lambda}$  et l'espace

$$\mathcal{X} := \left\{ u \in \mathcal{V}, \int_\Omega \rho u \cdot \Pi x = 0 \right\},$$

on déduit de l'équation (2.28) que le problème exprimé dans les nouvelles variables revient à trouver  $(u, \lambda, \theta)$  avec  $u(t) \in \mathcal{X}$ ,  $\lambda(t) \in \mathcal{M}(\lambda(t))$ ,  $\theta(t) \in \mathbb{R}$  tels que

$$\begin{aligned} m(\dot{s} + \dot{\theta}\Pi s, \Pi r) &= F_\theta(\Pi r) - b(\Pi r, \lambda), \\ m(\dot{s} + \dot{\theta}\Pi s, v) + a(u, v) + b(v, \lambda) &= F_\theta(v), \quad v \in \mathcal{X}, \\ b_{\nu_0}(u, \mu - \lambda) + b_{\tau_0}(s, \mu - \lambda) &\leq \langle \gamma_{C, \theta}, \mu_{\nu_0} - \lambda_{\nu_0} \rangle, \quad \mu \in \mathcal{M}(\lambda), \end{aligned} \quad (2.30)$$

où l'on a posé  $F_\theta(v) := F(R_\theta v)$ . Dans la première équation, on a posé  $r = x + u$ . Ce système d'équations assure la conservation de l'énergie dans le sens suivant.

**Lemme 5.** *La solution  $(u, \lambda, \theta)$  de (2.30) vérifie*

$$\dot{\mathcal{E}} = F_\theta(s) - b_{\tau_0}(s, \lambda),$$

où l'énergie  $\mathcal{E}$  est définie par

$$\mathcal{E} := \frac{1}{2} \left( \int_{\Omega} \rho |s|^2 + \int_{\Omega} \sigma(u) : \varepsilon(u) \right) = \frac{1}{2} (m(s, s) + a(u, u)).$$

La conservation du moment cinétique est aussi vérifiée au niveau continu.

**Lemme 6.** *La première équation de (2.30) est équivalente à la conservation du moment cinétique :*

$$\dot{\mathcal{J}} = F_\theta(\Pi r) - b(\Pi r, \lambda)$$

avec

$$\mathcal{J} := \int_{\Omega} \rho s \cdot \Pi r = m(s, \Pi r).$$

### 2.2.3 Discrétisation en temps et schémas de résolution

L'adaptation des techniques présentées à la section 2.1.4 permet de discrétiser sans difficulté les deux premières équations du système (2.30). La condition de non-pénétration donnée par la troisième équation est par contre plus délicate à traiter. D'un point de vue technique, la conservation de l'énergie au niveau discret lors d'un contact nécessite d'imposer une relation dite de *persistance*, introduite par Laursen et Chawla [8, 16]. Cette relation s'écrit  $\sigma_\nu d_\nu = 0$  et traduit le fait que la vitesse normale est nulle tant que le contact a lieu. Malheureusement, Laursen et Chawla ont également montré que cette relation était incompatible avec la condition de non-pénétration (2.29). Celle-ci est donc remplacée par la condition de persistance :

$$b_{\nu_0}(s^{n+\frac{1}{2}}, \lambda^{n+\frac{1}{2}}) = 0. \quad (2.31)$$

On peut montrer qu'avec cette modification, on autorise des pénétrations de l'ordre de  $s\Delta t$ . En contrepartie, la nouvelle relation permet d'obtenir la conservation de l'énergie. Combinée avec les conditions de frottement, l'équation (2.31) devient :

$$b(s^{n+\frac{1}{2}}, \mu - \lambda^{n+\frac{1}{2}}) \leq \langle \hat{\gamma}_{C, \theta}^n, \mu_{\nu_0} - \lambda_{\nu_0}^{n+\frac{1}{2}} \rangle, \quad \mu \in \mathcal{M}(\lambda^{n+\frac{1}{2}}).$$

On obtient finalement le schéma suivant :

**Algorithme 10.** *Étant donnés  $\theta_0 \in \mathbb{R}$ ,  $u_0 \in X$ ,  $s_0 \in [H^1(\Omega)]^2$ , des forces extérieures  $(f^{n+\frac{1}{2}})_{n \in \mathbb{N}}$  et  $(g^{n+\frac{1}{2}})_{n \in \mathbb{N}}$ , supposons que  $\theta^n \in \mathbb{R}$ ,  $u^n \in \mathcal{X}$ ,  $s^n$  ont déjà été calculés. Le calcul de  $\theta^{n+1} \in \mathbb{R}$ ,  $u^{n+1} \in \mathcal{X}$ ,  $s^{n+1} \in \mathcal{V}$  et  $\lambda^{n+\frac{1}{2}} \in \mathcal{M}(\lambda^{n+\frac{1}{2}})$  est effectué en résolvant :*

$$\begin{aligned} m(\dot{s}^{n+\frac{1}{2}} + \dot{\theta}^{n+\frac{1}{2}} \Pi s^{n+\frac{1}{2}}, \Pi r^{n+\frac{1}{2}}) + b(\Pi r^{n+\frac{1}{2}}, \lambda^{n+\frac{1}{2}}) &= F^{n+\frac{1}{2}}(\Pi r^{n+\frac{1}{2}}), \\ m(\dot{s}^{n+\frac{1}{2}} + \dot{\theta}^{n+\frac{1}{2}} \Pi s^{n+\frac{1}{2}}, v) + a(u^{n+\frac{1}{2}}, v) + b(v, \lambda^{n+\frac{1}{2}}) &= F^{n+\frac{1}{2}}(v) \quad v \in \mathcal{X}, \\ b(s^{n+\frac{1}{2}}, \mu - \lambda^{n+\frac{1}{2}}) &\leq \langle \hat{\gamma}_{C,\theta}^n, \mu_{\nu_0} - \lambda_{\nu_0}^{n+\frac{1}{2}} \rangle, \quad \mu \in \mathcal{M}(\lambda^{n+\frac{1}{2}}), \end{aligned} \quad (2.32)$$

système qui est complété par la contrainte cinématique :

$$s^{n+\frac{1}{2}} = \dot{u}^{n+\frac{1}{2}} + \dot{\theta}^{n+\frac{1}{2}} \Pi r^{n+\frac{1}{2}}.$$

On peut alors montrer le théorème suivant.

**Théorème 12.** *L'algorithme 10 préserve l'énergie et le moment cinétique au sens où*

$$\mathcal{E}^{n+1} - \mathcal{E}^n = \Delta t \left( F^{n+\frac{1}{2}}(s^{n+\frac{1}{2}}) - b_{\tau_0}(s^{n+\frac{1}{2}}, \lambda^{n+\frac{1}{2}}) \right),$$

où l'énergie discrète est définie par

$$\mathcal{E}^n := \frac{1}{2}(m(s^n, s^n) + a(u^n, u^n)),$$

et

$$\mathcal{J}^{n+1} - \mathcal{J}^n = \Delta t \left( F^{n+\frac{1}{2}}(\Pi r^{n+\frac{1}{2}}) - b(\Pi r^{n+\frac{1}{2}}, \lambda^{n+\frac{1}{2}}) \right),$$

où le moment cinétique discret est défini par

$$\mathcal{J}^n := m(s^n, \Pi r^n).$$

On peut également construire une version simplifiée de l'algorithme 10, en remplaçant le terme  $r$  par  $x$ . Je renvoie aux tests de la section 2.2.5 à l'article [I] pour plus de détails sur cette approche.

## 2.2.4 Résolution des non-linéarités

Pour résoudre le système (2.32), on a recours à une méthode de résolution d'inégalités variationnelles introduite par Kunisch et Ito [15]. Dans le cadre considéré ici, cette méthode d'activation de contraintes consiste à chaque pas de temps en la réalisation d'une boucle interne de mise à jour des points de contact, jusqu'à ce que les conditions de l'inéquation de (2.32) soit satisfaite. Cette itération est effectuée en même temps qu'une boucle de Newton pour trouver la valeur de  $\dot{\theta}^{n+\frac{1}{2}}$ .

## 2.2.5 Quelques tests

Encore une fois, je ne reproduis qu'un seul test ici, en renvoyant à la publication elle-même pour d'autres résultats numériques. On considère un disque élastique entrant en contact avec un plan. Le contact a lieu avec un frottement. La figure 2.5 montre différents instants au cours de l'impact. Les figures 2.6 et 2.7 montrent l'évolution de différentes grandeurs au cours de la

simulation. Sur cet exemple, on observe des résultats quasi-identiques avec trois algorithmes : un schéma de résolution standard non-linéaire, l'algorithme 10 et une version complètement linéaire où l'on a fait la simplification  $r \approx x$ .

Le gain apporté par les deux schémas issus de la formulation co-rotationnelle se mesure en terme de nombre de sous-boucles de résolution effectuées à chaque pas de temps. Celui-ci est divisé par 3 ou 4 dans les schémas co-rotationnels.

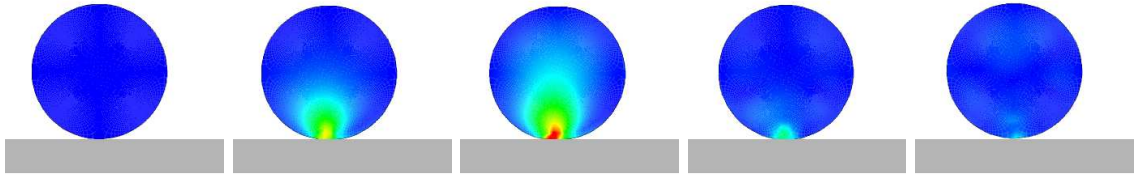


FIGURE 2.5 – Disque rebondissant sur un plan avec friction. Simulation à différents instants.

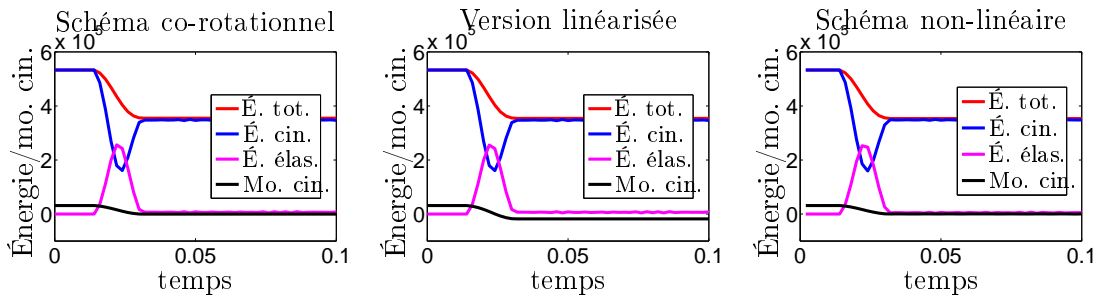


FIGURE 2.6 – Énergie et moment cinétique au cours de la simulation avec l'algorithme 10 (à gauche), sa version linéarisée (au milieu) et un schéma de résolution non-linéaire standard (à droite).

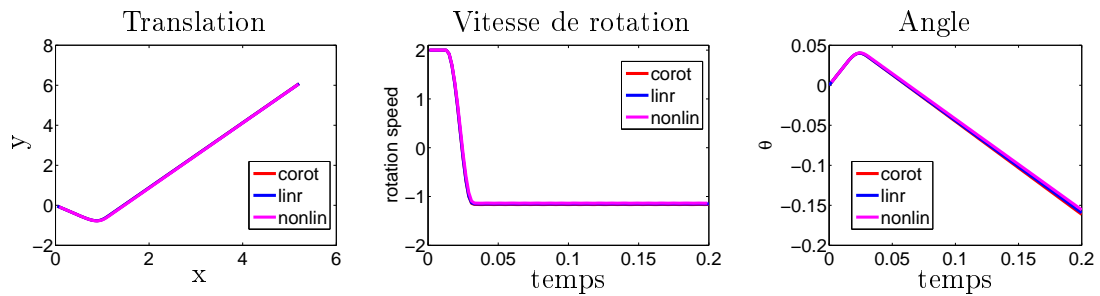


FIGURE 2.7 – À gauche : translation. Au milieu : vitesse de rotation. À droite : angle de rotation.



# Chapitre 3

## Algorithmes d'accélération par pré-calcul

**Résumé** : dans cette dernière partie, on présente deux méthodes permettant d'accélérer la résolution d'EDP grâce à une phase de pré-calcul.

La première méthode est dédiée à la résolution de l'équation de Schrödinger instationnaire décrivant l'interaction d'une particule quantique avec un laser. Le pré-calcul consiste à calculer un certain nombre d'opérateurs associés à l'évolution sur un pas de temps. Le paramètre d'indexation de cet ensemble est celui de la valeur du champ délivré par le laser. La méthode numérique résultante est dans certains contextes plus performante que la méthode du splitting d'opérateurs de Strang, habituellement utilisée.

La deuxième méthode concerne quant à elle un problème elliptique de minimisation sous contrainte d'inégalité, tel qu'on peut en rencontrer en simulation de contacts mécaniques. Le cadre de travail est celui des méthodes de base réduite, où l'espace de résolution est généré à partir d'échantillons obtenus par des calculs très précis de type éléments finis. Le cadre que l'on propose permet de résoudre le problème d'optimisation à l'aide d'outils standard et une méthode d'enrichissement est présentée pour assurer le bon conditionnement du système d'optimalité et donc la stabilité du calcul.

### Introduction

Cette dernière partie est consacrée à deux méthodes basées sur des techniques de pré-calcul. L'idée est ici de profiter d'une phase préliminaire éventuellement coûteuse en temps pour, le moment venu, résoudre rapidement un problème donné.

La première méthode fut en fait introduite par des chimistes pour traiter des problèmes de contrôle quantique. Comme je l'ai expliqué au chapitre 1 de la partie I de ce mémoire, ce domaine requiert de nombreuses résolutions de l'équation de Schrödinger. L'idée de la méthode consiste à pré-calculer un certain nombre de matrices intervenant dans ces résolutions.

La deuxième méthode concerne une méthode de base réduite dédiée aux inégalités variationnelles. Le travail a consisté à trouver une formulation de problèmes d'optimisation elliptiques sous contrainte d'inégalité adaptée aux bases réduites.

### 3.1 Méthode du *Toolkit*

Cette section décrit les résultats obtenus dans [L].

Les problèmes de contrôle optimal nécessitant souvent de nombreuses résolutions de l'équation d'évolution considérée, il est crucial de disposer de méthodes de résolution efficaces. La méthode dite du *Toolkit*, basée sur une discrétisation des valeurs du contrôle, a ainsi été introduite par des chimistes pour traiter des problèmes de contrôle bilinéaires. Cette technique correspond à la notion de quantification que l'on rencontre dans le domaine du traitement du signal. Dans un premier temps, on a proposé une première analyse de ce type de schéma [L]. Cette analyse est complétée par l'introduction de modification permettant d'augmenter l'ordre de convergence en temps et de rendre transparent l'effet de la quantification des valeurs du contrôle. Enfin, on a indiqué comment coupler cette méthode aux schémas monotones présentés à la section [J].

#### 3.1.1 Présentation de la méthode

La méthode dite du *Toolkit* fut introduite dans [29, 30] pour résoudre rapidement des équations de Schrödinger de la forme suivante :

$$\begin{cases} i\partial_t\psi(x, t) = H(\varepsilon(t), x)\psi(x, t), & x \in \mathbb{R}^\gamma \\ \psi(x, 0) = \psi_0(x), & x \in \mathbb{R}^\gamma, \end{cases} \quad (3.1)$$

où  $H(\varepsilon(t), x) = H_0 - \mu(x)\varepsilon(t)$  avec  $H_0 = -\Delta + V$ . Le terme  $\Delta$  est le Laplacien, qui est l'opérateur associé à l'énergie cinétique du système, et le terme  $V = V(x)$  représente le potentiel électrostatique dans lequel évolue le système. Je renvoie à l'introduction du chapitre 1 de la partie I pour plus d'informations sur les différents termes de cette équation.

L'idée de la méthode est de calculer dans une phase préliminaire des d'opérateurs de la forme  $\exp(-iH(\varepsilon, x)\Delta t)$  associés à un ensemble discret de valeurs de la variable  $\varepsilon$ . Concrètement, ceci revient à construire une famille de matrices indexée par des valeurs quantifiées du contrôle. Le travail qui suit ne traitant pas des questions de discrétisation en espace, les résultats et algorithmes seront présentés sous forme continue en espace.

Dans tout ce qui suit, on suppose le contrôle  $\varepsilon(t)$  borné en norme  $L^\infty$ .

$$\forall t \in [0, T], \quad \varepsilon(t) \in [\varepsilon_{\min}, \varepsilon_{\max}]. \quad (\mathcal{H})$$

La méthode repose sur une discrétisation des valeurs du contrôle selon :

$$\bar{\varepsilon}_\ell = \varepsilon_{\min} + \ell\Delta\varepsilon, \quad \ell = 0 \dots m, \quad (3.2)$$

où  $\Delta\varepsilon$  est le pas de discrétisation du champ et  $m = \frac{\varepsilon_{\max} - \varepsilon_{\min}}{\Delta\varepsilon}$ . L'algorithme est le suivant.

**Algorithme 11.** (*Méthode du toolkit*)

1. *Pré-calcul.* Calculer le toolkit, c'est-à-dire l'ensemble des opérateurs

$$S_\ell(\Delta t) \text{ for } \ell = 0, \dots, m,$$

où  $(S_\ell(t))_{t \in \mathbb{R}}$  est le semi-groupe à un paramètre généré par l'opérateur  $H_0 - \mu\bar{\varepsilon}_\ell$ , la suite  $(\varepsilon_\ell)_{\ell=0, \dots, m}$ , étant définie par (3.2).

2. Étant donné un contrôle  $\varepsilon$  satisfaisant l'hypothèse  $(\mathcal{H})$  et une condition initiale  $\psi_0^K = \psi_0$ , la suite d'approximations  $(\psi_j^K)_{j=0, \dots, N}$  de  $(\psi(t_j))_{j=0, \dots, N}$ , est obtenue itérativement suivant la boucle :

(a) Trouver :

$$\ell_j = \operatorname{argmin}_{\ell=1, \dots, m} \{|\varepsilon(t_{j+\frac{1}{2}}) - \bar{\varepsilon}_\ell|\},$$

(b) Calculer  $\psi_{j+1}^K = S_{\ell_j}(\Delta t)\psi_j^K$ .

Dans sa forme originale, la méthode ne stipulait pas de considérer la valeur du contrôle  $\varepsilon(t_{j+\frac{1}{2}})$  au point milieu. Ce choix permet cependant de gagner un ordre de convergence.

### 3.1.2 Analyse de la méthode

L'intérêt de l'algorithme apparaît dans le résultat suivant.

**Théorème 13.** *Soit  $\varepsilon \in W^{2,\infty}(0, T)$  et  $\psi$  la solution correspondante de (3.1). Soit également  $\psi^K$  l'approximation de  $\psi$  obtenue par l'algorithme 11 avec un pas de temps  $\Delta t > 0$  et de discrétisation du contrôle  $\Delta \varepsilon > 0$ . Il existe  $\lambda_1 > 0$ ,  $\lambda_2 > 0$ , indépendants de  $\|\varepsilon\|_{L^\infty(0, T)}$  tels que :*

$$\|\psi(T) - \psi^K(T)\|_{L^2} \leq \lambda_1 \Delta \varepsilon + \lambda_2 \Delta t^2.$$

De plus, il existe  $\nu_1 > 0$ ,  $\nu_2 > 0$  dépendants de  $\|\varepsilon\|_{W^{1,1}(0, T)}$  tels que :

$$\|\psi(T) - \psi^K(T)\|_{H^2} \leq \nu_1 \Delta \varepsilon + \nu_2 \Delta t^2.$$

Un avantage important de cette méthode est donc que l'approximation obtenue ne dépend pas de l'amplitude du contrôle considéré. Cette propriété n'est pas vérifiée dans d'autres algorithmes souvent utilisés, comme par exemple le *Splitting* de Strang.

### 3.1.3 Variantes et augmentation de l'ordre de convergence

L'analyse développée pour obtenir le théorème 13 a permis de construire deux variantes de l'algorithme 11 dont les ordres de convergences sont meilleurs.

#### Cas des champs faibles

Cette première variante permet d'augmenter l'ordre de convergence par rapport au paramètre  $\Delta t$ .

**Algorithme 12.** *((Méthode du toolkit améliorée dans le cas des champs faibles))*

1. *Pré-calcul.* Calculer le toolkit, c'est-à-dire l'ensemble d'opérateurs

$$S_\ell(\Delta t) \text{ for } \ell = 0, \dots, m,$$

où  $(S_\ell(t))_{t \in \mathbb{R}}$  est le semi-groupe à un paramètre généré par l'opérateur  $H_0 - \mu \bar{\varepsilon}_\ell$ , la suite  $(\bar{\varepsilon}_\ell)_{\ell=0, \dots, m}$ , étant définie par (3.2). Ajouter à cet ensemble les deux éléments spéciaux

$$\Omega = e^{\frac{1}{12}[H_0, \mu]\Delta t^3}, \Theta = e^{\frac{i}{24}\mu\Delta t^3}.$$

2. *Étant donné un contrôle  $\varepsilon$  satisfaisant l'hypothèse  $(\mathcal{H})$  et une condition initiale  $\psi_0^{IK} = \psi_0$ , la suite d'approximations  $(\psi_j^{IK})_{j=0, \dots, N}$  de  $(\psi(t_j))_{j=0, \dots, N}$ , est obtenue itérativement suivant la boucle :*

(a) Trouver :

$$\ell_j = \operatorname{argmin}_{\ell=1, \dots, m} \{|\varepsilon(t_{j+1/2}) - \bar{\varepsilon}_\ell|\},$$

(b) Calculer  $\alpha_j$  et  $\beta_j$  tels que :

$$\begin{aligned} \alpha_j &:= \frac{\varepsilon(t_{j+1}) - \varepsilon(t_j)}{\Delta t} = \dot{\varepsilon}(t_{j+\frac{1}{2}}) + \mathcal{O}(\Delta t^2), \\ \beta_j &:= \frac{\varepsilon(t_{j+1}) - 2\varepsilon(t_{j+\frac{1}{2}}) + \varepsilon(t_j)}{\Delta t^2} = \ddot{\varepsilon}(t_{j+\frac{1}{2}}) + \mathcal{O}(\Delta t^2). \end{aligned}$$

(c) Set  $\psi_{j+1}^{IK} = S_{\ell_j}(\Delta t)\Omega^{\alpha_j}\Theta^{\beta_j}\psi_j^{IK}$ .

Cet algorithme est analysé dans le théorème suivant.

**Théorème 14.** Soit  $\varepsilon \in W^{2,\infty}(0, T)$  et  $\psi$  la solution correspondante de (3.1). Soit également  $\psi^{IK}$  l'approximation de  $\psi$  obtenue par l'algorithme 12 avec un pas de temps  $\Delta t > 0$  et de discrétisation du contrôle  $\Delta\varepsilon > 0$ . Il existe  $\lambda'_1 > 0$ ,  $\lambda'_2 > 0$ , avec  $\lambda'_1$  indépendant de  $\|\varepsilon\|_{L^\infty(0, T)}$  tels que :

$$\|\psi(T) - \psi^K(T)\|_{L^2} \leq \lambda'_1 \Delta\varepsilon + \lambda'_2 \Delta t^3.$$

La dépendance de  $\lambda'_2$  à la norme du contrôle est révélée dans l'étude de convergence par le fait que l'erreur dépend du commutateur  $\left[[H_0, \mu], H_0 - \mu\bar{\varepsilon}\right]$ . Ce terme apparaît dans l'analyse par l'intermédiaire de la fonction  $\varphi_j(s) := S_j(t_{j+1} - s)\mu S_j(s - t_j)$  et de ses dérivées. Dans la preuve théorème 13, seule la première dérivée de  $\varphi_j$ , dont la norme est indépendante de  $\|\varepsilon\|_{L^\infty(0, T)}$  intervient. L'analyse menée dans la preuve du théorème 14 montre par contre que la dérivée seconde y joue un rôle. Or la norme de cette fonction dépendant de  $\|\varepsilon\|_{L^\infty(0, T)}$ .

### Cas des champs forts

Pour diminuer la taille du terme d'erreur associé à la quantification, on a proposé l'algorithme suivant.

**Algorithme 13.** (Méthode du toolkit améliorée dans le cas des champs forts.)

1. Pré-calcul. Calculer le toolkit, c'est-à-dire l'ensemble d'opérateurs

$$S_\ell(\Delta t) \text{ for } \ell = 0, \dots, m,$$

où  $(S_\ell(t))_{t \in \mathbb{R}}$  est le semi-groupe à un paramètre généré par l'opérateur  $H_0 - \mu\bar{\varepsilon}_\ell$ , la suite  $(\varepsilon_\ell)_{\ell=0, \dots, m}$ , étant définie par (3.2).

2. Étant donné un contrôle  $\varepsilon$  satisfaisant l'hypothèse  $(\mathcal{H})$  et une condition initiale  $\psi_0^{JK} = \psi_0$ , la suite d'approximations  $(\psi_j^{JK})_{j=0, \dots, N}$  de  $(\psi(t_j))_{j=0, \dots, N}$ , est obtenue itérativement suivant la boucle :

(a) Trouver  $\ell_j$  tel que :

$$\varepsilon(t_{j+1/2}) \in [\bar{\varepsilon}_{\ell_j}, \bar{\varepsilon}_{\ell_j+1}].$$

(b) Calculer  $\alpha_j$  et  $\beta_j$  tels que :

$$\begin{aligned} \alpha_j \bar{\varepsilon}_{\ell_j} + \beta_j \bar{\varepsilon}_{\ell_j+1} &= \varepsilon(t_{j+1/2}) \\ \alpha_j + \beta_j &= 1. \end{aligned}$$

(c) Calculer  $\psi_{j+1}^{JK} = S_{\ell_{j+1}}(\Delta t)^{\beta_j} S_{\ell_j}(\Delta t)^{\alpha_j} \psi_j^{JK}$ .

Dans cette dernière méthode, on doit calculer deux exponentiations à chaque pas de temps de la simulation. On peut cependant réduire ce coût à trois produits matrice-vecteur en pré-calculant les matrices permettant le passage entre deux bases diagonalisant deux éléments du toolkit liés à deux termes consécutifs. L'analyse de cette dernière méthode débouche sur le théorème suivant.

**Théorème 15.** Soit  $\varepsilon \in W^{3,\infty}(0,T)$  et  $\psi$  la solution correspondante de (3.1). Soit également  $\psi^{JK}$  l'approximation de  $\psi$  obtenue par l'algorithme 13 avec un pas de temps  $\Delta t > 0$  et de discrétisation du contrôle  $\Delta\varepsilon > 0$ . Il existe  $\lambda_1'' > 0$ ,  $\lambda_2'' > 0$ , indépendants de  $\|\varepsilon\|_{L^\infty(0,T)}$  tels que :

$$\|\psi(T) - \psi^{JK}(T)\|_{L^2} \leq \lambda_1'' \Delta\varepsilon \Delta t + \lambda_2'' \Delta t^2.$$

### 3.1.4 Tests numériques

Une série de tests numériques a ensuite été réalisée pour vérifier d'une part que les ordres obtenus dans les théorèmes étaient optimaux, d'autre part pour comparer les méthodes avec la méthode standard du splitting d'opérateurs de Strang. Toutes les expériences qui suivent ont été réalisées sur le modèle de la molécule *HCN* décrit plus précisément à la section 1.2.1 du chapitre 1 de la partie I et surtout dans l'article [A].

#### Ordres de convergence

Dans une première série de tests, on a évalué numériquement les ordres de convergence des différentes méthodes. Les résultats, tels qu'ils apparaissent sur la figure 3.1, montrent que les ordres des théorèmes correspondent à ceux obtenus empiriquement.

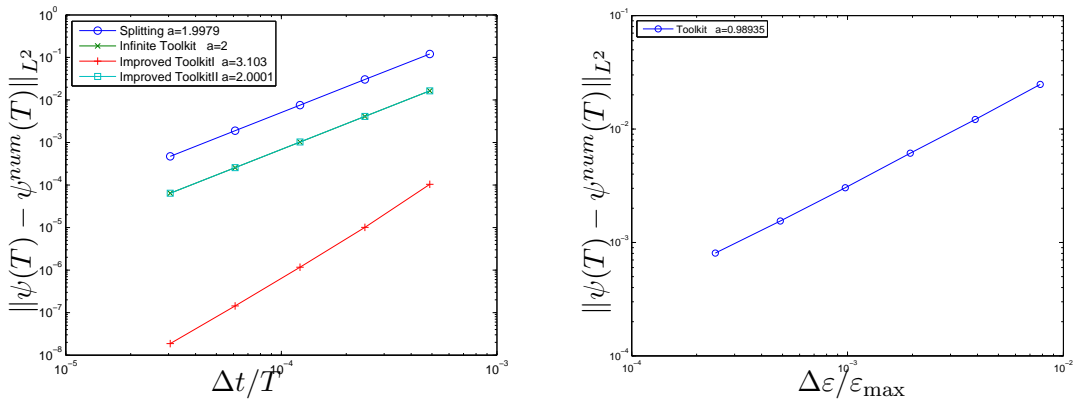


FIGURE 3.1 – À gauche : erreurs en fonction de  $\Delta t$ , lorsque  $\Delta\varepsilon = 0$  pour la méthode du Toolkit, la première variante correspondant à l'algorithme 12 et la seconde variante, correspondant à l'algorithme 13, testée avec  $\Delta\varepsilon = c\Delta t$ , avec  $c \in \mathbb{R}^+$ . Ici,  $\psi^{num}$  représente l'approximation obtenue par les différentes méthodes. Le coefficient  $a$  est la pente des droites, calculée par régression linéaire.

### Coût de calcul

Une seconde série de tests a permis de calculer effectivement le nombre de produits matrice-vecteur nécessaires pour obtenir une précision  $Tol$  fixée à l'avance. Le tableau 3.1 rend compte de ces valeurs. Il a été obtenu en divisant itérativement par deux les pas de temps et de discrétisation du contrôle jusqu'à ce que l'erreur soit inférieure à  $Tol$ . On a également testé une version quantifiée de la deuxième variante, où les coefficients  $\alpha_j$  et  $\beta_j$  sont également quantifiés (sur 100 valeurs) et où les produits  $S_{\ell_{j+1}}(\Delta t)^{\beta_j} S_{\ell_j}(\Delta t)^{\alpha_j}$  sont pré-calculés. Ces résultats montrent que

	$N = \frac{T}{\Delta t}$	Produits Matrice-Vecteur	$m = \frac{\varepsilon_{\max}}{\Delta \varepsilon}$
Splitting d'opérateurs	16384	32768	-
Toolkit (Algo. 11)	8192	8192	16384
Variante champs faibles (Algo. 12)	1024	2048	16384
Variante champs forts (Algo. 13)	4096	12288	16
Version quantifiée (Algo. 13)	4096	4096	6400

TABLE 3.1 – Valeurs numériques correspondant à une précision de  $Tol = 5.10^{-3}$ .

les méthodes de Toolkit donnent systématiquement des meilleurs résultats que le Splitting de Strang.

Il faut également noter que la seconde variante rend la quantification des valeurs du contrôle en quelque sorte transparente puisqu'un petit nombre d'éléments pré-calculés suffit à obtenir une bonne approximation.

### 3.1.5 Couplage avec les schémas monotones

*Cette section décrit les résultats obtenus dans [J].*

Un second travail a consisté à mettre en place un couplage entre la méthode du Toolkit présentée ci-dessus (dans la version correspondant à l'algorithme 11) et un algorithme monotone tel que présenté au chapitre 1 de la partie I. Un couplage naïf des algorithmes monotones et de la méthode du toolkit, peut par exemple consister en une simple projection à chaque itération du contrôle sur l'ensemble des valeurs quantifiées associées au Toolkit. Cette approche ne fonctionne généralement pas. La monotonie de l'algorithme n'est en effet pas garantie par ce procédé. La stratégie que j'ai proposée avec Gabriel Turinici et Mohammed Belhadj consiste à utiliser la factorisation présentée au chapitre 1 de la partie I pour en tirer un critère de choix d'une valeur optimale du contrôle à chaque pas de temps. Dans le cas d'une résolution à l'aide de la méthode du Toolkit, cette factorisation est de la forme :

$$J(\varepsilon') - J(\varepsilon) = \Delta t \sum_{j=1}^{N-1} (\varepsilon' - \varepsilon_{r_j}) K(\chi_j^\varepsilon, \psi_{j+1}^{\varepsilon'}), \quad (3.3)$$

pour une certaine fonction  $K$ . Dans cette équation,  $J$  est la fonctionnelle à minimiser,  $\varepsilon$  est un contrôle connu, à valeurs quantifiées (permettant l'utilisation de la méthode du Toolkit) et  $\varepsilon'$  est un contrôle à valeurs arbitraires, qui reste à déterminer pour optimiser  $J(\varepsilon')$ .

On peut maintenant expliciter l'algorithme. Dans la formule (3.3),  $\varepsilon$  et  $\varepsilon'$  correspondent respectivement à deux contrôles successifs  $\varepsilon^k$  et  $\varepsilon^{k+1}$ .

**Algorithme 14.** Supposons qu'on dispose à l'étape  $k$  d'un contrôle  $\varepsilon^k = (\varepsilon_{\ell_j^k})_{j=1,\dots,N-1}$ , où  $\ell_j^k$  est l'indice correspondant à la valeur quantifiée du contrôle utilisée au pas de temps  $j$ . Le calcul de  $\varepsilon^{k+1}$ , ou, de manière équivalente, de  $(r_j^{k+1})_{j=1,\dots,N-1}$  est effectué séquentiellement par rapport à  $j$  en 4 étapes.

- Calculer une valeur  $\tilde{\varepsilon}^{k+1}$  du contrôle minimisant le terme courant de la factorisation (3.3),
- Projeter cette valeur sur le Toolkit. On note  $\varepsilon_{\tilde{r}_j^{k+1}}$  cette valeur.
- Calculer  $(\varepsilon_{\tilde{r}_j^{k+1}} - \varepsilon_{r_j^k})K(\chi_j^\varepsilon, \psi_{j+1}^{\varepsilon'})$ .
- Si cette quantité est positive, garder l'ancienne valeur du contrôle en posant  $r_j^{k+1} = r_j^k$ . Sinon mettre à jour cette valeur en posant  $r_j^{k+1} = \tilde{r}_j^{k+1}$ .

Avec la méthode du Toolkit, seule la première des 4 étapes est coûteuse<sup>6</sup>. Ce résultat s'est traduit dans les tests numériques par le fait que cette méthode permet de reproduire à moindre coût les performances d'autres algorithmes, en particulier des algorithmes monotones.

## 3.2 Méthode de base réduite pour les inégalités variationnelles

Les méthodes de base réduite [23, 11, 20] font l'objet de nombreux travaux depuis une dizaine d'années mais n'avaient pas été utilisées dans des problèmes contenant des contraintes d'inégalité. Dans un travail en cours mené en collaboration avec Barbara Wohlmuth et Bernard Haasdonk, j'ai construit un cadre permettant un traitement efficace de tels problèmes.

### 3.2.1 Problème considéré

Le problème que l'on considère ici est elliptique. Son cadre est le suivant. On considère deux espaces de Hilbert  $V, W$  de dimension finie. En pratique, ces ensembles seront des espaces d'éléments finis. Les produits scalaires correspondant sont notés  $\langle \cdot, \cdot \rangle_V$ ,  $\langle \cdot, \cdot \rangle_W$  et  $\|\cdot\|_V$ ,  $\|\cdot\|_W$ . On se donne ensuite un ensemble  $X \subset V$  convexe fermé non vide et un cône convexe  $M \subset W$ . Le modèle auquel on s'intéresse est décrit à l'aide d'un vecteur de paramètres  $\boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^p$ ,  $p \in \mathbb{N}$ . Pour construire le système d'équations, on introduit une forme bilinéaire  $a(\cdot, \cdot; \boldsymbol{\mu})$  continue et elliptique sur  $V \times V$ . Les constantes de continuités et d'ellipticité sont notées respectivement  $\gamma_a(\boldsymbol{\mu})$  et  $\alpha(\boldsymbol{\mu})$ . On considère également une seconde forme bilinéaire continue  $b(\cdot, \cdot)$  cette fois-ci définie sur  $V \times W$ . On note  $\gamma_b > 0$  sa constante de continuité. On la suppose de plus inf-sup stable c'est-à-dire qu'il existe  $\beta > 0$  tel que

$$\inf_{\eta \in W} \sup_{v \in V} b(v, \eta) / (\|v\|_V \|\eta\|_W) \geq \beta > 0.$$

Enfin, on se donne deux formes linéaires continues  $f(\cdot; \boldsymbol{\mu}) \in V'$  et  $g(\cdot; \boldsymbol{\mu}) \in W'$ . Les constantes de continuités  $\gamma_a(\boldsymbol{\mu})$ ,  $\alpha(\boldsymbol{\mu})$  ainsi celles de  $f$  et  $g$  sont supposées uniformément bornées inférieurement par rapport au vecteur de paramètres  $\boldsymbol{\mu}$ .

Étant donné  $\boldsymbol{\mu} \in \mathcal{P}$ , le problème général que l'on envisage dans ce chapitre consiste à trouver un couple  $(u(\boldsymbol{\mu}), \lambda(\boldsymbol{\mu})) \in V \times M$  vérifiant :

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) + b(v, \lambda(\boldsymbol{\mu})) = f(v; \boldsymbol{\mu}), \quad v \in V \tag{3.4}$$

$$b(u(\boldsymbol{\mu}), \eta - \lambda(\boldsymbol{\mu})) \leq g(\eta - \lambda(\boldsymbol{\mu}); \boldsymbol{\mu}), \quad \eta \in M. \tag{3.5}$$

L'existence et l'unicité de la solution sont bien sûr garanties aux vues des hypothèses considérées. Ce type de formulation s'applique par exemple à des problèmes de contact statique.

6. Mais cette étape peut-être parallélisée, comme l'explique le chapitre 1 de la partie II.

### 3.2.2 Trois formulations adaptées aux bases réduites

Plutôt que de considérer une approche par éléments finis pour résoudre (3.4–3.5), on souhaite profiter d'une phase de pré-calcul permettant de construire une base de résolution de petite dimension. La première difficulté consiste à garantir la stabilité inf-sup du problème réduit ainsi obtenu. Dans ce but, on introduit un opérateur  $B : W \rightarrow V$  défini par application du théorème de représentation de Riesz selon la formule :

$$\langle v, B\eta \rangle_V = b(v, \eta), \quad \forall v \in V, \eta \in W.$$

En suivant la démarche proposée dans [24], on utilise cet opérateur pour construire l'espace réduit de la manière suivante :

- Soit  $S = \{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_N\} \subset \mathcal{P}$  un échantillon fini de paramètres et les couples de solutions  $(u(\boldsymbol{\mu}_i), \lambda(\boldsymbol{\mu}_i))_{i=1, \dots, N}$  correspondants.
- On définit  $W_N := \text{span}\{\lambda(\boldsymbol{\mu}_i)\}_{i=1}^N \subset W$  comme espace réduit de dimension  $N^W := \dim W_N$  et de base  $\{\xi_i\}_{i=1}^{N^W} \subset \{\lambda(\boldsymbol{\mu}_i)\}_{i=1}^N$ .
- L'espace réduit est défini par

$$V_N := \text{span}\{u(\boldsymbol{\mu}_i), B\lambda(\boldsymbol{\mu}_i)\} \subset V \quad (3.6)$$

On note sa dimension  $N^V := \dim V_N$  et une de ses bases orthonormales  $\{\varphi_i\}_{i=1}^{N^V}$ .

- On définit alors le cône

$$M_N := \left\{ \sum_{i=1}^{N^W} \alpha_i \lambda(\boldsymbol{\mu}_i) \mid \alpha_i \geq 0 \right\}$$

qui est également un sous ensemble de  $M$  convexe et fermé.

Ici, l'indice  $N$  n'est pas forcément égal à la dimension. Il représente le nombre d'éléments considérés pour construire la base réduite. L'enrichissement de la base, spécifié par (3.6) joue dans la suite un rôle aussi bien théorique que pratique. Il permet en effet de garantir l'existence et l'unicité d'une solution au problème réduit qui va être construit (voir le théorème 16) et garantit le bon conditionnement de ce dernier (voir les tests numériques présentés à la section 3.2.5).

Notons également que l'on peut choisir la base  $(\xi_i)_{i=1}^{N^W}$  de  $W_N$  de telle sorte que  $b$  soit diagonale, ce qui se traduit algébriquement par

$$b(\varphi_i, \xi_j) = \delta_{ij}. \quad (3.7)$$

Cette propriété des bases  $(\psi_i)_{i=1}^{N^V}$  et  $(\xi_i)_{i=1}^{N^W}$  est parfois appelée  $b$ -bi-orthogonalité.

Notre but est maintenant de calculer rapidement une solution réduite  $u_N(\boldsymbol{\mu}) \in V_N, \lambda_N(\boldsymbol{\mu}) \in W_N$  en résolvant, pour  $\boldsymbol{\mu} \in \mathcal{P}$ , le problème :

$$a(u_N(\boldsymbol{\mu}), v_N; \boldsymbol{\mu}) + b(v_N, \lambda_N(\boldsymbol{\mu})) = f(v_N; \boldsymbol{\mu}), \quad v_N \in V_N \quad (3.8)$$

$$b(u_N(\boldsymbol{\mu}), \eta_N - \lambda_N(\boldsymbol{\mu})) \leq g(\eta_N - \lambda_N(\boldsymbol{\mu}); \boldsymbol{\mu}), \quad \eta_N \in M_N \quad (3.9)$$

Dans ce problème, les solutions ne sont plus recherchées dans les espaces de grandes dimensions  $V$  et  $W$ , mais dans des espaces potentiellement plus petits. En introduisant les matrices et vecteurs

$$\begin{aligned} \underline{A}_N(\boldsymbol{\mu}) &:= (a(\varphi_j, \varphi_i; \boldsymbol{\mu}))_{i,j=1}^{N^V} \in \mathbb{R}^{N^V \times N^V}, \\ \underline{B}_N &:= (b(\varphi_i, \xi_j))_{i,j=1}^{N^V, N^W} \in \mathbb{R}^{N^V \times N^W}, \\ \underline{f}_N(\boldsymbol{\mu}) &:= (f(\varphi_i; \boldsymbol{\mu}))_{i=1}^{N^V} \in \mathbb{R}^{N^V}, \\ \underline{g}_N(\boldsymbol{\mu}) &:= (g(\xi_i; \boldsymbol{\mu}))_{i=1}^{N^W} \in \mathbb{R}^{N^W}. \end{aligned}$$

on peut aussi écrire ce problème sous la forme discrète décrite dans le lemme suivant.

**Lemme 7.** La solution  $(u_N(\boldsymbol{\mu}), \lambda_N(\boldsymbol{\mu}))$  du problème (3.8–3.9) avec  $u_N(\boldsymbol{\mu}) := \sum_{i=1}^{N^V} u_{N,i} \varphi_i$  et  $\lambda_N(\boldsymbol{\mu}) := \sum_{i=1}^{N^W} \lambda_{N,i} \xi_i$  où  $\underline{u}_N := (u_{N,i})_{i=1}^{N^V} \in \mathbb{R}^{N^V}$  and  $\underline{\lambda}_N := (\lambda_{N,i})_{i=1}^{N^W} \in \mathbb{R}^{N^W}$  est également l'unique solution du système

$$\begin{aligned} \underline{A}_N(\boldsymbol{\mu})\underline{u}_N(\boldsymbol{\mu}) + \underline{\lambda}_N(\boldsymbol{\mu}) &= \underline{f}_N(\boldsymbol{\mu}) \\ \underline{\lambda}_N(\boldsymbol{\mu}) &\geq 0 \\ \underline{g}_N(\boldsymbol{\mu}) - \underline{B}_N^T \underline{u}_N(\boldsymbol{\mu}) &\geq 0 \\ \underline{\lambda}_N(\boldsymbol{\mu})^T (\underline{g}_N(\boldsymbol{\mu}) - \underline{B}_N^T \underline{u}_N(\boldsymbol{\mu})) &= 0. \end{aligned}$$

Enfin, dans le cas fréquent où  $a$  est symétrique, le vecteur  $u_N(\boldsymbol{\mu})$  est l'unique solution du problème de programmation linéaire :

$$\begin{aligned} \min \quad & \frac{1}{2} \underline{u}^T A_N(\boldsymbol{\mu}) \underline{u}_N - f_N(\boldsymbol{\mu})^T \underline{u}_N \\ \text{t.q.} \quad & \underline{B}_N^T \underline{u}_N(\boldsymbol{\mu}) \leq \underline{g}_N(\boldsymbol{\mu}). \end{aligned}$$

Dans ce cas, la composante duale  $\underline{\lambda}_N(\boldsymbol{\mu})$  est l'unique solution de l'équation

$$\underline{A}_N \underline{u}_N + \underline{B}_N \underline{\lambda}_N = \underline{f}_N$$

avec  $(\underline{\lambda}_N)_i = 0$  si  $(\underline{B}_N^T \underline{u}_N - \underline{g}_N)_i < 0$ . Cette dernière formulation permet de résoudre le problème réduit par des outils standards d'optimisation quadratique. L'équivalence entre ses trois formulations est assez facile à obtenir. Plus délicate est la preuve du résultat suivant.

**Théorème 16.** En gardant les notations précédentes, on a les deux assertions suivantes.

- i) Le problème réduit (3.8–3.9) possède une unique solution  $(u_N(\boldsymbol{\mu}), \lambda_N(\boldsymbol{\mu})) \in V_N \times W_N$ .
- ii) Si pour un certain  $\boldsymbol{\mu} \in \mathcal{P}$ ,  $u(\boldsymbol{\mu}) \in V_N$  et  $\lambda(\boldsymbol{\mu}) \in M_N$ , alors  $u_N(\boldsymbol{\mu}) = u(\boldsymbol{\mu})$  et  $\lambda_N(\boldsymbol{\mu}) = \lambda(\boldsymbol{\mu})$ .

Dans la preuve de l'existence et de l'unicité, le fait d'avoir enrichi l'échantillon joue un rôle fondamental. Cette technique permet en effet d'obtenir une constante inf-sup pour la forme réduite  $B_N$ . De plus, cette constante est indépendante de  $N$ , ce qui garantit un bon conditionnement du problème.

La deuxième assertion du théorème est généralement appelée *propriété de reproduction des solutions*, puisqu'elle donne lieu au recouvrement des éléments de l'échantillon initial.

### 3.2.3 Pré-calcul et résolution en ligne

Si l'on suppose que  $a$ ,  $f$  et  $g$  peuvent être décomposées selon des familles  $(a^q(\cdot, \cdot))_{q=1, \dots, Q_a}$ ,  $(f^q(\cdot) \in V')$  et  $(g^q(\cdot))_{q=1, \dots, Q_g}$  selon

$$\begin{aligned} a(u, v; \boldsymbol{\mu}) &= \sum_{q=1}^{Q_a} \theta_a^q(\boldsymbol{\mu}) a^q(u, v), \\ f(v; \boldsymbol{\mu}) &= \sum_{q=1}^{Q_f} \theta_f^q(\boldsymbol{\mu}) f^q(v), \\ g(\eta; \boldsymbol{\mu}) &= \sum_{q=1}^{Q_g} \theta_g^q(\boldsymbol{\mu}) g^q(\eta). \end{aligned}$$

Le pré-calcul consiste en l'assemblage des matrices et vecteurs suivant :

$$\begin{aligned}\underline{A}_N^q &:= (a^q(\varphi_j, \varphi_i))_{i,j=1}^{N^V} \in \mathbb{R}^{N^V \times N^V}, \quad q = 1, \dots, Q_a \\ \underline{B}_N &:= (b(\varphi_i, \xi_j))_{i,j=1}^{N^V, N^W} \in \mathbb{R}^{N^V \times N^W}, \\ \underline{f}_N^q &:= (f^q(\varphi_i))_{i=1}^{N^V} \in \mathbb{R}^{N^V}, \quad q = 1, \dots, Q_f \\ \underline{g}_N^q &:= (g^q(\xi_i))_{i=1}^{N^W} \in \mathbb{R}^{N^W}, \quad q = 1, \dots, Q_g.\end{aligned}$$

et le calcul en ligne se compose de l'assemblage (peu coûteux dès lors que  $Q_a$ ,  $Q_f$ ,  $Q_g$  ne sont pas trop grands) des matrices et vecteurs suivant :

$$\begin{aligned}\underline{A}_N(\boldsymbol{\mu}) &= \sum_{q=1}^{Q_a} \theta_a^q(\boldsymbol{\mu}) \underline{A}_N^q \\ \underline{f}_N(\boldsymbol{\mu}) &= \sum_{q=1}^{Q_f} \theta_f^q(\boldsymbol{\mu}) \underline{f}_N^q \\ \underline{g}_N(\boldsymbol{\mu}) &= \sum_{q=1}^{Q_g} \theta_g^q(\boldsymbol{\mu}) \underline{g}_N^q\end{aligned}$$

ainsi que de la résolution d'une des versions du problème réduit présentée à la section précédente. Cette étape est également peu coûteuse car indépendante des dimensions de  $V$  et  $W$ .

### 3.2.4 Méthodes numériques utilisées

Pour la mise en œuvre pratique de la démarche décrite dans les paragraphes précédents, on utilise trois méthodes numériques. Les résolutions du problème initial qui permettent de construire les échantillons  $(u(\boldsymbol{\mu}_i), \lambda(\boldsymbol{\mu}_i))_{i=1, \dots, N}$  sont effectuées à l'aide d'un algorithme d'activation de contraintes dans une version correspondant à celle présentée dans [15]. L'extraction de la base réduite  $(\phi_i)_{i=1, \dots, N^V}$  se fait par un algorithme de décomposition en valeur singulière. Celui-ci permet de plus de sélectionner le nombre de composantes principales que l'on souhaite considérer, ce qui fait qu'en pratique on ne travaille pas forcément avec une base de  $V_N$ , mais juste avec une famille orthonormée plus petite. Enfin, pour résoudre le problème réduit, on applique à sa troisième formulation<sup>7</sup> un algorithme d'optimisation quadratique standard.

### 3.2.5 Test numériques

Je présente ici quelques exemples d'application de la démarche précédente basés sur un modèle simple de chaînette élastique.

#### Le modèle

On considère donc un problème de dimension 1, consistant en la simulation d'un fil élastique suspendu au dessus d'un obstacle. Le domaine d'espace  $\Omega$  est le segment  $[0, 1]$ , discrétisé uniformément par selon un pas d'espace  $\Delta x := 1/K$  avec  $K \in \mathbb{N}$ . L'espace de  $V$  est celui des éléments finis  $P^1$  :

$$V := \{v \in H_0^1(\Omega) \mid v|_{[k\Delta x, (k+1)\Delta x]} \in P_1, k = 0, \dots, K-1\}$$

---

7. on ne considère dans la suite que le cas où  $a$  est symétrique.

qui est donc de dimension  $H^V = K - 1$ .

L'espace dual est obtenu en utilisant une base duale d'éléments finis de  $W = V'$ , ce qui permet de travailler avec une matrice  $B$  vérifiant la propriété (3.7).

### Exemple d'application

Dans un premier test, les paramètres sont choisis dans l'ensemble  $\mathcal{P} := [-1, -0.2] \times [5 \cdot 10^{-3}, 10^{-2}] \subset \mathbb{R}^2$  et le vecteur des paramètres s'écrit sous la forme  $\boldsymbol{\mu} = (\mu_1, \mu_2)$ . Les formes bilinéaires  $a$  et  $b$  sont données par :

$$\begin{aligned} a(u, v; \boldsymbol{\mu}) &= \int_{\Omega} \nu(\boldsymbol{\mu}) \nabla u(x) \cdot \nabla v(x) dx, \quad v, u \in V \\ b(u, \eta) &= \eta(u), \quad u \in V, \eta \in W. \end{aligned}$$

où  $\nu(\boldsymbol{\mu}) = \mu_2$  caractérise l'élasticité de la chaînette. La matrice  $B$  est l'identité entre les  $N^V$  premiers éléments des bases de  $V_N$  et  $W_N$ . Les fonctions  $f$  et  $g$  sont définies par

$$f(v; \boldsymbol{\mu}) = \int_{\Omega} \gamma(x; \boldsymbol{\mu}) v(x) dx, \quad v \in V$$

où  $\gamma(x; \boldsymbol{\mu}) := \gamma_0 = 1$  correspond à la gravité, et par

$$g(\eta; \boldsymbol{\mu}) := \int_{\Omega} h(x; \boldsymbol{\mu}) J(\eta)(x) dx$$

qui représente l'obstacle, défini dans cet exemple par

$$h(x; \boldsymbol{\mu}) = -4(\sin(\pi x) + \mu_1 \sin(3\pi x)) - 5.$$

J'ai ici noté  $J$  l'injection canonique  $W \rightarrow L^2(\Omega)$ . Des exemples de solutions du problème initial sont représentés sur la figure 3.2. On choisit seulement 4 composantes principales, représentées sur la figure 3.3, pour décrire l'espace réduit  $V_N$ . Bien que la base réduite soit de petite dimension, on approche correctement la solution obtenue par éléments finis comme le montre la figure 3.4.

### Mesure de l'accélération

Dans un second exemple, on cherche à quantifier l'efficacité de notre méthode. On garde pour le modèle précédent, en choisissant toutefois une fonction  $h$  la fonction affine  $h(x) = 5x - 10$  et un ensemble de paramètres de dimension 1  $\mathcal{P} := [10^2, 10^3] \subset \mathbb{R}$  associé à l'élasticité  $\nu(\boldsymbol{\mu}) = \mu$ . Dans un premier exemple, on calcule l'erreur d'approximation entre l'approche par base réduite et la méthode d'éléments finis lorsque  $N$  et  $\mu$  varient. Par simplicité, on ne représente que les solutions primales. Les résultats sont décrits par la figure 3.5. On s'intéresse maintenant aux bénéfices apportés par l'enrichissement de la base, stipulé par (3.6). Pour ce faire, on calcule la valeur des constantes inf-sup  $\beta_N$  avec et sans cet enrichissement. Les résultats sont reportés dans la table suivante.

$N$	$\beta_N$ avec enrichissement	$\log_{10}(\beta_N)$ sans enrichissement
5	1.000000	-2.566240
10	1.000000	-5.647559
15	1.000000	-8.562339
20	1.000000	-11.409922
25	1.000000	-15.048048

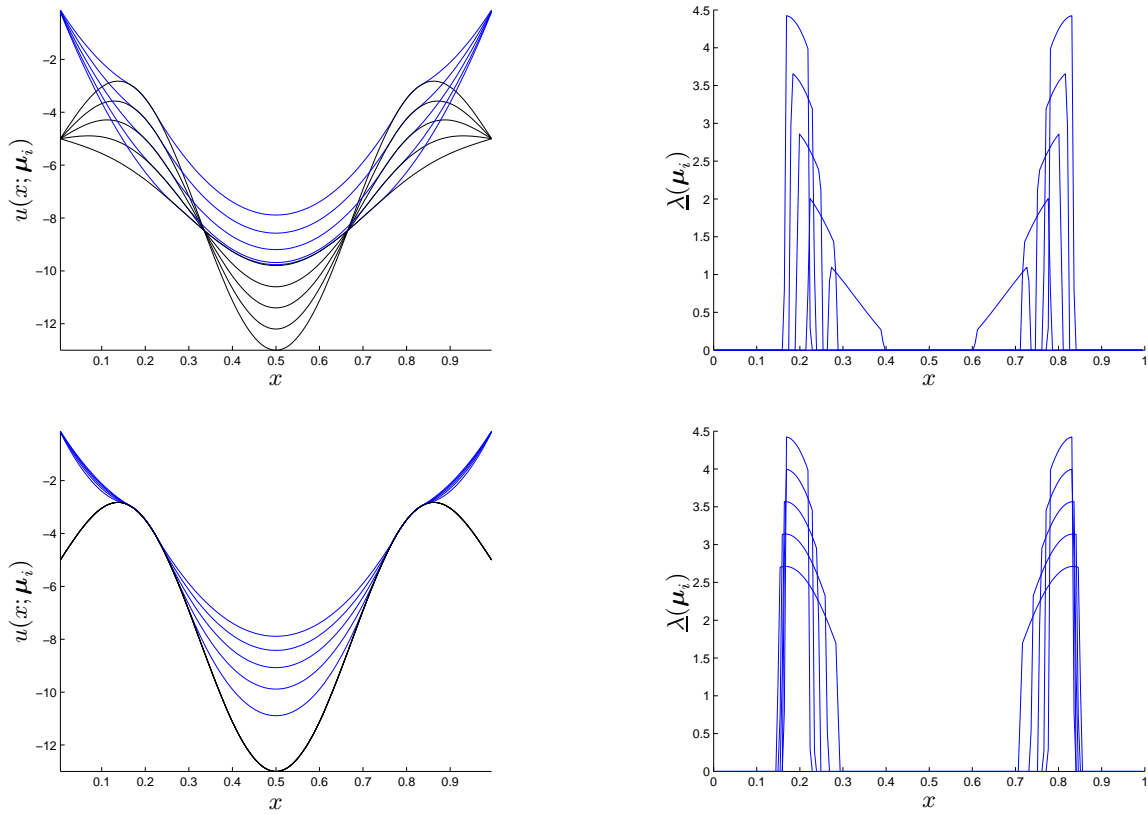


FIGURE 3.2 – Échantillons correspondant au problème de la chaînette. Les paramètres considérés sont associés à l'obstacle (figures du haut) et à l'élasticité (figure du bas). Les solutions primales et duales sont représentées respectivement sur les figures de droite et de gauche. Les échantillons sont représentés en bleu et les obstacles en noir.

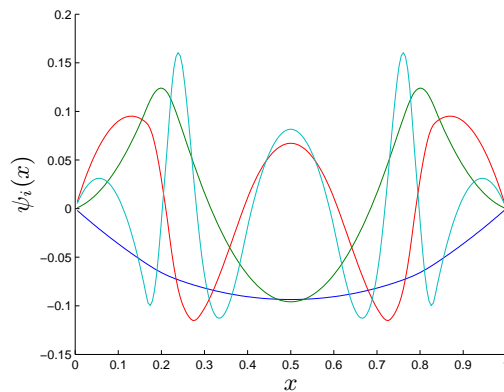


FIGURE 3.3 – Base réduite  $\{\psi_i\}_{i=1}^{H^V}$  extraite de  $V_N$ .

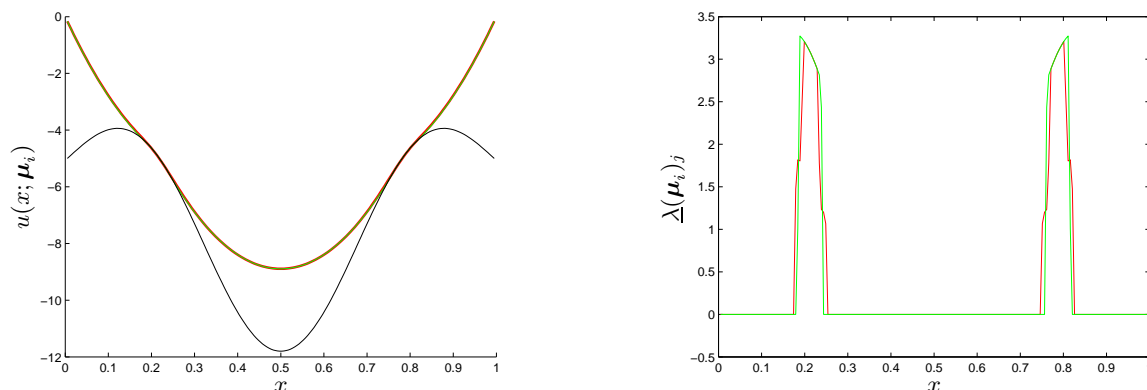


FIGURE 3.4 – Solutions associées au paramètre  $\boldsymbol{\mu} = (-0.7, 0.01)$  obtenues par éléments finis et par base réduite. À gauche : solutions primales. À droite : solutions duales. Les solutions éléments finis sont représentées en vert et celles associées à la base réduite en rouge.

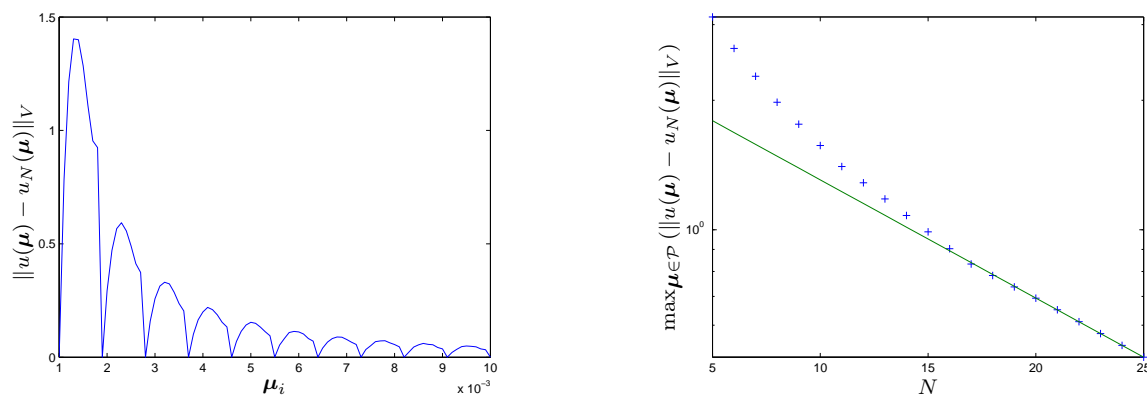


FIGURE 3.5 – Valeurs numériques de  $\|u(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})\|_V$ . À gauche : le paramètre  $\boldsymbol{\mu}$  décrit une grille fine de  $\mathcal{P}$ . Comme prévu dans le théorème 16, l'erreur s'annule lorsque  $\boldsymbol{\mu}$  prend des valeurs de l'échantillon. À droite sont représentées les valeurs de  $\max_{\boldsymbol{\mu} \in \mathcal{P}} (\|u(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})\|_V)$  en fonction de différentes valeurs de  $N$ . Les valeurs numériques sont représentées par des croix bleues et une régression linéaire effectuée sur les 5 dernières valeurs est représentée en vert.

La constance de  $\beta_N$  donne lieu à un meilleur conditionnement du système réduit qui se traduit numériquement par une diminution du temps de résolution. Ceci apparaît très clairement sur la figure 3.6 où l'on mesure les temps de calcul et nombre d'itérations requis pour la résolution en ligne du système réduit. Enfin, pour mesurer l'apport de l'approche par base réduite proposée

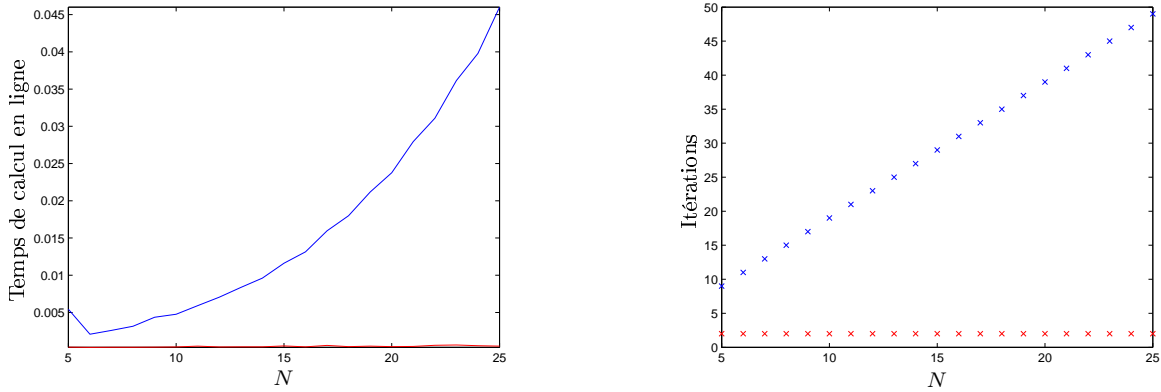


FIGURE 3.6 – Effet de l'enrichissement. À gauche : temps de calcul de la phase en ligne. À droite : nombre d'itérations de l'algorithme d'optimisation quadratique nécessaire à l'obtention d'une solution du problème réduit.

face à la méthode des éléments finis, on a tracé sur la figure 3.7 les temps de calculs nécessaires à l'obtention de la figure 3.5, où les deux solutions, par base réduite et par éléments finis devaient être calculées.

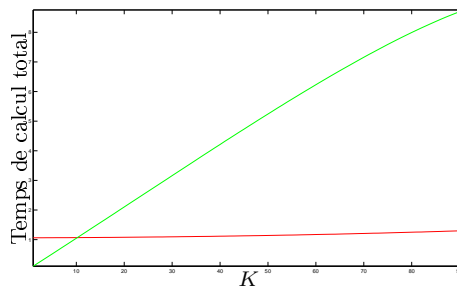


FIGURE 3.7 – Temps de calcul de la méthode des éléments finis et de la méthode par bases réduites nécessaires à l'obtention de la figure 3.5. La ligne rouge représente le temps de calcul des solutions réduites et la ligne verte celui des solutions éléments finis. Le nombre de résolutions est noté  $K$ .

Troisième partie

Perspectives



J'indique dans cette dernière partie les différents travaux qui pourraient prolonger les résultats décrits dans ce mémoire.

## Les algorithmes monotones

Les algorithmes monotones tels qu'ils ont été présentés s'appliquent à des équations d'évolution linéaires et des fonctionnelles de coût concaves par rapport à l'état. Une extension naturelle de l'approche consisterait donc à adapter la méthode à des équations non-linéaires et à des fonctionnelles convexes. Une possibilité pour effectuer ce pas serait de considérer une factorisation implicite par rapport à l'adjoint et à introduire une sous-boucle de résolution à chaque étape de l'algorithme. Des tests effectués pendant le stage de M2 de Mehdi Benamouche ont montré la pertinence de cette approche. Avec Karine Beauchard, nous étudions actuellement cette méthode.

Une seconde piste à explorer consisterait à dégager des critères généraux de convergence des schémas. Dans l'état actuel des choses, celle-ci repose sur la compacité des points critiques et l'analyticité de la fonctionnelle de coût. Ces hypothèses peuvent certainement être affaiblies.

## Transport optimal

Comme indiqué au chapitre 2 de la partie 1, les indicateurs d'appariement locaux ont été utilisés pour mettre en place un algorithme de calcul de plans de transport optimaux. La méthode suivie est assez simpliste, puisqu'elle consiste à appliquer séquentiellement les indicateurs sur tous les sous-ensembles de points du problème. Une démarche plus astucieuse doit être mise en place pour ordonner et réduire le nombre de calculs d'indicateur. Cette démarche pourrait par exemple reposer sur un classement des sous-ensembles suivant des propriétés liées aux distances respectives entre leurs points.

Évidemment, il serait également souhaitable de mettre en place une stratégie de calcul pour les coûts concaves en dimension supérieure.

## Parallélisation en temps

Les algorithmes de parallélisation présentés au chapitre 1 de la partie 2 devraient pouvoir être interprétés en terme de solveurs basés sur un pré-conditionnement des systèmes d'optimalités considérés. Des résultats de ce type ont par exemple d'ores-et-déjà obtenus par Sarkis *et al* dans [22]. Une telle analyse donnerait sans doute lieu à des améliorations de la méthode.

Les deux méthodes présentées traitent d'équations linéaires hyperboliques et paraboliques. Une autre piste consisterait donc à adapter l'approche à des équations non-linéaires. Un tel travail commencerait sans doute par une nouvelle définition de la trajectoire de référence utilisée pour construire le système de cibles intermédiaires. Cette piste est actuellement étudiée par Kamel Riahi dans le cadre d'une thèse que je co-encadre avec Yvon Maday.

## Formulation co-rotationnelle

Le premier travail à mener pour approfondir la compréhension des schémas proposés au chapitre 2 de la partie 2 consiste en une analyse d'erreur. Celle-ci pourrait partir du résultat d'existence et de régularité de la solution du système non discrétisé obtenu dans [13].

Une seconde étape pourrait en être l'adaptation à des modèles d'interactions fluide-structure.

Des tests ont par exemple été effectués dans ce sens en collaboration avec Benjamin Mauroy dans le cadre de la simulation de globules rouges.

### **Usage du pré-calcul**

La méthode de base réduite proposée au chapitre 3 de la partie 2 a uniquement été testée sur l'exemple simple d'une chaînette. Des études complémentaires pourraient donc être menées sur des exemples plus complexes, comme les inégalités variationnelles que l'on peut rencontrer en élasto-dynamique ou encore en finance, dans le cas de la simulation d'options américaines.

# Bibliographie

- [1] T. Aubin. Un théorème de compacité. *Comptes-Rendus Mathématiques de l'Académie des Sciences*, 256 :5042–5044, 1963.
- [2] J. M. Ball, J. E. Marsden, and M. Slemrod. Controllability for distributed bilinear systems. *SIAM J. Cont. and Opt.*, 20(4) :575–597, 1982.
- [3] J.-D. Benamou and Y. Brenier. A computational fluid mechanics solution to the Monge–Kantorovich mass transfer problem. *Numerische Mathematik*, 84(3) :375–393, 2000.
- [4] D. P. Bertsekas. Auction algorithms. In *Encyclopedia of Optimization*, pages 128–132. 2009.
- [5] M. A. Biot and J. E. Romain. Mechanics of incremental deformations : Theory of elasticity and viscoelasticity of initially stressed solids and fluids. *Physics Today*, 18(11) :68–72, 1965.
- [6] H. G. Bock and K. J. Plitt. Multiple shooting algorithm for direct solution of optimal control problems. In D. Forsyth, editor, *Proc. of 9th IFAC World Congress*. Springer, 1984.
- [7] R. Burkard, M. Dell’Amico, and S. Martello. *Assignment Problems*. SIAM, 2008.
- [8] V. Chawla and T. Laursen. Energy consistent algorithms for frictional contact problems. *Internat. J. Numer. Methods Engrg.*, 42 :799–827, 1998.
- [9] M. J. P. Cullen and R. J. Purser. Properties of the lagrangian semigeostrophic equations. *Journal of the Atmospheric Sciences*, 46(17) :2684–2697, 1989.
- [10] C.A. Felippa and B. Haugen. A unified formulation of small-strain corotational finite elements : I. theory. *Computer Methods in Applied Mechanics and Engineering*, 194(21-24) :2285 – 2335, 2005. Computational Methods for Shells.
- [11] J. P. Fink and W. C. Rheinboldt. On the error behavior of the reduced basis technique for nonlinear finite element approximations. *ZAMM - Zeitschrift für Angewandte Mathematik und Mechanik*, 63(1) :21–28, 1983.
- [12] M. J. Gander and S. Vandewalle. Analysis of the parareal time-parallel time-integration method. *SIAM Journal on Scientific Computing*, 29(2) :556–578, 2007.
- [13] C. Grandmont, Y. Maday, and P. Métier. Modeling and analysis of an elastic problem with large displacements and small strains. *Journal of Elasticity*, 87(1) :29–72, 2007.
- [14] E. Haber, T. Rehman, and 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.
- [15] M. Hintermüller, K. Ito, and K. Kunisch. The primal-dual active set strategy as a semismooth Newton method. *SIAM J. Optim.*, 13(3) :865–888, 2003.
- [16] T. Laursen and V. Chawla. Design of energy conserving algorithms for frictionless dynamic contact problems. *Internat. J. Numer. Methods Engrg.*, 40 :836–886, 1997.

- [17] C. Le Bris, M. Mirrahimi, H. Rabitz, and G. Turinici. Hamiltonian identification for quantum systems : well-posedness and numerical approaches. *ESAIM : COCV*, 13(2) :378–395, 2007.
- [18] J.-L. Lions, Y. Maday, and G. Turinici. A "parareal" in time discretization of pde's. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, 332(7) :661 – 668, 2001.
- [19] G. Loeper and F. Rapetti. Numerical solution of the monge-ampère equation by a newton's algorithm. *Comptes Rendus Mathématique*, 340(4) :319–324, 2005.
- [20] Y. Maday, A. T. Patera, and G. Turinici. Global a priori convergence theory for reduced-basis approximations of single-parameter symmetric coercive elliptic partial differential equations. *Comptes Rendus Mathématique*, 335(3) :289 – 294, 2002.
- [21] Y. Maday and G. Turinici. New formulations of monotonically convergent quantum control algorithms. *J. Chem. Phys.*, 118 (18) : 81918196, 2003.
- [22] T. P. Mathew, M. Sarkis, and C. E. Schaerer. Analysis of block parareal preconditioners for parabolic optimal control problems. *SIAM Journal on Scientific Computing*, 32(3) :1180–1200, 2010.
- [23] T. A. Porsching. Estimation of the error in the reduced basis method solution of nonlinear equations. *Math. Comp.*, 45(172) :487–496, 1985.
- [24] G. Rozza. *Shape design by optimal flow control and reduced basis techniques : Applications to bypass configurations in haemodynamics*. PhD thesis, École Polytechnique Fédérale de Lausanne, November 2005.
- [25] L. Simon. Asymptotics for a class of non-linear evolution equations, with applications to geometric problems. *Ann. of Math.*, 118 :525–571, 1983.
- [26] D. Tannor, V. Kazakov, and V. Orlov. Control of photochemical branching : novel procedures for finding optimal pulses and global upper bounds. In : *Broeckhove, J., Lathouwers, L. (Eds.), Time Dependent Quantum Molecular Dynamics, Plenum, New York.*, pages 347–360, 1992.
- [27] B. Fraeijs De Veubeke. The dynamics of flexible bodies. *International Journal of Engineering Science*, 14(10) :895 – 913, 1976.
- [28] C. Villani. *Topics in optimal transportation*. American Mathematical Society, 2003.
- [29] F.L. Yip, D.A. Mazziotti, and H. Rabitz. A local-time algorithm for achieving quantum control. *J. Phys. Chem. A.*, 107 :7264–7269, 2003.
- [30] F.L. Yip, D.A. Mazziotti, and H. Rabitz. A propagation toolkit to design quantum control. *J. Chem. Phys.*, 118(18) :8168–8172, 2003.
- [31] W. Zhu and H. Rabitz. A rapid monotonically convergent iteration algorithm for quantum optimal control over the expectation value of a positive definite operator. *The Journal of Chemical Physics*, 109(2) :385–391, 1998.