



HAL
open science

Algorithmes hybrides pour le contrôle optimal des systèmes non linéaires

Aude Rondepierre

► **To cite this version:**

Aude Rondepierre. Algorithmes hybrides pour le contrôle optimal des systèmes non linéaires. Mathématiques [math]. Institut National Polytechnique de Grenoble - INPG, 2006. Français. NNT : . tel-00112203

HAL Id: tel-00112203

<https://theses.hal.science/tel-00112203>

Submitted on 7 Nov 2006

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

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

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

THÈSE

pour obtenir le grade de

DOCTEUR DE L'INPG

Spécialité : “Mathématiques Appliquées”

préparée au laboratoire de Modélisation et Calcul (LMC - IMAG)

dans le cadre de l'École Doctorale

“Mathématiques, Sciences et Technologies de l'Information, Informatique”

présentée et soutenue publiquement

par

Aude RONDEPIERRE

le 18 Juillet 2006

**Algorithmes hybrides pour le contrôle optimal
des systèmes non linéaires**

Directeurs de thèse :

Jean DELLA DORA, Jean-Guillaume DUMAS

JURY

M. Bernard BROGLIATO	Président du jury
M. Eugène ASARIN	Rapporteur
M. Dominikus NOLL	Rapporteur
M. Mazen ALAMIR	Examineur
M. Jean DELLA DORA	Directeur de thèse
M. Jean-Guillaume DUMAS	Co-directeur de thèse

Remerciements

Je tiens tout d'abord à remercier mes deux directeurs de thèse : Jean Della Dora pour m'avoir fait découvrir les systèmes hybrides et m'avoir accueillie au sein de l'équipe MOSAIC et Jean-Guillaume Dumas pour son enthousiasme et son soutien qui m'ont grandement aidé à mener à bien mes travaux de thèse et m'ont donné le goût de la recherche.

Je remercie chaleureusement Eugène Asarin et Dominikus Noll pour m'avoir fait l'honneur de rapporter cette thèse. Leurs remarques constructives et les discussions que nous avons eu, ont contribué à me donner du recul et m'encouragent à avancer.

Ma reconnaissance va également à Bernard Brogliato qui m'a fait l'honneur de présider mon jury ainsi qu'à Mazen Alamir pour avoir accepté le rôle d'examineur et pour ses remarques pertinentes qui m'ont aidé dans l'élaboration de la version finale de ce manuscrit.

J'adresse également un très grand merci à tous ceux que j'ai côtoyé ces dernières années au LMC pour leur gentillesse et leur efficacité. Une pensée particulière pour Claudine, Cathy et Juana. Mes remerciements vont également à tous ceux qui m'ont entouré, permanents ou thésards du LMC et d'ailleurs, pour les bons moments partagés ensemble. En particulier, je tiens à saluer Françoise, Valérie, Sylvain, Roland, Arnaud, Dominique, Claudine et Rodney.

Merci aux amis qui m'ont entouré et changé les idées : Karine, Nicolas, Jadou, Valou, Manu, Timo, Nicolas, Hélène, Amaël, Yannick et Béné, Hélène et Seb, un grand merci ! Merci à ceux qui sont loin mais toujours présents : Pierrick, Sandra, Julien, Antoine, Flo et Flo.

Merci Grumph, Carine, Irène, Yann, Olivier, Pramote, Cyrille, Alex, Julie, Aude, Laurent, Fred, Etienne et Antoine pour votre amitié et les soirées jeux endiablées. Merci Fred pour m'avoir fait découvrir une passion "poulpesque" et pour tout le reste ! Merci Laurent pour les longs moments passés à refaire le monde.

Merci Basile pour tout ce que nous partageons, les coups de pied et le reste ! Merci Claire pour ta présence et ton grand coeur. Votre amitié est précieuse.

Merci ma Floflo d'être une petite soeur rigolote, adorable et si proche. Qu'est-ce que je ferais sans toi ? Une mention spéciale à celle qui a eu le courage de lire ma thèse et de m'aider à la corriger !

Un immense merci à ma famille : merci Papa et Maman pour votre soutien inconditionnel et votre aide. Je ne vous remercierais jamais assez pour tout ça ! Une ovation à Maman qui a cuisiné et préparé mon pot de thèse de main de maître ! Merci Bertrand et Axelle d'être aussi adorables et de me changer les idées. Merci ma Mamie avec qui je passe de si bons moments. Une pensée spéciale pour mes beaux-parents pour leur gentillesse et leur soutien.

Enfin et surtout, un grand merci Vincent de m'accompagner et d'être là...

Table des matières

Table des figures	9
1 Introduction	13
1.1 Problématique	13
1.2 Méthodes numériques en commande optimale	16
1.3 Le calcul hybride	18
1.3.1 Définition des systèmes hybrides	19
1.3.2 Le calcul hybride	20
I Modélisation des systèmes de contrôle non linéaires par les systèmes hybrides	23
2 Approximation affine par morceaux des systèmes non linéaires	27
2.1 Calcul de l'approximation affine par morceaux	28
2.1.1 Interpolation du champ de vecteurs non linéaire	28
2.1.2 Propriétés de l'approximation	30
2.1.3 Une autre approche : les fonctions implicites	32
2.2 Transformation canonique des systèmes affines locaux	33
2.2.1 Décomposition canonique des systèmes affines	33
2.2.2 Algorithmes de décomposition canonique par blocs	35
2.2.3 Transformation locale du système	39
2.3 Convergence du schéma d'interpolation	41
2.3.1 Choix des hypothèses de régularité	42
2.3.2 Calcul de l'erreur d'interpolation	44
2.3.3 Convergence de l'approximation	47
3 Construction d'un modèle hybride de contrôle	51
3.1 Maillage implicite de l'espace état-contrôle	52
3.1.1 Caractérisation du maillage implicite	52
3.1.2 Triangulation de $\mathbb{R}^n \times \mathbb{U}_m$ et taille du maillage	54
3.2 Définition du modèle hybride	60
3.2.1 Choix des états discrets et définition de l'automate hybride	60
3.2.2 Construction à la volée de l'état courant	63
3.2.3 Calcul des contraintes locales sur le contrôle	65

II	Approximation du domaine contrôlable	71
4	Différents niveaux de contrôlabilité	75
4.1	Contrôlabilité et domaine contrôlable	76
4.1.1	Contrôlabilité des systèmes non linéaires ou atteignabilité?	76
4.1.2	Cas des systèmes linéaires : étude de la partie non contrôlable	79
4.1.3	Contrôlabilité de l'automate hybride	81
4.2	Approche hybride du domaine contrôlable non linéaire	83
4.2.1	Approximation d'ensembles : la distance d'Hausdorff	83
4.2.2	Étude de l'erreur globale d'approximation	84
4.2.3	Deux approches de la contrôlabilité d'un point initial donné	86
4.3	Réduction de l'exploration : domaine contrôlable sur un ensemble de cellules .	88
4.3.1	Domaine contrôlable à l'échelle d'une cellule	89
4.3.2	Propagation le long d'un chemin de cellules	90
4.3.3	Propagation du domaine contrôlable en p itérations	92
5	Approximation convexe du domaine contrôlable hybride	93
5.1	Adaptation des outils de simulation des systèmes hybrides au contrôle	94
5.1.1	Calcul des trajectoires continues pour un contrôle $u = FX + g$	95
5.1.2	Conditions d'entrée dans la cellule Δ_q	97
5.1.3	Simulation de l'évolution discrète	99
5.2	Propriétés topologiques du domaine contrôlable hybride liées à la convexité .	102
5.2.1	Cas des systèmes hybrides polyédraux affines	104
5.2.2	Cas du modèle hybride \mathcal{H} construit au chapitre 3	106
5.2.3	Discussion sur la convexité du domaine contrôlable hybride dans une cellule D_q	107
5.3	Algorithme de calcul d'une approximation convexe du domaine contrôlable .	109
5.3.1	Approximation convexe contrôlable dans un mode q	109
5.3.2	Cas des automates hybrides polyédraux, affines par morceaux	112
5.3.3	Approximation convexe sur un chemin	118
5.4	Validité de l'approximation convexe	123
5.4.1	Étude de l'erreur d'approximation	123
5.4.2	Contrôlabilité d'un point initial donné	124
III	Recherche de solutions optimales	127
6	Approche hybride pour le contrôle optimal des systèmes non linéaires	131
6.1	Conditions d'existence de trajectoires optimales	132
6.2	Première approche : principe du maximum de Pontriaguine	134
6.2.1	Formulation hamiltonienne des problèmes de contrôle optimal	134
6.2.2	Énoncé d'un principe du maximum hybride	137
6.2.3	Étude de l'approximation hybride	138
6.3	Seconde approche : solutions de viscosité des équations d'Hamilton-Jacobi-Bellman	140
6.3.1	Contrôle optimal et solutions de viscosité	140
6.3.2	Approche hybride pour la résolution de l'équation (HJB)	144

6.4	Cas particulier des problèmes de contrôle optimal affines	146
6.4.1	Arcs extrémaux réguliers et règles de commutation	148
6.4.2	Arcs extrémaux singuliers	150
6.4.3	Arcs extrémaux mixtes et conditions de jonction	155
7	Stratégie de résolution du problème de contrôle optimal hybride	161
7.1	Résolution locale du problème de contrôle hybride	162
7.1.1	Minimisation de l'hamiltonien dans une cellule D_q	163
7.1.2	Conditions nécessaires locales d'optimalité des contrôles $u = FX + g$.	165
7.2	Reconstruction des solutions extrémales candidates à l'optimalité	167
7.2.1	Structure du contrôle optimal dans un mode discret	168
7.2.2	Algorithme de simulation et problème d'initialisation de l'état adjoint	171
7.3	Une autre approche : recherche de trajectoires optimales dans un chemin de cellules	175
8	Réalisation logicielle et application	177
8.1	Réalisation logicielle en Maple	177
8.1.1	Bibliothèque	177
8.1.2	Quelques détails techniques	181
8.2	Problème du transfert orbital d'un satellite en temps minimal	182
8.2.1	Modélisation du problème de contrôle	182
8.2.2	Étude du domaine contrôlable	184
8.2.3	Recherche de trajectoires optimales	185
	Conclusion et perspectives	189
	Annexes	192
A	Quelques rappels de géométrie	193
A.1	Polyèdres et polytopes	193
A.2	Maillage d'un domaine en polytopes	194
B	Preuves des algorithmes pour le calcul de la forme de Kalman	195
B.1	Calcul de la forme de Kalman par l'algorithme de Keller-Gehrig	195
B.1.1	Algorithme	195
B.1.2	Démonstration du théorème 2.2.2	195
B.2	Calcul de la forme de Kalman par LU-Krylov	198
B.2.1	Algorithme et principe de démonstration	198
B.2.2	Démonstration	199

Table des figures

1.1	Tableau comparatif des méthodes directes et indirectes emprunté à [81] . . .	18
2.1	Définition de l'approximation affine par morceaux sur un maillage Δ de l'espace état-contrôle.	30
2.2	Principe de base de calcul de la matrice compressée de Krylov	37
2.3	Les cellules Δ_1 et Δ_2 (à gauche) et leurs images après transformation canonique (à droite)	41
3.1	Exemples de maillages, pour $n=m=1$, vérifiant la propriété 1. La figure (a) vérifie également la propriété 2. Ce n'est pas le cas de (b).	53
3.2	Découpage du cube $b+[0, h]^3$ en simplexes : sur la figure ci-dessus, on représente les simplexes $D_{b,\varphi} = Conv(\sigma_{\varphi,0}, \sigma_{\varphi,1}, \sigma_{\varphi,2}, \sigma_{\varphi,3})$, pour $\varphi = id$ et $\varphi = \varphi_0 = (3, 2, 1)$	55
3.3	Exemple de deux simplexes qui ne se recollent pas à l'intersection de deux cubes adjacents	56
3.4	Exemple de trajectoire discontinue dans l'espace état-contrôle $\mathbb{R} \times [0, u_{max}]$.	61
3.5	Illustration de la définition d'un automate hybride	62
3.6	Définition de l'ensemble $U_{q'}(X_0)$ des contraintes locales sur le contrôle dans une cellule D_q	65
3.7	Intersection $P \cap \Delta_{q'}$ pour $(n, m) = (1, 2)$ à gauche, $(n, m) = (2, 1)$ à droite. .	67
3.8	Problème de contrôlabilité	73
4.1	(a) Domaine contrôlable et (b) Trajectoires admissibles reliant X_0 à 0 en temps fini, du système (4.3).	77
4.2	Représentation d'une exécution d'un système hybride affine par morceaux. . .	82
4.3	Maillage d'un domaine à différentes échelles : h (en traits pleins), $\frac{h}{2}$ (en tiret-pointillés) et $\frac{h}{4}$ (en pointillés)	87
4.4	Différents niveaux d'approximation de la contrôlabilité d'un point donné par les systèmes hybrides	88
4.5	Définition du domaine Ω_γ associé à un chemin fini de cellules adjacentes contenant la cible 0.	89
4.6	Propagation du domaine contrôlable le long d'un chemin de cellules	90
4.7	Exemple de trajectoire exclue du sous-domaine contrôlable $\mathcal{C}_{\mathcal{H}}(D_{q_1} \cup D_{q_2})$. .	91
4.8	Propagation du domaine contrôlable à partir de la cellule D_{q_0} sur trois itérations	92
5.1	Sous-espace affine ($u = FX + g$) support de la trajectoire état-contrôle dans $\mathbb{R}^n \times \mathbb{U}_m$	96

5.2	Condition d'entrée dans une cellule $\Delta_{q'}$ de l'espace état-contrôle	98
5.3	Position du système \mathcal{H} dans l'espace d'état et dans l'espace état-contrôle . . .	101
5.4	Choix de la cellule état-contrôle dans un mode q donné en fonction de la position du sous-espace affine ($u = FX + g$)	101
5.5	Les points situés dans le domaine hachuré sont contrôlables jusqu'à la cible τ	104
5.6	Idée pour prolonger le domaine contrôlable obtenu en temps fixé.	107
5.7	Principe de calcul d'une approximation convexe contrôlable dans un état q contenant la cible : $\tau_q = \{0\}$. Le long de la trajectoire $X[0, u_1]$ (resp. $X[0, u_2]$), seuls les points compris entre 0 et X_1 (resp. 0 et X_2), sont contrôlables; par contre, aucun point de $X[0, u_3]$ n'est contrôlable dans D_q	110
5.8	Choix des contrôles $u_i(t)$ en fonction de la position $X(t)$ pour le calcul d'une approximation convexe du domaine contrôlable	111
5.9	Principe de l'approximation convexe du domaine contrôlable dans un état q .	112
5.10	(a) Domaine contrôlable exact (hachuré) (b) Approximation convexe donnée par l'algorithme 9 (en pointillés)	114
5.11	Subdivision par dichotomie de l'arête $[s_i, s_j]$ du domaine de contrôle \mathbb{U}_m . . .	115
5.12	Quel que soit $u \in [u_i, u_j]$, le point de sortie de la trajectoire $X[0, u]$ de la cellule D_q est déjà dans l'approximation convexe contrôlable	117
5.13	Amélioration de l'approximation convexe du domaine contrôlable hybride dans une cellule	118
5.14	Approximation convexe contrôlable calculée par l'algorithme 8 pour différentes valeurs du pas de discrétisation : $h \in \{2, 1, 0.5, 0.25, 0.1, 0.05, 0.01\}$	119
5.15	Construction d'une approximation convexe du domaine contrôlable sur une suite donnée ($q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8$) de modes discrets de l'automate \mathcal{H} .	120
5.16	Approximations contrôlables successives sur trois cellules D_{q_1}, D_{q_2} et D_{q_3} pour un pas $h = 1$ de discrétisation pour le problème du ressort non linéaire. . . .	122
5.17	Approximations contrôlables aux échelles $h = 2$ et $h = 1$ pour le problème du ressort non linéaire.	122
5.18	Approximation convexe contrôlable en dimension 3 pour l'exemple 5.3.4. . . .	123
6.1	Partition de l'espace (X, λ) en régions où le contrôle optimal est constant . .	146
6.2	Arcs régulier (à gauche) et singulier (à droite) avec commutation	147
6.3	Graphe des commutations autorisées (exemple de Johnson et Gibson [53]) . .	150
6.4	Frontières singulières potentielles du problème de Johnson et Gibson	155
6.5	Frontière mixte dans le problème de Johnson et Gibson et trajectoire optimale du système à partir du point initial $(-7, 0)$	159
7.1	La trajectoire état-contrôle optimale se déplace selon les arêtes du maillage Δ de $\mathbb{R}^n \times \mathbb{U}_m$	169
7.2	Tracé de l'hamiltonien H en fonction de $\alpha \in [0, 2\pi]$ où $X_0 = [0, 6]$ et $h = 1$. Les valeurs de α qui annulent H sont : $\alpha = \frac{3\pi}{2} \approx 2,094395103$ ou $\alpha \approx 1,901093561$ sachant que la valeur optimale de α pour le problème non linéaire est : $\alpha \approx 2,136718750$ (cf [81]).	175
8.1	Organisation des bibliothèques Maple pour la résolution hybride des problèmes de contrôle non linéaires (les flèches indiquent les composantes sur lesquelles s'appuient ces bibliothèques.	178

8.2	Constantes physiques pour le problème de transfert orbital	183
8.3	Conditions initiales et finales pour le problème de transfert orbital	184
8.4	Ensembles atteignables vis-à-vis (a) de l'excentricité $e = (e_x, e_y)$ (b) des paramètres (P, e_y, L) , à l'échelle la plus grossière ($h = 6$).	185
8.5	Approximation du domaine atteignable du satellite à l'échelle $h = 6$ dans un repère cartésien (r_1, r_2)	186
8.6	Extrémale de référence du problème de transfert d'un satellite	186
A.1	Exemples de d -simplexes de \mathbb{R}^3 pour $d = 0, 1, 2, 3$	194

Chapitre 1

Introduction

1.1 Problématique

Historiquement, la théorie du contrôle s'inscrit dans la continuité du calcul des variations et analyse les propriétés de tout système dynamique commandé i.e. sur lequel nous pouvons exercer un contrôle. Le contrôle optimal introduit une notion de rendement qu'il cherche à optimiser en faisant passer le système contrôlé d'un état initial vers un état final donnés.

Avec le développement des systèmes automatisés, les problèmes issus de l'industrie sont de plus en plus riches en problèmes d'optimisation qui peuvent être décrits sous la forme de problèmes de contrôle. Les applications de la théorie du contrôle optimal sont de ce fait extrêmement nombreuses dans des domaines très différents tels que l'aérospatiale (problèmes de transfert orbital, de planification de vols, du contrôle du trafic aérien [67]), l'automatique et la robotique (problèmes de coordination de mouvements de robots [38] par exemple) ou la biologie (contrôle de réseaux de régulation génique) pour n'en citer que quelques uns.

Cette théorie a connu un véritable essor depuis les années cinquante avec la découverte d'outils puissants tels que le principe de programmation dynamique de R. Bellman [9] ou le principe du maximum de Pontriaguine [70, 42]. Ces deux résultats jouent un rôle central en théorie du contrôle et ont donné lieu à deux types d'approches des problèmes de contrôle optimal : l'une est basée sur l'utilisation du principe du maximum de Pontriaguine, l'autre sur les propriétés de la fonction valeur du problème.

Cependant, la plupart de méthodes de résolution actuelles sont numériques. Deux familles de méthodes numériques se distinguent actuellement : les méthodes de tir indirectes et les méthodes directes [46, 81].

Les méthodes de tir indirectes sont basées sur l'utilisation du principe du maximum de Pontriaguine qui fournit une formulation hamiltonienne du problème de contrôle optimal considéré ainsi que des conditions nécessaires d'optimalité. Le principe de ces méthodes est d'exprimer le contrôle sous la forme d'une fonction régulière de l'état et de l'état adjoint et de se ramener ainsi à la résolution d'un problème aux valeurs limites que l'on peut résoudre par exemple par une méthode de Newton [11, 46]. Ces méthodes efficaces en toute dimension peuvent être difficiles à mettre en oeuvre (notamment pour la prise en compte de contraintes sur l'état) et demandent une connaissance a priori de la trajectoire optimale.

Les méthodes directes s'appuient sur la discrétisation en temps du problème de contrôle formulé en termes d'équation d'Hamilton-Jacobi-Bellman. Il en résulte un problème de programmation non linéaire en dimension finie, en général de grande taille, que nous pouvons alors résoudre par des outils d'optimisation non spécifiques à la théorie du contrôle. Ces méthodes efficaces en petite dimension sont donc généralement simples à mettre en oeuvre mais sont trop gourmandes en mémoire pour que l'on puisse espérer les appliquer en dimension supérieure à 6 ou 7.

L'enjeu actuel est donc d'obtenir un compromis entre méthodes directes et méthodes indirectes afin de pouvoir traiter des problèmes de taille conséquente avec une bonne précision et sans avoir à effectuer de pré-simulations pour localiser les solutions optimales recherchées.

Dans ce manuscrit, nous proposons une nouvelle approche des problèmes de contrôle par des méthodes de calcul hybride. Nous nous intéressons plus spécifiquement à deux questions essentielles : la contrôlabilité (i.e. l'existence de trajectoires pour des conditions initiales et finales fixées) et la recherche de solutions optimales.

L'idée défendue dans cette thèse est que la modélisation par les systèmes hybrides permet la résolution approchée des problèmes de contrôle non linéaires sans connaissance a priori du comportement du système étudié. L'objectif est donc de développer par le calcul hybride combinant calcul formel et analyse numérique, des outils mathématiques et algorithmiques efficaces pour l'étude de dynamiques contrôlées non linéaires en toute dimension.

Ce manuscrit s'articule autour de trois grandes parties : la modélisation hybride des systèmes de contrôle non linéaires, la contrôlabilité puis la recherche de solutions optimales approchées grâce à l'approximation hybride.

Modélisation des systèmes de contrôle non linéaires par les systèmes hybrides

Cette première partie est consacrée à la modélisation des systèmes de contrôle non linéaires par une nouvelle classe de systèmes hybrides affines par morceaux.

Le chapitre 2 propose une généralisation des méthodes de calcul hybride aux systèmes contrôlés. Un soin particulier est apporté à l'étude de l'erreur et de la convergence de l'approximation affine par morceaux en fonction des hypothèses de régularité du champ non linéaire modélisé.

Le chapitre 3 introduit une nouvelle classe de systèmes hybrides affines par morceaux particulièrement adaptés à la modélisation des systèmes de contrôle non linéaires. Nous y développons un ensemble de méthodes et d'algorithmes permettant le calcul à la volée d'un maillage de l'espace état-contrôle ainsi que de l'état discret du système hybride.

Approximation du domaine contrôlable

Dans cette deuxième partie, nous nous intéressons au problème de la contrôlabilité à l'origine des systèmes de contrôle.

Le chapitre 4 présente une étude de la contrôlabilité des systèmes non linéaires et hybrides, basée sur la caractérisation du domaine contrôlable de ces systèmes. Dans un premier temps, nous nous intéressons à la quantification de l'approximation du domaine contrôlable d'un système non linéaire par le domaine contrôlable du modèle hybride construit au chapitre 3. Dans un second temps, nous proposons une approche constructive de la contrôlabilité d'un point donné permettant de réduire l'exploration des états discrets de l'automate hybride.

Dans le chapitre 5, nous développons un algorithme d'approximation convexe du domaine contrôlable sur des chemins discrets pour la classe des systèmes hybrides affines par morceaux dans des chemins discrets de modes discrets. Cet algorithme repose sur des algorithmes de simulation des systèmes hybrides, généralisés dans la première partie de ce chapitre aux dynamiques hybrides contrôlées.

Recherche de solutions optimales

Cette dernière partie est dédiée à la recherche de solutions optimales des problèmes de contrôle non linéaires et hybrides.

Dans le chapitre 6, nous justifions la pertinence de la modélisation hybride des problèmes de contrôle non linéaires à travers deux approches : la première repose sur l'utilisation du principe du maximum de Pontriaguine, nous amenant à définir un principe du maximum hybride adapté aux modèles hybrides affines par morceaux. La seconde s'appuie sur la caractérisation des problèmes de contrôle optimal en termes de solutions de viscosité d'une équation d'Hamilton-Jacobi-Bellman. Nous montrons alors que l'approche hybride permet une résolution approchée de ce type d'équation.

Le chapitre 7 propose une stratégie de résolution des problèmes de contrôle optimal hybrides basée sur l'utilisation du principe du maximum hybride défini au chapitre précédent. Nous montrons en particulier comment exploiter les conditions nécessaires d'optimalité pour en déduire la forme générale du contrôle optimal.

Un dernier chapitre est dédié à la description des librairies Maple que nous avons développées et qui mettent en oeuvre les techniques et les algorithmes présentés tout au long de ce manuscrit. Nous concluons ce chapitre par une application de ces outils au problème du transfert orbital de satellite et par une confrontation des résultats obtenus aux méthodes numériques existantes.

Enfin, un bilan de ces travaux et des contributions que nous apportons est présenté dans le chapitre de conclusion.

Ce premier chapitre présente les concepts généraux qui serviront de point de départ à notre étude. La section 1.2 effectue une synthèse des différentes méthodes numériques en théorie du contrôle optimal dont l'étude comparative complète peut être trouvée dans [81]. Dans la section 1.3, nous proposons une introduction à la théorie des systèmes hybrides et au calcul hybride.

1.2 Méthodes numériques en commande optimale

Cette partie a pour vocation de donner au lecteur une idée des principes des méthodes numériques de résolution des problèmes de contrôle optimal. Nous distinguons deux familles de méthodes numériques : les méthodes de tir indirectes et les méthodes directes. Après avoir très succinctement présenté le principe de chacune de ces méthodes, nous présentons un tableau comparatif des caractéristiques de ces méthodes emprunté à E. Trélat au chapitre 9 de son livre [81].

Nous considérons dans cette partie un système de contrôle général de la forme :

$$\dot{X}(t) = f(X(t), u(t)), \quad (1.1)$$

où les contrôles admissibles sont des fonctions mesurables bornées à valeurs dans un sous-ensemble \mathbb{U} de \mathbb{R}^m . Pour tout contrôle u admissible, nous introduisons alors la fonction coût :

$$J(X_0, u) = \int_0^{t_f} l(X(t), u(t)) dt.$$

Le problème de contrôle optimal consiste à amener le système (1.1) d'un état initial vers un état final donné en minimisant le coût considéré et en temps T fixé.

Méthodes indirectes

Les méthodes de tir indirectes sont basées sur l'utilisation du principe du maximum de Pontriaguine [70] qui nous donne des conditions nécessaires d'optimalité du premier ordre. Nous introduisons l'hamiltonien associé au problème de contrôle considéré :

$$H(X, u, \lambda) = l(X, u) + \lambda^T f(X, u).$$

Nous avons alors à résoudre le problème d'optimisation suivant, dicté par le principe de Pontriaguine :

$$\left\{ \begin{array}{l} \dot{X}(t) = \frac{\partial H}{\partial \lambda}(X(t), u(t), \lambda(t)), \quad \dot{\lambda}(t)^T = -\frac{\partial H}{\partial X}(X(t), u(t), \lambda(t)) \\ X(0) = X_0, \quad \lambda(t_f) = 0 \\ u(t) \in \underset{v \in \mathbb{U}}{\operatorname{argmin}} H(X(t), v, \lambda(t)) \end{array} \right.$$

Le point clé des méthodes de tir est de caractériser le contrôle : $u(t) \in \underset{v \in \mathbb{U}}{\operatorname{argmin}} H(X(t), v, \lambda(t))$ comme une fonction régulière de l'état X et de l'état adjoint λ :

$$u(t) = \psi(X(t), \lambda(t)).$$

Posons $Z(t) = (X(t), \lambda(t))$. Nous supposons que les conditions initiales et finales sur l'état et l'état adjoint s'écrivent sous la forme :

$$R(Z(0), Z(t_f)) = 0.$$

En injectant cette expression dans le système hamiltonien, nous obtenons un système en $Z = (X, \lambda)$:

$$\begin{cases} \dot{Z}(t) = F(Z(t)) \\ R(Z(0), Z(t_f)) = 0 \end{cases} \quad (1.2)$$

sachant que : $F(X, \lambda) = \left(f(X, \psi(X, \lambda)), -\frac{\partial H}{\partial X}(X, \psi(X, \lambda), \lambda) \right)$.

L'idée est alors d'introduire la solution $Z(Z_0, t)$ du système $\dot{Z}(t) = F(Z(t))$ associé à une condition initiale $Z(0) = Z_0$. Par conséquent, $Z(Z_0, t)$ est une solution du système (1.2) si et seulement si la condition initiale $R(Z(0), Z(t_f)) = 0$ est vérifiée, ce qui revient à trouver une condition initiale Z_0 vérifiant :

$$R(Z_0, Z(Z_0, t_f)) = 0,$$

Ce type d'équation peut être résolue par une méthode de Newton.

L'un des inconvénients de cette méthode est le problème de l'initialisation de l'état adjoint qui a peu de signification physique. Le domaine de convergence de la méthode de tir étant petit, la marge de manoeuvre pour le choix de l'état adjoint est très restreinte.

Remarque 1.2.1. *Il existe également des méthodes de tir multiple qui permettent d'améliorer la stabilité de la méthode de tir présentée ci-dessus. L'idée est de découper l'intervalle de temps $[0, t_f]$ en N intervalles $[t_i, t_{i+1}]$ sur lesquels on applique une méthode de tir simple en prenant soin de respecter les conditions de recollement des solutions aux instants t_i de jonction.*

Méthodes directes

Les méthodes directes consistent à transformer le problème de contrôle optimal en un problème d'optimisation non linéaire en dimension finie.

Une première classe de techniques consiste à effectuer une discrétisation totale du problème de contrôle optimal considéré se ramenant ainsi un problème de programmation non linéaire classique.

Une seconde classe de techniques s'appuient sur le principe de programmation dynamique et les propriétés de la fonction valeur :

$$V(X_0) = \inf_{u \in L^\infty([0, t_f], \mathbb{U})} J(X_0, u)$$

dont on démontre qu'elle est une solution de viscosité d'une équation d'Hamilton-Jacobi-Bellman :

$$\frac{\partial V}{\partial t} + \sup_{u \in \mathbb{U}_m} \left\{ -l(X, u) - \frac{\partial V}{\partial X} f(X, u) \right\} = 0.$$

Ces techniques utilisent une version discrète du principe de programmation dynamique sur une discrétisation en espace et/ou en temps de cette équation d'Hamilton-Jacobi-Bellman [7, 10, 18]. Le schéma aux différences finies est un exemple de schéma simple de discrétisation permettant de résoudre cette équation.

Malheureusement, malgré le développement de techniques efficaces de discrétisation (grâce à l'utilisation de maillages adaptatifs par exemple), la complexité de ces algorithmes croît exponentiellement avec la dimension de l'espace d'état et les rend inapplicables en grande dimension.

Bilan

Le tableau de la figure 1.2 présente une synthèse des caractéristiques de chaque méthode.

En résumé, pour tout problème où l'on souhaite une très grande précision numérique (comme en particulier dans les problèmes issus de l'aérospatiale), le choix d'une méthode de tir indirecte est incontournable. Pour pallier au problème de l'initialisation de l'état adjoint et du domaine de convergence de la méthode, des méthodes combinant des méthodes directes et des méthodes indirectes ont été élaborées. L'idée est d'obtenir grâce aux méthodes directes une première approximation de la trajectoire optimale, ce qui nous donne des informations sur l'état adjoint optimal et donc des conditions d'initialisation pour appliquer ensuite une méthode de tir indirecte.

Méthodes directes	Méthodes indirectes
Mise en oeuvre simple, sans connaissance a priori	Connaissance a priori de la structure de la trajectoire optimale
Peu sensibles au choix de la condition initiale	Très sensibles au choix de la condition initiale
Facilité de la prise en compte de contraintes sur l'état	Difficulté théorique de la prise en compte de contraintes sur l'état
Contrôles (globalement) optimaux en boucle fermée	Contrôles (localement) optimaux en boucle ouverte
Précision numérique basse ou moyenne	Très grande précision numérique
Efficaces en basse dimension	Efficaces en toute dimension
Gourmandise en mémoire	Calculs parallélisables
Problème de minima locaux	Petit domaine de convergence

FIG. 1.1 – Tableau comparatif des méthodes directes et indirectes emprunté à [81]

1.3 Le calcul hybride

Les systèmes hybrides constituent un cadre mathématique pour la modélisation et l'étude de tout phénomène constitué de processus continus interagissant avec ou supervisés par des processus discrets. Ils se prêtent également bien à la modélisation de systèmes physiques continus soumis par exemple à de brusques variations de certaines de leurs variables. Ces variations sont alors représentées par des événements discrets.

Les applications des systèmes hybrides sont extrêmement nombreuses et variées que ce soit en informatique, en robotique [38], pour l'étude de réseaux de régulation biologiques [80]

ou dans bien d'autres domaines encore. Avec le développement rapide des systèmes automatisés, les systèmes hybrides sont devenus une discipline à part entière de l'automatique. Une synthèse très complète de leur rôle pour la modélisation, l'étude et la commande des systèmes automatisés est présentée dans [85].

Les enjeux de la théorie des systèmes hybrides sont de façon générale d'apporter des solutions en termes de modélisation, de méthodes et de performance à de nombreux problèmes mal traités par les approches classiques.

Cette partie est une courte introduction à la théorie des systèmes hybrides. Après avoir défini le concept général de système hybride et identifié certaines classes remarquables de ces systèmes, nous présentons le calcul hybride qui est l'outil de modélisation des systèmes complexes par les systèmes hybrides utilisé dans ce manuscrit.

1.3.1 Définition des systèmes hybrides

Du fait de la richesse de leurs applications, les systèmes hybrides ont longtemps manqué d'un cadre théorique unifié. Entre autres, les travaux de M.S. Branicky [15] ont permis d'établir une définition très générale des automates hybrides que nous utilisons maintenant sous une forme proche de celle proposée par A. Girard [44] :

Définition 1.3.1. *Un système hybride contrôlé est un septuple*

$$\mathcal{H} = (\mathcal{Q}, \mathcal{E}, \mathcal{D}, \mathcal{U}, \mathcal{F}, \mathcal{G}, \mathcal{R})$$

où :

1. \mathcal{Q} est l'ensemble dénombrable des états discrets (ou modes).
2. $\mathcal{E} \subset \mathcal{Q} \times \mathcal{Q}$ est l'ensemble des transitions.
3. $\mathcal{D} = \{D_q / q \in \mathcal{Q}\}$ est la collection des domaines de l'espace d'état.
 $\forall q \in \mathcal{Q}, D_q \subset \mathbb{R}^n$ et $\overset{\circ}{D}_q \neq \emptyset$
4. $\mathcal{U} = \{U_q ; q \in \mathcal{Q}\}$ est la collection des domaines de contrôle.
 $\forall q \in \mathcal{Q}, U_q$ est un sous-ensemble de \mathbb{R}^m .
5. $\mathcal{F} = \{\mathcal{F}_q / q \in \mathcal{Q}\}$ la collection des champs de vecteurs.
 $\forall q \in \mathcal{Q}, f_q : D_q \times U_q \rightarrow \mathbb{R}^n$
6. $\mathcal{G} = \{G_e / e \in \mathcal{E}\}$ est la collection des gardes de l'automate.
 $\forall e = (q, q') \in \mathcal{E}, G_e \subset D_q$
7. $\mathcal{R} = \{R_e / e \in \mathcal{E}\}$ est la collection des fonctions Reset.
 $\forall e = (q, q') \in \mathcal{E}, R_e : G_e \rightarrow \mathcal{P}(D_{q'})$ où $\mathcal{P}(D_{q'})$ désigne l'ensemble des parties de $D_{q'}$. On suppose que, pour tout $X \in G_e, R_e(X) \neq \emptyset$.

A tout instant, l'état d'un système hybride est décrit par deux variables : l'une discrète notée $q(t)$ à valeurs dans l'ensemble dénombrable \mathcal{Q} des états discrets, la seconde continue notée $X(t)$ à valeurs dans l'espace des phases. Dans chaque état $q \in \mathcal{Q}$, la variable X appartient au domaine D_q et son évolution est soumise à la dynamique $\dot{X}(t) = f_q(X(t), u(t))$ du mode considéré sachant que la variable u prend ses valeurs dans l'ensemble U_q .

La dynamique d'un système hybride est donc décrite par le couplage d'une composante discrète $(\mathcal{Q}, \mathcal{E})$ et d'une composante continue $(\mathcal{D}, \mathcal{U}, \mathcal{F})$. Ce couplage se fait par l'intermédiaire des gardes du système. En effet, pour qu'une transition e du système hybride puisse se produire à un instant t , il faut que l'état continu $X(t)$ appartienne à l'ensemble G_e .

Du fait de cette définition, les systèmes hybrides sont généralement représentés au moyen de graphes ou d'automates dont les états correspondent aux modes discrets du système. Dans la suite de ce manuscrit, nous employons indifféremment les termes de système hybride ou d'*automate hybride*.

La classe des systèmes hybrides polyédraux

Les systèmes hybrides polyédraux sont des systèmes hybrides dont la collection de domaines $\mathcal{D} = \{D_q / q \in \mathcal{Q}\}$ forme un maillage en polyèdres de l'espace d'état \mathbb{R}^n et dont la collection de fonctions Reset vérifie :

$$\forall e \in \mathcal{E}, R_e(X) = \{X\}.$$

Ainsi, l'ensemble \mathcal{Q} des états discrets du système correspond à l'ensemble dénombrable des indices du maillage \mathcal{D} . Une transition $e = (q, q') \in \mathcal{E}$ entre deux états q et q' est possible si et seulement si les cellules associées D_q et $D_{q'}$ ont une frontière en commun. Cette frontière notée G_e est appelée garde du système associée à la transition e . De plus, à chaque transition, la variable continue X n'est pas ré-initialisée ce qui assure la continuité de la trajectoire du système.

La sous-classe des systèmes hybrides affines

La sous-classe la plus connue et la plus utilisée des systèmes hybrides polyédraux est celle des systèmes hybrides affines par morceaux :

Définition 1.3.2 (Système hybride polyédral affine). *Un système hybride polyédral affine est un système hybride polyédral dont la dynamique est affine dans chaque état discret.*

Ces systèmes sont en effet devenus un outil pertinent et performant dans la modélisation de nombreux phénomènes physiques (en théorie des circuits électroniques ou en biologie par exemple) pour deux raisons. Ils permettent tout d'abord de construire des modèles réalistes qui préservent les caractéristiques fondamentales des phénomènes physique. Ensuite, par leur relative simplicité, ils rendent possible l'analyse mathématique et la conception d'outils algorithmiques efficaces.

1.3.2 Le calcul hybride

Le calcul hybride introduit en 2001 dans [29] et développé par la suite dans [44] est une méthodologie pour l'approximation des systèmes dynamiques non linéaires par des modèles simplifiés pour lesquels nous disposons d'outils mathématiques et algorithmiques permettant leur analyse.

Considérons une équation différentielle autonome de la forme :

$$\dot{X}(t) = f(X(t)).$$

Les méthodes numériques classiques de type Euler ou Runge-Kutta proposent traditionnellement une discrétisation en temps de ce type d'équation et fournissent une discrétisation des trajectoires solutions.

L'idée du calcul hybride est radicalement différente : plutôt que de chercher une approximation de la dérivée $\dot{X}(t)$, nous nous intéressons au calcul d'une approximation du champ non linéaire f par le biais d'une discrétisation de l'espace d'état. Le principe est le suivant : on construit un maillage de l'espace d'état sur lequel on calcule localement des approximations affines du champ non linéaire que l'on souhaite modéliser. Le maillage associé à la dynamique approchée définit implicitement un système hybride que l'on étudie.

Le succès de cette méthode repose sur le choix des approximations dans chaque cellule du maillage de l'espace d'état. Elles doivent en effet être suffisamment élaborées afin d'obtenir un modèle réaliste qui préserve les caractéristiques fondamentales du système initial, mais aussi suffisamment simples pour permettre l'étude mathématique et algorithmique. Dans le contexte des systèmes non contrôlés autonomes, les travaux d'A. Girard ont montré que le choix d'approximants linéaires dans chaque cellule du maillage réalise ce compromis quelle que soit la dimension du système [44].

Les points forts du calcul hybride

L'un des premiers atouts du calcul hybride tient dans la souplesse et la simplicité de sa mise en oeuvre sans connaissance a priori et quelle que soit la dimension du système que l'on souhaite modéliser. En particulier, pour la simulation des systèmes hybrides obtenus par le biais du calcul hybride, l'astuce consiste à ne pas précalculer le maillage de l'espace d'état et donc à construire l'automate hybride à la volée.

Ensuite, un deuxième atout réside dans le choix des approximations affines dans chaque cellule du maillage de l'espace pour lesquels des résolutions formelles sont possibles. Nous voulons cependant insister sur le fait que le calcul hybride ne permet pas d'obtenir des modèles faciles à étudier mais des modèles pour lesquels nous disposons d'outils mathématiques qui nous permettent d'aborder leur étude. Là où les méthodes numériques demandent des études de stabilité et de conditionnement, l'utilisation du calcul formel permet de garantir et de démontrer certaines propriétés des résultats obtenus.

Nous verrons dans le chapitre 3 une application des méthodes de calcul hybride aux systèmes dynamiques avec des termes de contrôle, la difficulté étant le manque de régularité en général des fonctions de contrôle.

Première partie

Modélisation des systèmes de contrôle non linéaires par les systèmes hybrides

Cette première partie est consacrée à la modélisation des systèmes de contrôle non linéaires par les systèmes hybrides affines par morceaux.

On considère des systèmes dynamiques non linéaires de la forme :

$$\dot{X}(t) = f(X(t), u(t)),$$

où la variable u désigne le contrôle, supposé borné, exercé sur le système. Le but de cette partie est de proposer une modélisation qui nous permette d'approcher de manière satisfaisante le comportement (solutions, portrait de phase, trajectoires optimales) du système non linéaire considéré.

L'outil de modélisation présenté dans ce manuscrit est la classe des systèmes hybrides affines par morceaux introduite au chapitre 1. L'idée est de calculer une approximation affine par morceaux de la dynamique contrôlée non linéaire que l'on se souhaite modéliser. À partir de cette approximation construite sur un maillage implicite de l'espace état-contrôle, on définit un modèle hybride contrôlé qui nous permet d'approcher les solutions des problèmes de contrôle traités dans les parties II et III.

Cette partie présente des méthodes et algorithmes permettant de modéliser tout système de contrôle non linéaire par le biais des systèmes hybrides affines par morceaux. Le chapitre 2 est consacré au calcul, puis à l'étude d'une approximation affine par morceaux et continue d'un système dynamique contrôlé non linéaire donné. Justifiant la pertinence de la modélisation proposée, nous y détaillons une analyse de l'erreur puis de la convergence de cette approximation en fonction des hypothèses de régularité choisies sur le champ f . Le chapitre 3 utilise les outils de modélisation développés au chapitre 2 pour construire un modèle hybride affine par morceaux adapté aux systèmes de contrôle non linéaires.

Chapitre 2

Approximation affine par morceaux des systèmes non linéaires

Dans ce chapitre nous nous intéressons à l'approximation par des modèles continus, affines par morceaux, des systèmes non linéaires de la forme :

$$\dot{X}(t) = f(X(t), u(t)), \quad (2.1)$$

où f est une application continue de $\mathbb{R}^n \times \mathbb{R}^m$ dans \mathbb{R}^n . Par ailleurs, les contrôles admissibles u sont des fonctions mesurables bornées à valeurs dans un polytope \mathbb{U}_m de \mathbb{R}^m .

Historiquement, la classe des systèmes affines par morceaux a été réellement reconnue comme un outil performant pour la modélisation des systèmes non linéaires avec l'essor de la théorie des circuits et des réseaux. En effet, un grand nombre de non linéarités rencontrées dans les circuits électriques, ont des comportements affines par morceaux : par exemple, les diodes et les transistors qui sont des composants clés des circuits électriques, sont naturellement modélisés par des systèmes affines par morceaux. Motivé par le besoin de simulations efficaces et la perspective de traiter des circuits à grande échelle, un effort tout particulier est accordé à la représentation efficace des systèmes affines par morceaux [54, 76].

Une approche de la modélisation des systèmes non linéaires du type $Y = g(X)$ pour la théorie des circuits [20], ou $\dot{X}(t) = g(X(t))$ pour les systèmes différentiels ordinaires autonomes [43], est basée sur le découpage de l'espace d'état en simplexes. Une approximation affine par morceaux du champ non linéaire g est calculée par interpolation du champ non linéaire g aux sommets du maillage [20, 43]. L'avantage de cette approche réside dans la simplicité de la formulation théorique, ce qui nous permet de l'appliquer facilement à des systèmes non linéaires définis a priori en toute dimension.

Dans ce chapitre, nous proposons une généralisation de cette méthode aux systèmes de contrôle non linéaires tels que le système (2.1). Le principe est le suivant : on se donne un maillage en simplexes de l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$. Dans chaque cellule du maillage, on construit une approximation affine de la forme : $\dot{X}(t) = AX(t) + Bu(t) + c$ du champ non linéaire f , calculée par interpolation aux sommets de la cellule. Cette approximation est

ensuite décomposée sous la forme canonique de Kalman [56, 57] :

$$\dot{X}(t) = \begin{bmatrix} A_1 & A_2 \\ 0 & A_3 \end{bmatrix} X(t) + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} c_1 \\ c_2 \end{bmatrix},$$

faisant ainsi apparaître une partie indépendante du contrôle. Le maillage, associé à la dynamique par morceaux, définit un système hybride que nous explicitons au chapitre 3.

Dans un premier temps, nous construisons une approximation affine par morceaux du système (2.1) par interpolation du champ de vecteurs non linéaire f aux sommets du maillage. Nous détaillons ensuite un nouvel algorithme permettant de mettre les systèmes affines locaux sous la forme canonique de Kalman. Nous concluons ce chapitre par une étude détaillée de l'erreur d'interpolation, ainsi que de la convergence de l'approximation.

Certains des résultats exposés dans les paragraphes 2.1 et 2.3, sont inspirés de ceux présentés dans [44, chapitre 11] dans le cadre des équations différentielles ordinaires autonomes. Cependant, dans notre démarche, les résultats de [44] concernant la convergence du schéma d'approximation ne sont généralisables aux systèmes de contrôle non linéaires que sous des hypothèses de régularité fortes sur le champ f , hypothèses rarement vérifiées en théorie du contrôle. Nous proposons en conséquence une étude approfondie et adaptée de la convergence du schéma en fonction de la régularité du champ non linéaire f et de son approximation.

2.1 Calcul de l'approximation affine par morceaux

Soit $\Delta = (\Delta_i)_{i \in I}$ un maillage simplicial donné de $\mathbb{R}^n \times \mathbb{U}_m$, où I est un ensemble dénombrable d'indices. Nous voulons approcher le système non linéaire (2.1) par un système :

$$\dot{X}(t) = f_h(X(t), u(t)), \tag{2.2}$$

où f_h est une approximation continue, affine par morceaux du champ de vecteurs f dans l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$.

Il existe différentes méthodes de linéarisation par morceaux. Par exemple, dans le cas de systèmes dynamiques autonomes : $\dot{X}(t) = g(X(t))$, les auteurs de [29, 43, 4] utilisent une interpolation multi-dimensionnelle sur un maillage implicite de l'espace d'état, définissant ainsi une approximation affine par morceaux dans chaque cellule (simplexe) du maillage. On peut également linéariser séparément chaque équation du système non linéaire par des méthodes de fonctions implicites, comme nous l'avons déjà fait sur le modèle biologique du neurone [34].

La méthode présentée ci-après est celle de l'interpolation du champ f aux noeuds du maillage Δ de l'espace état-contrôle. Dans un premier temps, nous appliquons les techniques d'interpolation présentées dans [20, 44] dans le cas des systèmes ordinaires, aux systèmes de contrôle non linéaires. Nous étudions ensuite les propriétés de l'approximation f_h ainsi construite.

2.1.1 Interpolation du champ de vecteurs non linéaire

Dans ce paragraphe, nous voulons calculer une approximation affine par morceaux f_h du champ de vecteurs non linéaire f . Dans [20, 44], les auteurs présentent des techniques d'inter-

polution aux sommets d'un maillage donné de l'espace d'état \mathbb{R}^n , pour des dynamiques non contrôlées (i.e. indépendantes de tout contrôle). Ces techniques s'étendent aux dynamiques contrôlées telles que (2.1), permettant ainsi le calcul d'une approximation affine par morceaux par rapport à la position X et au contrôle u .

Il suffit en effet de considérer le champ f comme une fonction à $(n+m)$ variables : toute application affine définie sur \mathbb{R}^{n+m} étant caractérisée de façon unique par sa valeur en $n+m+1$ points affinement indépendants, on peut alors calculer dans chaque simplexe du maillage Δ une approximation affine du champ f par interpolation aux sommets.

Soit Δ_i une cellule quelconque du maillage Δ définie comme l'enveloppe convexe de ses sommets : $\Delta_i = \text{Conv}(\sigma_{i,1}, \dots, \sigma_{i,n+m+1})$. On introduit f_i l'approximation affine de f dans la cellule Δ_i :

$$f_i(X, u) = A_i X + B_i u + c_i = [A_i \mid B_i] \begin{bmatrix} X \\ u \end{bmatrix} + c_i. \quad (2.3)$$

L'approximation f_i est calculée par interpolation du champ de vecteurs f aux sommets de la cellule Δ_i et vérifie donc : $\forall j \in \{1, \dots, n+m+1\}$, $f_i(\sigma_{i,j}) = f(\sigma_{i,j})$, soit :

$$\forall j \in \{1, \dots, n+m+1\}, [A_i \mid B_i] \sigma_{i,j} + c_i = f(\sigma_{i,j}).$$

De ces contraintes d'interpolation, on déduit les relations suivantes :

$$\forall j \in \{1, \dots, n+m+1\}, f(\sigma_{i,j}) - f(\sigma_{i,1}) = [A_i \mid B_i] (\sigma_{i,j} - \sigma_{i,1}) \quad (2.4)$$

$$c_i = f(\sigma_{i,j}) - [A_i \mid B_i] \sigma_{i,j} \quad (2.5)$$

qui vont nous permettre d'explicitier la fonction affine f_i .

On note M_i la matrice de taille $(n+m) \times (n+m)$ formée des vecteurs colonnes $\sigma_{i,j} - \sigma_{i,1}$, $j = 2, \dots, n+m+1$ et F_i la matrice de taille $(n+m) \times n$ formée des vecteurs colonnes $f(\sigma_{i,j}) - f(\sigma_{i,1})$, $j = 2, \dots, n+m+1$.

Les contraintes d'interpolation (2.4) peuvent alors s'exprimer sous forme matricielle : $F_i = [A_i \mid B_i] M_i$. De plus, par indépendance affine des sommets du simplexe Δ_i , la matrice carrée M_i est inversible ; d'où :

$$\begin{aligned} [A_i \mid B_i] &= F_i M_i^{-1} \\ c_i &= f(\sigma_{i,1}) - [A_i \mid B_i] \sigma_{i,1} \end{aligned} \quad (2.6)$$

Remarque 2.1.1. La constante c_i peut être calculée à partir de n'importe quel sommet $\sigma_{i,j}$ du simplexe Δ_i . En effet, d'après la contrainte d'interpolation (2.4), on a :

$$\forall j \in \{1, \dots, n+m+1\}, f(\sigma_{i,j}) + [A_i \mid B_i] \sigma_{i,j} = f(\sigma_{i,1}) + [A_i \mid B_i] \sigma_{i,1}.$$

L'approximation affine par morceaux f_h de f est donc définie par :

$$f_h(X, u) = f_i(X, u) = A_i X + B_i u + c_i, \text{ si } (X, u) \in \Delta_i$$

et est illustrée sur la figure 2.1.

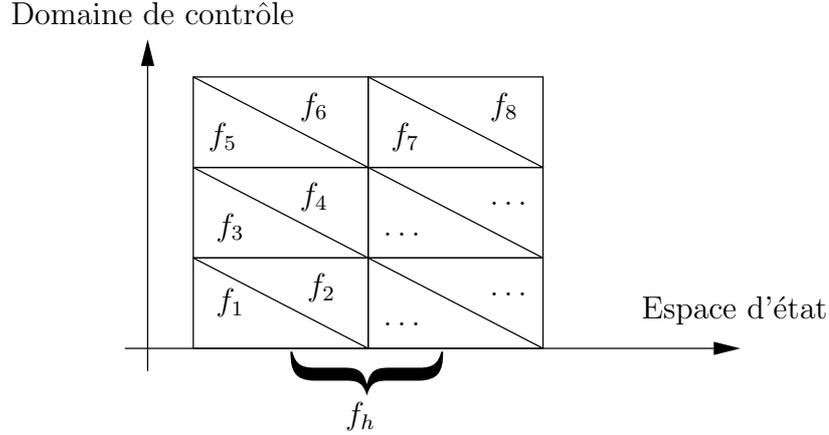


FIG. 2.1 – Définition de l’approximation affine par morceaux sur un maillage Δ de l’espace état-contrôle.

2.1.2 Propriétés de l’approximation

Étudions maintenant les propriétés de l’approximation affine par morceaux f_h du champ non linéaire initial f construite au paragraphe 2.1.1.

On rappelle que h désigne la taille du maillage $\Delta = (\Delta_i)_{i \in I}$, définie par :

$$h = \sup_{i \in I} h_i \quad \text{avec : } h_i = \sup_{x, y \in \Delta_i} \|x - y\|,$$

où $\|\cdot\|$ dénote la norme ∞ sur \mathbb{R}^{n+m} .

Proposition 2.1.1. f_h est continue sur $\mathbb{R}^n \times \mathbb{U}_m$.

Idée de démonstration. La continuité de f_h est une conséquence directe du fait que les simplexes Δ_i du maillage Δ ne se chevauchent pas. En effet, à l’intérieur de toute cellule du maillage Δ , l’approximation f_h est affine donc continue. Il suffit donc de montrer que f_h se recolle continûment à l’intersection de deux cellules de Δ . Ce dernier point a été démontré dans [44, proposition 11.1.3] dans le cas de l’approximation des systèmes autonomes et s’applique ici en dimension $n + m$ au champ f et au maillage Δ de $\mathbb{R}^n \times \mathbb{U}_m$.

On se donne un point X à l’intersection de deux simplexes Δ_i et Δ_j du maillage Δ . Le point X s’écrit donc comme combinaison convexe de sommets communs à Δ_i et Δ_j . On montre alors que $f_i(X) = f_j(X)$, où f_i et f_j désignent les champs de vecteurs affines locaux associés respectivement aux cellules Δ_i et Δ_j , ce qui prouve la continuité de f_h . □

La continuité des champs f et f_h nous permet d’en déduire le résultat suivant :

Lemme 2.1.1. f et f_h sont localement bornées par rapport à X , uniformément par rapport à u sur $\mathbb{R}^n \times \mathbb{U}_m$, i.e. :

$$\forall R > 0, \exists C_R > 0, \forall (X, u) \in B(0, R) \times \mathbb{U}_m, \|f_h(X, u)\| \leq C_R.$$

Si, de plus, f est bornée sur $\mathbb{R}^n \times \mathbb{U}_m$, alors f_h l’est aussi (avec la même borne).

Preuve. Le premier résultat du lemme 2.1.1 est une conséquence immédiate de la continuité des fonctions f et f_h sur $\mathbb{R}^n \times \mathbb{U}_m$: en effet, quel que soit $R > 0$, f et f_h sont continues sur le compact $\bar{B}(0, R) \times \mathbb{U}_m$ et donc bornées sur ce compact.

On suppose maintenant que le champ initial f est borné sur $\mathbb{R}^n \times \mathbb{U}_m$:

$$\forall (X, u) \in \mathbb{R}^n \times \mathbb{U}_m, \|f(X, u)\| \leq C.$$

Soit $(X, u) \in \mathbb{R}^n \times \mathbb{U}_m$: il existe une cellule $\Delta_i = \text{Conv}(\sigma_1, \dots, \sigma_{n+m+1})$ du maillage Δ de $\mathbb{R}^n \times \mathbb{U}_m$ telle que : $(X, u) \in \Delta_i$. (X, u) peut alors s'écrire comme combinaison convexe des sommets de Δ_i :

$$\exists (\alpha_j)_{j=1 \dots n+m+1} \in [0, 1]^{n+m+1}, \sum_{j=1}^{n+m+1} \alpha_j = 1, (X, u) = \sum_{j=1}^{n+m+1} \alpha_j \sigma_j.$$

L'approximation f_h étant affine dans la cellule Δ_i , on a : $f_h(X, u) = \sum_{j=1}^{n+m+1} \alpha_j f_h(\sigma_j)$. De plus, d'après les contraintes d'interpolation aux sommets de Δ_i , on sait que : $\forall j = 1, \dots, n+m+1$, $f_h(\sigma_j) = f(\sigma_j)$, d'où :

$$\begin{aligned} \|f_h(X, u)\| &= \left\| \sum_{j=1}^{n+m+1} \alpha_j f_h(\sigma_j) \right\| = \left\| \sum_{j=1}^{n+m+1} \alpha_j f(\sigma_j) \right\| \\ &\leq \sum_{j=1}^{n+m+1} \alpha_j \|f(\sigma_j)\| \leq \left(\sum_{j=1}^{n+m+1} \alpha_j \right) C = C. \end{aligned}$$

□

Proposition 2.1.2. f_h est lipschitzienne sur tout compact de \mathbb{R}^n par rapport à X , uniformément par rapport à u , i.e. pour tout compact Ω de \mathbb{R}^n , la restriction de f_h à $\Omega \times \mathbb{U}_m$ est lipschitzienne par rapport à X , uniformément par rapport à u :

$$\forall \Omega \Subset \mathbb{R}^n, \exists L_\Omega > 0, \forall (X, Y) \in \Omega^2, \forall u \in \mathbb{U}_m, \|f_h(X, u) - f_h(Y, u)\| \leq L_\Omega \|X - Y\|.$$

Preuve. Soit Ω un compact de \mathbb{R}^n . On introduit l'enveloppe convexe de Ω notée $\text{Conv}(\Omega)$, i.e. le plus petit convexe qui contient Ω . Puisque Ω est compact, $\text{Conv}(\Omega)$ l'est aussi.

Dans le cadre de dynamiques non contrôlées du type : $\dot{X}(t) = g(X(t))$, la proposition [44, 11.1.3] garantit que l'approximation du champ g construite par interpolation sur un maillage en simplexes $(\Delta_i)_{i \in I}$ d'un domaine compact convexe D de l'espace d'état, est continue et lipschitzienne sur D de constante $L_h = \max_{i \in I} \|A_i\|$.

Appliquée au domaine $\text{Conv}(\Omega)$ et au champ f , cette proposition nous permet de conclure que l'approximation f_h est L_Ω -lipschitzienne par rapport à X sur $\text{Conv}(\Omega)$ en posant :

$$L_\Omega = \sup_{i \in J(\text{Conv}(\Omega))} \|A_i\|$$

où $J(\text{Conv}(\Omega))$ est l'ensemble des indices des cellules Δ_i du maillage Δ qui intersectent le domaine $\text{Conv}(\Omega) \times \mathbb{U}_m$:

$$J(\text{Conv}(\Omega)) = \{i \in I ; \Delta_i \cap (\text{Conv}(\Omega) \times \mathbb{U}_m)\}.$$

De plus $Conv(\Omega) \times \mathbb{U}_m$ est le produit cartésien de deux compacts respectivement de \mathbb{R}^n et \mathbb{R}^m ; c'est donc un compact. On en déduit alors que $Conv(\Omega) \times \mathbb{U}_m$ intersecte un nombre fini de cellules du maillage Δ , soit : $card J(Conv(\Omega)) < +\infty$. La constante de Lipschitz L_Ω est donc bien définie.

□

Pour chaque contrôle $u(\cdot)$ fixé, ces propriétés assurent l'existence et l'unicité d'une solution maximale définie presque partout, du problème de Cauchy :

$$\begin{cases} \dot{X}(t) &= f_h(X(t), u(t)) \\ X(0) &= X_0 \end{cases}$$

Nous reviendrons plus en détails sur les questions d'existence et d'unicité des solutions d'un problème de Cauchy en théorie du contrôle au paragraphe 2.3.1.

2.1.3 Une autre approche : les fonctions implicites

Jusqu'à présent nous avons proposé une méthode d'approximation des dynamiques non linéaires par interpolation affine aux sommets d'un maillage donné. Cependant, les travaux que nous avons réalisé concernant la modélisation de l'activité électrique d'un neurone isolé [34], nous suggèrent une autre méthode : l'idée essentielle est de considérer chacune des équations du système non linéaire que l'on souhaite approximer et de les linéariser par morceaux séparément les unes des autres grâce à leur représentation implicite. On construit ainsi implicitement un maillage de l'espace d'état, sur lequel est définie une approximation affine par morceaux du système initial.

On suppose que la fonction f est de classe C^1 sur $\mathbb{R}^n \times \mathbb{U}_m$. On pose : $f = (f_1, \dots, f_n)$. Le système non linéaire $\dot{X}(t) = f(X(t), u(t))$ s'écrit donc également sous la forme :

$$\dot{X}_i(t) = f_i(X(t), u(t)), \quad i = 1, \dots, n,$$

où : $\forall t \in \mathbb{R}, X(t) = (X_1(t), \dots, X_n(t))$.

D'après le théorème des fonctions implicites, si la matrice jacobienne $\frac{\partial f_i}{\partial u}(x, u)$ est inversible, alors il existe une application $\varphi_i : \mathbb{R}^n \rightarrow \mathbb{R}^m$ telle que, pour un réel α donné, on ait :

$$f_i(X, u) = \alpha \Leftrightarrow u = \varphi_i(X).$$

La fonction réelle f_i est alors remplacée par la fonction : $(X, u) \mapsto \varphi_i(X) + b_i^T u(t)$, où b_i est un vecteur de \mathbb{R}^m . Il suffit alors d'appliquer les techniques de linéarisation par morceaux développées par [44] au champ de vecteurs scalaire φ_i pour en déduire une approximation affine par morceaux $\tilde{\varphi}_i$ de la fonction initiale f_i . On en déduit alors une approximation affine par morceaux du système initial (2.1) définie de la façon suivante :

$$\dot{X}(t) = \tilde{\varphi}(X) + Bu(t),$$

en posant : $B = [b_1 | \dots | b_n]^T$ et $\tilde{\varphi} = (\tilde{\varphi}_1, \dots, \tilde{\varphi}_n)$. $\tilde{\varphi}$ est une approximation affine par morceaux du champ de vecteurs non linéaire initial (2.1).

L'avantage de cette méthode réside dans le fait que l'approximation affine par morceaux est géométriquement plus proche du champ initial que celle calculée par interpolation au paragraphe 2.1.1, ce qui se traduit par une approximation de meilleure qualité des solutions du système initial. Cependant, en grande dimension, le calcul des représentations implicites des n équations du système non linéaire puis l'interpolation du champ φ rendent cette méthode algorithmiquement difficile à mettre en oeuvre.

2.2 Transformation canonique des systèmes affines locaux

Précédemment, nous avons proposé une méthode d'approximation d'un champ non linéaire par une fonction affine par morceaux sur un maillage donné. Nous nous intéressons maintenant aux systèmes affines définis dans chaque cellule du maillage Δ . Pour des raisons de lisibilité, nous omettons dans ce paragraphe l'indice de la cellule considérée. Ainsi on considère le système de contrôle affine suivant :

$$\dot{X}(t) = AX(t) + Bu(t) + c \quad (2.7)$$

où A est une matrice $n \times n$ et B une matrice $n \times m$.

Lorsque le système (2.7) n'est pas de rang plein (i.e. $rg([B \ AB \ \dots \ A^{n-1}B]) < n$), R.E. Kalman a mis en évidence l'existence d'une partie du système indépendante du contrôle, dite *incontrôlable* [56, 57, 58].

Théorème 2.2.1 ([56] Structure Canonique de Kalman). *Soient A et B deux matrices réelles de tailles respectives $n \times n$ et $n \times m$. Il existe une matrice réelle T inversible, telle que :*

$$T^{-1}AT = \begin{bmatrix} A_1 & A_2 \\ 0 & A_3 \end{bmatrix} \quad T^{-1}B = \begin{bmatrix} B_1 \\ 0 \end{bmatrix},$$

où A_1 est une matrice carrée de dimension r et B_1 une matrice de taille $r \times m$ en notant : $r = rg([B \ AB \ \dots \ A^{n-1}B]) = rg([B_1 \ A_1B_1 \ \dots \ A_1^{n-1}B_1])$.

Il existe de nombreux algorithmes numériques qui calculent la forme canonique de Kalman (entre autres : Polepack [64]). Cependant l'utilisation de ces méthodes avec des données numériques en entrée peut rendre inutile ce calcul. En effet, les situations où les matrices A et B sont connues de façon exacte, sont fréquentes et dans ce cas il est très important de pouvoir calculer exactement le rang du système affine considéré afin d'extraire la partie incontrôlable du système. C'est pour cette raison que nous proposons ici un nouvel algorithme exact de calcul de la décomposition canonique de Kalman. De surcroît, contrairement à la décomposition initiale de Kalman, la forme canonique que nous calculons, renvoie une matrice A_1 presque diagonale ce qui nous permet ultérieurement d'accélérer les calculs.

Nous commençons par rappeler le principe de la décomposition de Kalman en termes de décomposition de l'espace d'état \mathbb{R}^n . Nous décrivons ensuite notre algorithme de calcul exact de la forme de Kalman.

2.2.1 Décomposition canonique des systèmes affines

Le principe de la démonstration du théorème 2.2.1 repose sur la décomposition de l'espace d'état en deux sous-espaces : $W \oplus \bar{W}$, sur lesquels les matrices A et B s'écrivent sous la forme

canonique donnée par le théorème 2.2.1. Une méthode classique consiste à introduire le sous-espace linéaire :

$$W(A, B) = Vect(B, AB, \dots, A^{n-1}B),$$

et à vérifier que cet espace satisfait les propriétés suivantes :

- i. $Im(B) \subset W(A, B)$
- ii. $W(A, B)$ est A -invariant i.e. : $\forall x \in W(A, B), Ax \in W(A, B)$.

On démontre alors le lemme suivant :

Lemme 2.2.1. $W(A, B)$ est le plus petit sous-espace de \mathbb{R}^n qui vérifie les propriétés i. et ii..

Preuve. L'assertion i. se déduit de la définition de $W(A, B)$; ii. est une conséquence du théorème de Cayley-Hamilton.

Soit V un sous-espace de \mathbb{R}^n vérifiant les propriétés i. et ii.. On veut montrer que le sous-espace V contient nécessairement $W(A, B)$: $W(A, B) \subset V$.

Soit $y \in W(A, B)$. Par définition, $(B, AB, \dots, A^{n-1}B)$ est une famille génératrice du sous-espace $W(A, B)$; il existe donc $(x_0, \dots, x_{n-1}) \in (\mathbb{R}^n)^n$ tels que :

$$y = Bx_0 + ABx_1 + A^{n-1}Bx_{n-1}.$$

On a de plus supposé : $Im(B) \subset V$, ce qui implique : $\forall i = 1, \dots, n-1, Bx_i \in V$. Or V est A -invariant d'où : $\forall (i, j) \in [|0, n-1|]^2, A^j Bx_i \in V$. On en déduit donc : $y \in V$.

□

Il s'agit ensuite de déterminer une base (v_1, \dots, v_r) du sous-espace $W(A, B)$ que l'on complète en une base $\mathcal{V} = (v_1, \dots, v_n)$ de \mathbb{R}^n . On choisit alors comme matrice T du théorème 2.2.1 la matrice de passage de la base canonique de \mathbb{R}^n à la base \mathcal{V} calculée. Il reste alors à vérifier que $T^{-1}AT$ et $T^{-1}B$ sont bien sous forme canonique.

On note $\mathcal{E} = (e_1, \dots, e_n)$ la base canonique de \mathbb{R}^n . Par définition de la matrice de passage T , $T^{-1}AT$ est la matrice A exprimée dans la base canonique. On pose :

$$T^{-1}AT = \left[\begin{array}{cc|c} Av_1 \dots Av_r & Av_{r+1} \dots Av_n & v_1 \\ \hline A_1 & A_2 & \vdots \\ \hline A_4 & A_3 & v_r \\ & & \vdots \\ & & v_{r+1} \\ & & \vdots \\ & & v_n \end{array} \right]$$

D'après la propriété ii., $W(A, B)$ est A -invariant ce qui implique : $\forall i = 1, \dots, r, Av_i \in W(A, B) = Vect(v_1, \dots, v_r)$, soit : $A_4 = 0$.

Calculons maintenant la matrice $T^{-1}B$: c'est par définition la matrice formée des vecteurs colonnes $\{T^{-1}Be_i ; i = 1, \dots, n\}$ exprimés dans la base canonique :

$$T^{-1}B = [T^{-1}Be_1 \mid \dots \mid T^{-1}Be_n].$$

Or, d'après la propriété i., les vecteurs $Be_i \in Im(B)$ appartiennent à l'espace $W(A, B)$, soit :

$$\forall i \in \{1, \dots, n\}, T^{-1}Be_i \in T^{-1}W(A, B).$$

Par construction, (v_1, \dots, v_r) est une base de $W(A, B)$; $(T^{-1}v_1, \dots, T^{-1}v_r)$ est donc une base du sous-espace $T^{-1}W(A, B)$. De plus, T est la matrice de passage de la base canonique \mathcal{E} à la base \mathcal{V} , ce qui nous donne les relations suivantes :

$$\forall i \in \{1, \dots, n\}, T^{-1}v_i = e_i.$$

On en déduit alors que la famille (e_1, \dots, e_r) est une base du sous-espace $T^{-1}W(A, B)$. Tout vecteur de $T^{-1}W(A, B)$ et donc en particulier les $T^{-1}Be_i$, s'écrivent comme combinaison linéaire des e_i , $i = 1, \dots, r$ dont les $(n - r)$ dernières composantes sont nulles. On conclut alors que $T^{-1}B$ est bien de la forme cherchée.

2.2.2 Algorithmes de décomposition canonique par blocs

Nous décrivons maintenant un nouvel algorithme de calcul de la forme canonique de Kalman. L'idée est d'utiliser une version par blocs des algorithmes d'algèbre linéaire, comme cela a été fait dans [31]. Une telle approche nous permet d'améliorer la complexité algébrique des calculs et de traiter des problèmes de plus grande dimension plus rapidement. Elle permet en effet non seulement un calcul exact efficace du rang de tout système linéaire, mais aussi l'utilisation de la décomposition LQUP [50, 32] (aujourd'hui aussi rapide que les routines numériques) pour effectuer la décomposition canonique.

Les algorithmes présentés ci-après et réalisés en collaboration avec C. Pernet, sont décrits de manière beaucoup plus complète dans [66]. Ils sont également décrits dans [35, 36]. Les preuves des algorithmes décrits dans la suite, sont données en annexe B.

On introduit la matrice de Kalman :

$$K(A, B) = [B \mid AB \mid \dots \mid A^{n-1}B].$$

L'algorithme de base du calcul de la forme de Kalman consiste à décomposer la matrice $K(A, B)$ via une élimination de Gauss, de façon à faire apparaître une base de l'espace $W(A, B)$ défini précédemment. On en déduit alors la matrice de transformation T du théorème 2.2.1 ainsi que l'expression du système affine considéré sous forme canonique.

D'après une idée de G. Villard [83], la complexité de cet algorithme de base peut être améliorée d'un facteur n en utilisant l'algorithme rapide de Keller-Gehrig [59]; le calcul de la matrice $K(A, B)$ étant très coûteux, l'idée est de traiter virtuellement $K(A, B)$ et de ne jamais la calculer explicitement. L'algorithme de calcul de la forme de Kalman se décompose alors en deux étapes :

1. Définir une matrice \overline{K} , appelée *matrice compressée de Krylov*, calculée en sélectionnant au plus n colonnes indépendantes de K .
2. Calculer la forme de Kalman en utilisant \overline{K} .

La complexité algébrique de cet algorithme est alors en $\mathcal{O}(n^\omega \log n)$ où ω désigne l'exposant intervenant dans la complexité du produit matriciel; les travaux de D. Coppersmith et S. Winograd [22] ont montré que cet exposant est compris entre 2 et 2,3755. Théoriquement, le meilleur algorithme connu a un exposant proche de 2,3755; les algorithmes classiques utilisables ont une complexité en $\mathcal{O}(n^3)$ ou $\mathcal{O}(n^{\log_2 7})$.

Après avoir détaillé le calcul de la forme de Kalman par l'algorithme de Keller-Gehrig, nous présentons un second algorithme basé sur d'autres techniques de calcul du polynôme caractéristique présentées dans [33]. Bien que la complexité de ce second algorithme ne soit qu'en $\mathcal{O}(n^3)$, celui-ci se révèle plus rapide en pratique.

Remarque 2.2.1. *Dans la suite, on note r le rang de la matrice de Kalman $K(A, B)$ de taille $n \times nm$. On a donc nécessairement : $r \leq n$.*

a. Calcul de la forme de Kalman en utilisant l'algorithme de Keller-Gehrig

Définition de la matrice compressée \overline{K} de Krylov.

La première étape de l'algorithme est de calculer la matrice compressée \overline{K} de Krylov. C'est une matrice de taille $r \times n$ obtenue en sélectionnant r colonnes indépendantes dans $K(A, B)$. Le principe du calcul de \overline{K} est le suivant : on introduit les vecteurs colonnes $b_i, i = 1, \dots, m$, de la matrice $B : B = [b_1 \mid \dots \mid b_m]$ et on considère la matrice de Krylov K engendrée par ces vecteurs :

$$K = [b_1 \mid \dots \mid A^{n-1}b_1 \mid \dots \mid b_m \mid \dots \mid A^{n-1}b_m].$$

La matrice K correspond à l'ordre de ses vecteurs colonnes près à la matrice de Kalman $K(A, B)$ et est donc de même rang. Le calcul de la matrice compressée est basée sur l'observation suivante : si un vecteur colonne $A^i b_j$ est linéairement dépendant de tous les vecteurs colonnes le précédant dans K , alors nécessairement les itérées $A^k b_j$ ($k > i$) de ce vecteur le sont également. Par conséquent, \overline{K} peut s'écrire sous la forme :

$$\overline{K} = [b_1 \mid \dots \mid A^{d_1} b_1 \mid \dots \mid b_p \mid \dots \mid A^{d_p} b_p], \quad (2.8)$$

où $d_i \in \{0, \dots, n\}$ est la plus grande puissance de A , telle que le vecteur $A^{d_i} b_i$ soit linéairement indépendant de tous les vecteurs colonnes le précédant dans K .

Comme le montre la figure 2.2, la matrice compressée \overline{K} est construite au fur et à mesure, en parcourant les vecteurs colonnes de la matrice K : la matrice \overline{K} est initialisée à $[b_1]$. On ajoute ensuite successivement les puissances $A^i b_1$ de b_1 , tant que le rang du système triangulé augmente de un à chaque itération. Si le rang n'augmente pas, cela signifie que nous avons trouvé une itérée $A^{d_1+1} b_1$ linéairement dépendante des précédentes. La matrice \overline{K} est alors ré-initialisée à $[b_1, \dots, A^{d_1} b_1]$ et on lui applique le même processus, en parcourant la liste des itérées $A^i b_2$ pour $i \in \{1, \dots, n-1\}$. L'algorithme se termine quand le rang du système est égal à n , ou alors quand la liste des vecteurs colonnes de K a été entièrement parcourue.

À chaque itération, on effectue une élimination de Gauss sur un système triangulé plus un vecteur, soit un coût en $\mathcal{O}(n^2)$ opérations. Étant donné qu'on effectue autant d'itérations qu'il y a de colonnes linéairement indépendantes, la complexité totale est en $\mathcal{O}((r+m)n^2)$. Cet algorithme améliore donc déjà la complexité du calcul de la forme de Kalman d'un facteur n par rapport aux méthodes classiques où la matrice $K(A, B)$ est complètement calculée.

Calcul de la matrice compressée \overline{K} par l'algorithme de Keller-Gehrig.

La complexité de l'algorithme précédent peut encore être améliorée en utilisant l'algorithme de Keller-Gehrig. Le calcul de la matrice \overline{K} est basé sur les remarques suivantes :

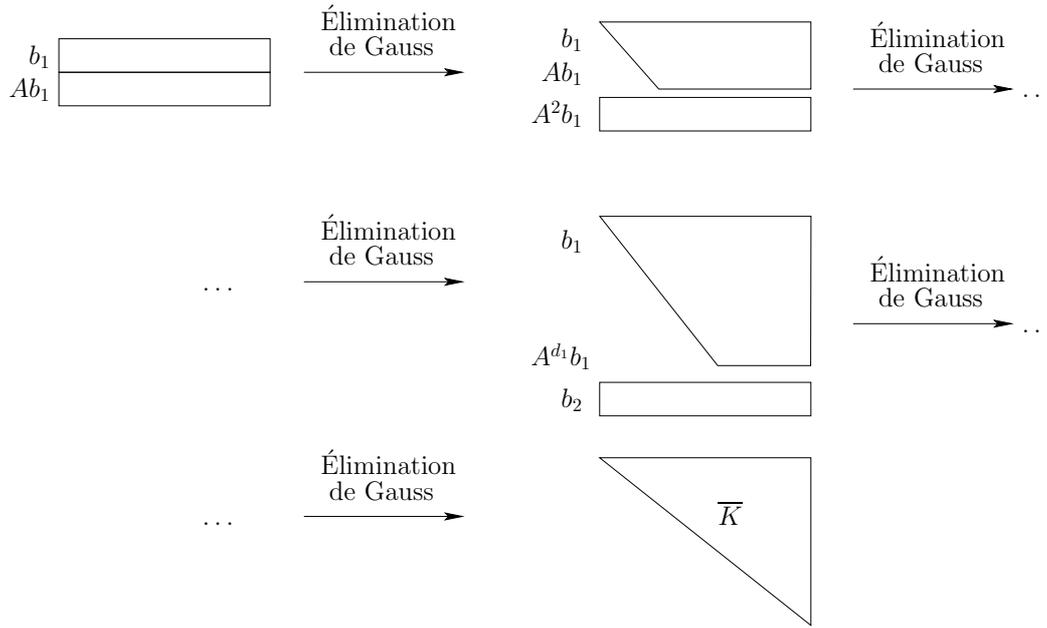


FIG. 2.2 – Principe de base de calcul de la matrice compressée de Krylov

- Une élimination de Gauss par blocs permet de connaître la position des colonnes linéairement indépendantes dans une matrice donnée. Keller-Gehrig a introduit une élimination dite *step-form* avec une complexité en $\mathcal{O}(n^\omega)$ [59].
- Il est possible de calculer toutes les itérées de la matrice B en seulement $\mathcal{O}(n^\omega \log_2 n)$ opérations. On calcule, en effet, les itérées carrées de la matrice A , i.e. les matrices :

$$A, A^2, \dots, A^{2^i}, \dots, A^{2^{\lceil \log_2 n \rceil - 1}}$$

Ainsi, toutes les itérées d'un vecteur colonne b donné de B sont calculées grâce au schéma suivant :

$$V_0 = b, \quad V_{i+1} = [V_i | A^{2^i} V_i].$$

Nous pouvons alors construire un algorithme de calcul de la matrice compressée de Krylov, inspiré sur l'algorithme de Keller-Gehrig :

1. On pose : $V_0 := B$;
2. Pour i variant de 0 à $(\log_2 n) - 1$,
 - (a) Calculer le produit $A^{2^i} V_i$, soit $\mathcal{O}(n^\omega)$ opérations.
 - (b) Former la matrice W_i obtenue en intercalant les colonnes de V_i et de $A^{2^i} V_i$, de façon à avoir, dans l'ordre, les itérées de b_1 , puis celles de b_2 et ainsi de suite.
 - (c) Former la matrice V_{i+1} , obtenue en sélectionnant par l'élimination de Gauss par blocs, les colonnes linéairement indépendantes de W_i , soit $\mathcal{O}(n^\omega)$ opérations.

Remarque 2.2.2. L'étape (b) de l'algorithme décrit ci-dessus, n'est pas nécessaire pour le calcul de la forme de Kalman. Le fait de réordonner les vecteurs colonnes à chaque itération, nous permet seulement d'obtenir des blocs de matrices compagnons dans la forme de Kalman, ce qui est utile pour le calcul du polynôme caractéristique [66].

Cet algorithme calcule donc la matrice compressée de Krylov de la matrice de Kalman $K(A, B)$ en $\lceil \log_2 n \rceil$ itérations i.e. en seulement $\mathcal{O}(n^\omega \log_2 n)$ opérations.

Calcul de la forme de Kalman

La dernière partie de l'algorithme consiste à retrouver la matrice de transformation T du théorème 2.2.1. Par construction, les r vecteurs colonnes de la matrice compressée \overline{K} forment une base du sous-espace A -invariant $W(A, B)$. L'idée est maintenant de compléter cette base en une base de l'espace tout entier.

La matrice T s'obtient en complétant la matrice \overline{K} en une matrice inversible selon la technique utilisée dans [33, théorème 2.1] : on calcule la factorisation LUP de la matrice \overline{K}^T :

$$\overline{K}^T = L [U_1 \mid U_2] P.$$

où L est une matrice $r \times r$ triangulaire inférieure par blocs, U_1 est une matrice $r \times r$ triangulaire supérieure inversible et P est une permutation d'ordre n . On remplace ensuite la matrice $U = [U_1 \mid U_2]$ par la matrice inversible $\left[\begin{array}{c|c} U_1 & U_2 \\ \hline 0 & I_{n-r} \end{array} \right]$ et la matrice L par $\left[\begin{array}{c|c} L & 0 \\ \hline 0 & I_{n-r} \end{array} \right]$, ce qui nous donne la matrice inversible cherchée :

$$T^T = \left[\begin{array}{c|c} L & 0 \\ \hline 0 & I_{n-r} \end{array} \right] \left[\begin{array}{c|c} U_1 & U_2 \\ \hline 0 & I_{n-r} \end{array} \right] P = \left[\overline{K} \mid P^T \left[\begin{array}{c} 0 \\ I_{n-r} \end{array} \right] \right]^T,$$

et la décomposition canonique de Kalman (cf algorithme 1).

On obtient alors le résultat suivant (démontré en annexe B, paragraphe B.1.2) :

Théorème 2.2.2. *Soient $A \in \mathcal{M}_{n,n}(\mathbb{R})$ et $B \in \mathcal{M}_{n,m}(\mathbb{R})$. L'algorithme 1 est correct et nécessite $\mathcal{O}(n^\omega \log n)$ opérations arithmétiques en utilisant la compression de Keller-Gehrig.*

b. Calcul de la forme de Kalman par l'algorithme LU-Krylov

L'algorithme présenté ci-après et inspiré de l'algorithme 2.2 de [33], calcule une matrice \tilde{K} (qui n'est pas une matrice compressée de Krylov) mais qui joue le même rôle que \overline{K} pour le calcul de la matrice de passage T cherchée.

Cet algorithme repose également sur le calcul des itérées de Krylov et autant que possible sur le produit matriciel. Les itérées de Krylov étant calculées à partir de produits matrice-vecteur, la complexité est en $\mathcal{O}(n^3)$. Toutefois cet algorithme se révèle plus rapide en pratique que celui de Keller-Gehrig pour le calcul du polynôme caractéristique [33] et donc a fortiori pour celui de la forme de Kalman.

La matrice de passage T est obtenue alors par le même procédé que précédemment en posant :

$$T = \left[\tilde{K} \mid P^T \left[\begin{array}{c} 0 \\ I_{n-r} \end{array} \right] \right]$$

et l'algorithme 1 s'applique alors de la même façon. On démontre en annexe B, paragraphe B.2.2 le résultat suivant :

Théorème 2.2.3. *Soit A une matrice $n \times n$ et B une matrice $n \times m$. L'algorithme 1 de calcul de la forme de Kalman est correct et nécessite $\mathcal{O}(n^3)$ opérations en utilisant l'algorithme 2.*

Algorithme 1 Forme de Kalman par blocs**Données :** A une matrice $n \times n$, B une matrice $n \times m$, définies sur un corps**Sortie :** r, T, A_1, A_2, A_3, B_1 (cf théorème 2.2.1)

- 1: $(\overline{K}, r) = \text{CompressedKrylovMatrix}(A, B)$ {par l'algorithme de Keller-Gehrig}
- 2: **si** ($r=n$) **alors**
- 3: Renvoyer $(n, Id, A, \emptyset, \emptyset, B)$
- 4: **sinon**
- 5: $(L, [U_1 U_2], P) = \text{LUP}(\overline{K}^T)$
- 6: $T = \left[\begin{array}{c|c} \overline{K} & P^T \begin{bmatrix} 0 \\ I_{n-r} \end{bmatrix} \end{array} \right]$
- 7: $B_1 = L^{-T} U_1^{-T} P B$
- 8: $A' = P A^T P^T = \begin{bmatrix} A'_{11} & A'_{12} \\ A'_{21} & A'_{22} \end{bmatrix}$
- 9: $C_1 = L^{-T} U_1^{-T} A'_{12}$; $C_2 = A'_{22} - U_2^T U_1^{-T} A'_{12}$
- 10: **pour tout** j **faire**
- 11: Soit t_j les indices des colonnes dans \overline{K} de la dernière itérée l_j du $j^{\text{ème}}$ bloc.
- 12: $m_j = l_j L_{1\dots t_j, 1\dots t_j}^{-1}$
- 13: **fin pour**
- 14: Construire la matrice polycyclique H en plaçant chaque vecteur colonne m_j à l'indice de colonne t_j et en ajoutant 1 sur la sous-diagonale de toutes les autres colonnes.
- 15: Renvoyer (r, T, H, C_1, C_2, B_1)
- 16: **fin si**

Remarque 2.2.3. *En comparant les algorithmes 1 et 2, on s'aperçoit que la dernière élimination dans l'algorithme 2 correspond à la première élimination dans l'algorithme 1. L'idée est donc de fusionner les deux algorithmes afin d'éviter les opérations inutiles. On calcule ainsi simultanément la matrice compressée et la forme de Kalman. La complexité reste cependant en $\mathcal{O}(n^3)$. L'algorithme complet est détaillé dans [66].*

En conclusion, l'implémentation et la preuve constructive de la décomposition de Kalman sont basées sur la factorisation LQUP et le calcul matriciel par blocs. Le recours aux primitives BLAS (Basic Linear Algebra Subroutines), même pour des calculs symboliques [32], nous permet d'améliorer le temps de calcul et de concurrencer les performances numériques.

2.2.3 Transformation locale du système

Revenons maintenant au système affine par morceaux : $\dot{X}(t) = f_h(X(t), u(t))$ défini au début de ce chapitre. D'après le théorème de Kalman, dans toute cellule Δ_i du maillage de l'espace état-contrôle, le système affine associé : $\dot{X}(t) = A_i X(t) + B_i u(t) + c_i$ peut être remplacé par un système affine canonique de la forme :

$$\dot{Y}(t) = \begin{bmatrix} A_{i,1} & A_{i,2} \\ 0 & A_{i,3} \end{bmatrix} Y(t) + \begin{bmatrix} B_{i,1} \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} c_{i,1} \\ c_{i,2} \end{bmatrix}, \quad (2.9)$$

en posant : $T_i^{-1} c_i = [c_{i,1} \mid c_{i,2}]^T$ et en effectuant le changement de variable d'état : $Y(t) = T_i^{-1} X(t)$, où T_i est la matrice de passage du théorème 2.2.1.

Algorithme 2 Matrice de passage de Kalman par LU-Krylov**Données :** A une matrice $n \times n$, B une matrice $n \times m$ définies sur un corps**Sortie :** $(\tilde{K}, \text{rang}(\tilde{K}))$

- 1: Sélectionner la première colonne b de B .
- 2: $\begin{cases} K_1 = [b & Ab & A^2b & \dots] \\ (L, [U_1|U_2], P) = \text{LUP}(K_1^T), r_1 = \text{rang}(K_1) \end{cases}$
 {La matrice K_1 est calculée à la volée : on calcule ainsi au plus $2r_1$ colonnes. La matrice U_1 est de taille $r_1 \times r_1$.}
- 3: **si** ($r_1 = n$) **alors**
- 4: Renvoyer (K_1, r_1)
- 5: **sinon**
- 6: $A' = PAP^T = \begin{bmatrix} A'_{11} & A'_{12} \\ A'_{21} & A'_{22} \end{bmatrix}$ où A'_{11} est $r_1 \times r_1$.
- 7: $A_R = A'_{22} - U_2^T U_1^{-T} A'_{12}$
 {La matrice A_R est appelée *complément de Schur* de A'_{11} dans A .}
- 8: $B' = \begin{bmatrix} L^{-T} U_1^{-T} & 0 \\ -U_2^T U_1^{-T} & I \end{bmatrix} PB$
- 9: Calculer la matrice de permutation Q telle que $B'Q = \begin{bmatrix} X & Y \\ 0 & B_R \end{bmatrix}$
- 10: Appel récursif de $(K_2, r_2) = \text{LUCKM}(A_R, B_R)$
- 11: $\tilde{K} = \begin{bmatrix} K_1 & P^T \begin{bmatrix} 0 \\ K_2 \end{bmatrix} \end{bmatrix}$
- 12: Renvoyer $(\tilde{K}, r_1 + r_2)$
- 13: **fin si**

De plus, toute matrice T_i est associée à une cellule Δ_i donnée du maillage Δ . Cette remarque signifie en particulier qu'il n'est pas possible de définir une approximation affine par morceaux canonique du champ non linéaire initial f .

En effet, le changement de variable induit par le théorème de Kalman dans chaque cellule Δ_i du maillage, s'écrit :

$$\begin{bmatrix} Y \\ u \end{bmatrix} = \tilde{T}_i \begin{bmatrix} X \\ u \end{bmatrix} \text{ en posant : } \tilde{T}_i = \begin{bmatrix} T_i^{-1} & 0 \\ 0 & I_m \end{bmatrix}. \quad (2.10)$$

Comme le montre l'exemple 2.2.1, sauf cas particuliers (les matrices \tilde{T}_i sont par exemple toutes identiques), la famille $(\tilde{T}_i \Delta_i)_{i \in I}$ n'est pas un maillage de $\mathbb{R}^n \times \mathbb{U}_m$.

Exemple 2.2.1. On se place dans le cas où $n = 2$ et $m = 1$. Le but de cet exemple est de montrer qu'on ne peut en général pas définir de fonction affine par morceaux canonique.

Considérons deux cellules adjacentes Δ_1 et Δ_2 d'un maillage supposé de l'espace état-contrôle $\mathbb{R}^2 \times [0, 1]$:

$$\begin{aligned} \Delta_1 &= \text{Conv}\{(0, 0, 0), (1, 0, 0), (1, 1, 0), (1, 0, 1)\}, \\ \Delta_2 &= \text{Conv}\{(0, 0, 0), (0, 1, 0), (1, 1, 0), (1, 0, 1)\}, \end{aligned}$$

sur lesquelles on définit une fonction f_h continue, affine par morceaux vérifiant les contraintes d'interpolation du paragraphe 2.1 :

$$f_h(X, u) = \begin{cases} f_1(X, u) & \text{si } (X, u) \in \Delta_1 \\ f_2(X, u) & \text{si } (X, u) \in \Delta_2 \end{cases}$$

avec : $f_1(X, u) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} X + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$ et $f_2(X, u) = \begin{bmatrix} 0 & 1 \\ 2 & 0 \end{bmatrix} X + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$.

L'idée est alors de calculer les matrices de changement de variable \tilde{T}_1 et \tilde{T}_2 induite par la décomposition de Kalman, respectivement dans les cellules Δ_1 et Δ_2 . On en déduit alors des fonctions affines canoniques définies sur les images respectives $\tilde{T}_1^{-1}\Delta_1$ et $\tilde{T}_1^{-1}\Delta_2$ de Δ_1 et Δ_2 . Pour pouvoir définir une nouvelle fonction affine par morceaux, il faudrait que les cellules $\tilde{\Delta}_1 = \tilde{T}_1^{-1}\Delta_1$ et $\tilde{\Delta}_2 = \tilde{T}_2^{-1}\Delta_2$ soient ou bien d'intersection vide, ou bien adjacentes.

- Le système $\dot{X}(t) = f_2(X, u)$ est déjà sous forme canonique. Il est donc inutile de calculer la décomposition de Kalman de ce système et on a :

$$\tilde{T}_2 = \begin{bmatrix} I_2 & 0 \\ 0 & 1 \end{bmatrix} = I_3 \quad \text{et} \quad \tilde{\Delta}_2 = \Delta_2.$$

- La matrice de Kalman associée au système $\dot{X}(t) = f_1(X, u)$ est égale à : $K_1 = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}$ et est de rang 1 ; on effectue alors le calcul de la décomposition canonique du système, ce qui nous donne :

$$\tilde{T}_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{et} \quad \tilde{\Delta}_1 = \text{Conv}((0, 0, 0), (0, 1, 0), (1, 1, 0), (0, 1, 1)).$$

La figure 2.2.1 montre que les intérieurs des simplexes $\tilde{\Delta}_1$ et $\tilde{\Delta}_2$ sont d'intersection non vide, ce qui interdit la définition d'un maillage contenant ces deux cellules.

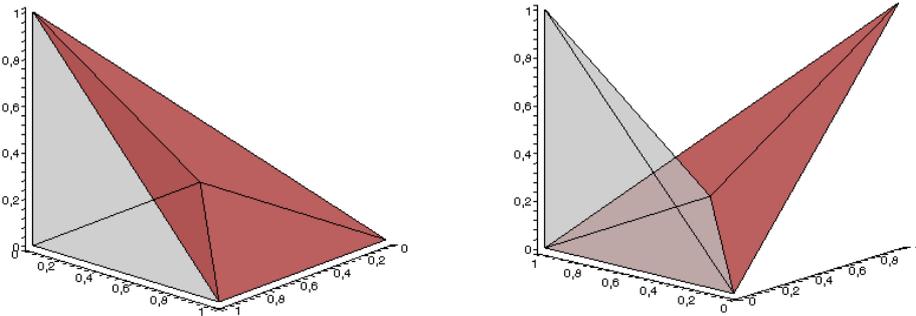


FIG. 2.3 – Les cellules Δ_1 et Δ_2 (à gauche) et leurs images après transformation canonique (à droite)

2.3 Convergence du schéma d'interpolation

Dans les paragraphes précédents, nous avons développé des outils pour l'approximation de champs de vecteurs non linéaires par des fonctions affines par morceaux. Nous nous intéressons

maintenant à l'évaluation de l'erreur d'interpolation commise ainsi qu'à la convergence de l'approximation.

Concernant l'approximation des solutions de systèmes non linéaires autonomes du type : $\dot{x}(t) = g(x(t))$, les travaux d'A. Girard [44] proposent une étude détaillée de l'erreur d'approximation sur un domaine D de l'espace d'état sous certaines hypothèses de régularité de la fonction g . Par exemple si g est lipschitzienne sur D , l'erreur d'interpolation est d'ordre $O(h)$ et est donc linéaire en la taille h du maillage. En faisant des hypothèses supplémentaires sur g , on montre que l'approximation est d'ordre supérieur (cf [44, chapitre 11]).

En théorie du contrôle, du fait du manque de régularité du contrôle, de telles hypothèses de régularité sur le champ de vecteurs non linéaire sont difficiles à obtenir : une fois le contrôle $u(\cdot)$ fixé, tout système de contrôle : $\dot{X}(t) = g(X(t), u(t))$ peut s'écrire sous la forme d'un système différentiel ordinaire non autonome :

$$\dot{X}(t) = G(t, X(t)), \quad (2.11)$$

en posant : $\forall t \geq 0, G(t, X(t)) = g(X(t), u(t))$. Le contrôle $u(\cdot)$ étant supposé seulement mesurable, la continuité même du champ $G : (t, X) \mapsto g(X, u(t))$ n'est donc pas garantie.

Nous commençons donc notre étude par une mise au point sur les différentes hypothèses de régularité choisies sur la fonction f . Selon chacune de ces hypothèses, nous étudions ensuite l'erreur d'interpolation due au schéma d'approximation présenté au paragraphe 2.1. Nous terminons par des résultats de convergence des solutions du système affine par morceaux vers les solutions du système non linéaire.

2.3.1 Choix des hypothèses de régularité

Le choix de ces hypothèses est lié à l'application du théorème Cauchy-Lipschitz à la théorie du contrôle. Pour des systèmes du type : $\dot{x}(t) = g(x(t))$, le théorème classique affirme l'existence et l'unicité d'une solution maximale définie presque partout pourvu que la fonction g soit continue et localement lipschitzienne par rapport à x . En théorie du contrôle, il s'agit, pour un contrôle $u(\cdot)$ fixé, de vérifier que la fonction $(t, x) \rightarrow f(x, u(t))$ satisfait ces hypothèses.

La première chose à faire est donc d'énoncer une version générale du théorème de Cauchy-Lipschitz, adaptée à la théorie du contrôle. Ce problème a été traité de façon très complète dans [74, appendice C] et [81, chapitre 11]. Nous en rappelons ici les éléments essentiels nécessaires à notre étude.

Définition du problème de Cauchy

Comme nous l'avons remarqué en introduction du paragraphe 2.3, pour un contrôle $u(\cdot)$ fixé, le système non linéaire (2.1) peut s'écrire sous la forme d'un système différentiel ordinaire non autonome en introduisant la fonction $F : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ définie par la relation :

$$\forall t \geq 0, F(t, X(t)) = f(X(t), u(t)).$$

On peut alors définir un nouveau problème de Cauchy :

$$\begin{cases} \dot{X}(t) = F(t, X(t)) \\ X(0) = X_0 \end{cases} . \quad (2.12)$$

Une solution de ce problème est un couple $(J, X(\cdot))$ où J est un intervalle de \mathbb{R} contenant 0 et $X(\cdot)$ est une fonction continue, dérivable par morceaux sur J , vérifiant (2.12) presque partout sur J . On a alors le théorème suivant :

Théorème 2.3.1 ([81, th 11.1.1] **Théorème de Cauchy-Lipschitz**). *On suppose que la fonction $F : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ vérifie les hypothèses suivantes :*

1. *F est localement lipschitzienne par rapport à X au sens suivant :*

$$\forall X \in \mathbb{R}^n, \exists r > 0, \exists k \in L_{loc}^1(\mathbb{R}, \mathbb{R}^+), \\ \forall t \in \mathbb{R}, \forall (X_1, X_2) \in B(X, r)^2, \|F(t, X_2) - F(t, X_1)\| \leq k(t) \|X_2 - X_1\|,$$

2. *F est localement intégrable par rapport à t , i.e. :*

$$\forall X \in \mathbb{R}^n, \exists \beta \in L_{loc}^1(\mathbb{R}, \mathbb{R}^+), \forall t \in \mathbb{R}, \|F(t, X)\| \leq \beta(t).$$

Alors, pour toute donnée initiale $(t_0, X_0) \in \mathbb{R} \times \mathbb{R}^n$, il existe une unique solution maximale du problème de Cauchy (2.12).

De plus, sous les hypothèses du théorème 2.3.1, la solution du problème de Cauchy considéré peut s'écrire de façon équivalente sous forme intégrale :

$$\forall t \in J, X(t) = X_0 + \int_0^t F(s, X(s)) ds = X_0 + \int_0^t f(X(s), u(s)) ds. \quad (2.13)$$

Choix des hypothèses sur f

Le problème est maintenant de choisir des hypothèses sur la fonction f de façon à ce que, pour un contrôle admissible u fixé, la fonction $F : (t, X) \mapsto f(X, u(t))$ associée vérifie les conditions du théorème 2.3.1.

D'après la définition du problème de contrôle non linéaire, le champ de vecteurs f est supposé continu sur $\mathbb{R}^n \times \mathbb{U}_m$. En particulier, pour tout $X \in \mathbb{R}^n$ fixé, $f(X, \cdot)$ est continue sur le compact \mathbb{U}_m , ce qui implique qu'elle est bornée (et atteint ses bornes) sur \mathbb{U}_m :

$$\forall X \in \mathbb{R}^n, \exists M_X > 0, \forall u \in \mathbb{U}_m, \|f(X, u)\| \leq M_X. \quad (2.14)$$

On peut alors en déduire que la fonction $F : (t, X) \mapsto f(X, u(t))$ est localement intégrable par rapport à t . En effet, grâce à l'inégalité (2.14), on majore $\|F(t, X)\|$ par la constante M_X indépendante du temps et donc localement intégrable sur \mathbb{R} :

$$\forall X \in \mathbb{R}^n, \forall t \in \mathbb{R}, \|F(t, X)\| = \|f(X, u(t))\| \leq M_X.$$

Par conséquent, l'hypothèse clé du théorème 2.3.1 est l'hypothèse 1. À partir de maintenant, nous allons donc considérer différents jeux d'hypothèses sur la régularité de la fonction f permettant de vérifier les conditions du théorème de Cauchy-Lipschitz :

Hypothèse a $f : \mathbb{R}^n \times \mathbb{U}_m \rightarrow \mathbb{R}^n$ est L_a -lipschitzienne sur $\mathbb{R}^n \times \mathbb{U}_m$, au sens où :

$$\forall ((X_1, u_1), (X_2, u_2)) \in (\mathbb{R}^n \times \mathbb{U}_m)^2, \|f(X_2, u_2) - f(X_1, u_1)\| \leq L_a[\|X_2 - X_1\| + \|u_2 - u_1\|]$$

Hypothèse b f est continue, globalement L_b -lipschitzienne par rapport à X , uniformément par rapport à u , i.e. :

$$\forall u \in \mathbb{U}_m, \forall (X_1, X_2) \in (\mathbb{R}^n)^2, \|f(X_2, u) - f(X_1, u)\| \leq L_b \|X_2 - X_1\|.$$

Hypothèse c f est de classe C^1 sur $\mathbb{R}^n \times \mathbb{U}_m$.

Hypothèse d f est continue, lipschitzienne sur tout compact par rapport à X , uniformément par rapport à u i.e. :

$$\forall \Omega \Subset \mathbb{R}^n, \exists L_\Omega > 0, \forall (X_1, X_2) \in \Omega^2, \|f(X_2, u) - f(X_1, u)\| \leq L_\Omega \|X_2 - X_1\|.$$

On remarque alors que ces quatre hypothèses forment un enchaînement logique :

$$\begin{aligned} \text{Hypothèse a} &\Rightarrow \text{Hypothèse b} \Rightarrow \text{Hypothèse d} \\ \text{Hypothèse c} &\Rightarrow \text{Hypothèse d} \end{aligned}$$

La première série d'implications ne pose aucun problème. La seconde, moins immédiate, est une conséquence du lemme suivant :

Lemme 2.3.1. *Si f est de classe C^1 sur $\mathbb{R}^n \times \mathbb{U}_m$, alors f est lipschitzienne sur tout compact par rapport à X , uniformément par rapport à u .*

Preuve. Soient Ω un compact de \mathbb{R}^n et $u \in \mathbb{U}_m$. Par hypothèses, la fonction f est de classe C^1 sur $\mathbb{R}^n \times \mathbb{U}_m$. Par conséquent, sa dérivée partielle $\frac{\partial f}{\partial X}(\cdot, u)$ est continue sur \mathbb{R}^n et donc en particulier sur le compact Ω . On en déduit alors qu'elle est bornée sur Ω :

$$\exists L_\Omega > 0, \forall \tilde{X} \in \Omega, \left\| \frac{\partial f}{\partial X}(\tilde{X}, u) \right\| \leq L_\Omega.$$

Grâce à l'inégalité des accroissements finis [19, th 3.3.2.], on peut conclure :

$$\forall (X_1, X_2) \in \Omega^2, \|f(X_2, u) - f(X_1, u)\| \leq L_\Omega \|X_2 - X_1\|.$$

□

Ainsi dans la suite, il suffit donc de démontrer les résultats de convergence selon l'hypothèse d pour qu'ils soient vrais selon les hypothèses a, b et c. En particulier, les conditions du théorème 2.3.1 sont vérifiées quelle que soit l'hypothèse considérée.

2.3.2 Calcul de l'erreur d'interpolation

Nous nous intéressons maintenant au calcul de l'erreur d'interpolation de l'approximation f_h construite au paragraphe 2.1 en fonction de la régularité du champ de vecteurs initial f .

La méthode est la suivante : nous évaluons dans un premier temps l'erreur d'interpolation commise au niveau d'une cellule $\Delta_{q'}$ du maillage Δ de l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$. Si l'erreur ainsi calculée ne dépend pas de la cellule $\Delta_{q'}$, alors nous en déduisons l'erreur d'interpolation globale sur $\mathbb{R}^n \times \mathbb{U}_m$; nous obtenons sinon une évaluation de l'erreur sur tout compact $\Omega \times \mathbb{U}_m$ de l'espace état-contrôle.

Proposition 2.3.1 (Erreur d'interpolation).

i. Si f est L_a -lipschitzienne sur $\mathbb{R}^n \times \mathbb{U}_m$ (hypothèse a), alors [44, proposition 11.1.1] :

$$\sup_{(X,u) \in \mathbb{R}^n \times \mathbb{U}_m} \|f(X, u) - f_h(X, u)\| \leq \frac{4L_a(n+m)}{n+m+1} h.$$

ii. Si f est de classe C^1 sur $\mathbb{R}^n \times \mathbb{U}_m$ (hypothèse c), alors pour tout compact Ω de \mathbb{R}^n , il existe $L_\Omega > 0$, tel que :

$$\sup_{(X,u) \in \Omega \times \mathbb{U}_m} \|f(X, u) - f_h(X, u)\| \leq \frac{4L_\Omega(n+m)}{n+m+1} h,$$

$$\text{en posant : } L_\Omega = \max_{q' \in I / \Delta_{q'} \cap (\Omega \times \mathbb{U}_m) \neq \emptyset} \left(\sup_{(X,u) \in \Delta_{q'}} \|Df\| \right).$$

Preuve. Cette démonstration repose sur l'évaluation de l'erreur d'interpolation commise au niveau d'une cellule $\Delta_{q'}$ du maillage état-contrôle. Commençons donc par faire le point sur les propriétés de f dans $\Delta_{q'}$:

– Si f vérifie l'hypothèse a, alors on a :

$$\forall ((X_1, u_1), (X_2, u_2)) \in \Delta_{q'}, \|f(X_2, u_2) - f(X_1, u_1)\| \leq L_a [\|X_2 - X_1\| + \|u_2 - u_1\|].$$

– Si f vérifie l'hypothèse c, alors sa différentielle Df existe et est continue sur $\mathbb{R}^n \times \mathbb{U}_m$ et donc en particulier sur le compact $\overline{\Delta_{q'}}$. Elle est donc bornée sur $\overline{\Delta_{q'}}$:

$$\exists L_{\Delta_{q'}} > 0, \forall (X, u) \in \Delta_{q'}, \|Df(X, u)\| \leq L_{\Delta_{q'}}.$$

Grâce à l'inégalité des accroissements finis, on déduit :

$$\forall ((X_1, u_1), (X_2, u_2)) \in \Delta_{q'}, \|f(X_2, u_2) - f(X_1, u_1)\| \leq L_{\Delta_{q'}} [\|X_2 - X_1\| + \|u_2 - u_1\|].$$

Dans la suite de la démonstration, on utilise la notation L pour désigner L_a (resp. $L_{\Delta_{q'}}$) si f vérifie l'hypothèse a (resp. l'hypothèse c).

Calcul de l'erreur d'interpolation au niveau d'une cellule du maillage Δ

Soit q' un indice du maillage Δ . On note :

$$\sigma_1 = (X_1, u_1), \dots, \sigma_{n+m+1} = (X_{n+m+1}, u_{n+m+1})$$

les sommets de la cellule $\Delta_{q'}$ associée. Soit (X, u) un point quelconque de $\Delta_{q'}$; (X, u) s'exprime donc comme combinaison convexe des sommets σ_i de $\Delta_{q'}$, i.e. :

$$\exists (\alpha_i)_{i=1 \dots n+m+1} \in [0, 1]^{n+m+1}, \sum_{i=1}^{n+m+1} \alpha_i = 1 \text{ et } (X, u) = \sum_{i=1}^{n+m+1} \alpha_i (X_i, u_i).$$

On choisit maintenant un indice $k_0 \in \{1, \dots, n+m+1\}$ tel que : $\alpha_{k_0} \geq \frac{1}{n+m+1}$. Évaluons la distance de $f(X, u)$ à $f_h(X, u)$:

$$\|f(X, u) - f_h(X, u)\| \leq \|f(X, u) - f_h(X_{k_0}, u_{k_0})\| + \|f_h(X_{k_0}, u_{k_0}) - f_h(X, u)\|.$$

Or, par construction, en tout noeud σ du maillage Δ : $f(\sigma) = f_h(\sigma)$. D'où :

$$\begin{aligned} \|f(X, u) - f_h(X_{k_0}, u_{k_0})\| &= \|f(X, u) - f(X_{k_0}, u_{k_0})\| \\ &\leq L (\|X - X_{k_0}\| + \|u - u_{k_0}\|) \\ &\leq 2L(1 - \alpha_{k_0})h. \end{aligned}$$

et

$$\begin{aligned} \|f_h(X_{k_0}, u_{k_0}) - f_h(X, u)\| &\leq \sum_{p=1}^{n+m+1} \alpha_p \|f_h(X_{k_0}, u_{k_0}) - f_h(X_p, u_p)\| \\ &\leq \sum_{p=1}^{n+m+1} \alpha_p \|f(X_{k_0}, u_{k_0}) - f(X_p, u_p)\| \\ &\leq L \sum_{p=1}^{n+m+1} \alpha_p (\|X_p - X_{k_0}\| + \|u_p - u_{k_0}\|) \leq 4L(1 - \alpha_{k_0})h \end{aligned}$$

Comme $4L(1 - \alpha_{k_0})h \leq \frac{4L(n+m)}{n+m+1}h$, on conclut :

$$\sup_{(X,u) \in \Delta_{q'}} \|f(X, u) - f_h(X, u)\| \leq \frac{4L(n+m)}{n+m+1} \quad (2.15)$$

Calcul de l'erreur globale d'interpolation

Si f vérifie l'hypothèse a, alors $L = L_a$ ne dépend pas du choix de $\Delta_{q'}$ et donc la relation (2.15) s'étend à l'espace tout entier :

$$\sup_{(X,u) \in \mathbb{R}^n \times \mathbb{U}_m} \|f(X, u) - f_h(X, u)\| \leq \frac{4L(n+m)}{n+m+1}h.$$

Supposons maintenant que f vérifie l'hypothèse c. Soit Ω un compact de \mathbb{R}^n ; on introduit l'ensemble $k(\Omega)$ des cellules du maillage Δ qui recouvrent $\Omega \times \mathbb{U}_m$, i.e. telles que :

$$k(\Omega) = \{q' \in I; \Delta_{q'} \cap (\Omega \times \mathbb{U}_m) \neq \emptyset\}.$$

On a donc également la relation : $\Omega \times \mathbb{U}_m \subset \bigcup_{q' \in k(\Omega)} \Delta_{q'}$, ce qui implique :

$$\sup_{(X,u) \in \Omega \times \mathbb{U}_m} \|f(X, u) - f_h(X, u)\| \leq \max_{q' \in k(\Omega)} \left(\sup_{(X,u) \in \Delta_{q'}} \|f(X, u) - f_h(X, u)\| \right).$$

On introduit alors la constante : $L_\Omega = \max_{q' \in k(\Omega)} L_{\Delta_{q'}}$. Le domaine Ω étant compact et donc borné, on a : $\text{card } k(\Omega) < +\infty$ et donc L_Ω est bien définie. On obtient ainsi la relation cherchée :

$$\sup_{(X,u) \in \Omega \times \mathbb{U}_m} \|f(X, u) - f_h(X, u)\| \leq \frac{4L_\Omega(n+m)}{n+m+1}h.$$

□

Dans la suite de ce manuscrit, $\varepsilon(h)$ désigne l'erreur d'interpolation globale commise en fonction du pas $h > 0$ du maillage de l'espace état-contrôle et $\varepsilon_\Omega(h)$ l'erreur d'interpolation commise sur un compact Ω de l'espace d'état.

2.3.3 Convergence de l'approximation

Soit $X_0 \in \mathbb{R}^n$ un point initial donné de l'espace d'état. Dans ce paragraphe, nous étudions la convergence des solutions du système affine par morceaux vers les solutions du système non linéaire initial pour un contrôle $u(\cdot)$ admissible donné.

On note $X(\cdot)$ la solution du système non linéaire : $\dot{X}(t) = f(X(t), u(t))$ pour la condition initiale $X(0) = X_0$ et $X_h(\cdot)$ la solution du système non linéaire : $\dot{X}_h(t) = f(X_h(t), u(t))$ pour la même condition initiale $X_h(0) = X_0$.

Solution approchée du système non linéaire

Dans notre contexte, la solution $X_h(\cdot)$ du système affine par morceaux n'a d'intérêt que si elle approche effectivement la solution du système non linéaire initial (2.1).

Comme cela existe déjà pour l'étude des équations différentielles ordinaires (voir [19, chap II, §1.3], [30] entre autres), nous sommes amenés à introduire la notion de solution approchée d'un système contrôlé, solution de régularité plus faible que dans le cadre classique :

Définition 2.3.1 (Solution ϵ -approchée). *Une fonction φ continue, dérivable par morceaux, est une solution ϵ -approchée du système : $\dot{X}(t) = f(X(t), u(t))$, $X(0) = X_0$, si la condition suivante est vérifiée :*

$$\forall t \geq 0, \text{ pour lequel } \dot{\varphi}(t) \text{ est défini, } \|\dot{\varphi}(t) - f(\varphi(t), u(t))\| \leq \epsilon. \quad (2.16)$$

Pour un contrôle et une condition initiale fixés, on démontre alors que les solutions du système affine par morceaux (2.2) approchent les solutions du système non linéaire (2.1) :

Lemme 2.3.2. *$X_h(\cdot)$ est une solution $\varepsilon(h)$ -approchée continue et dérivable par morceaux du système non linéaire : $\dot{X}(t) = f(X(t), u(t))$.*

Preuve. $X_h(\cdot)$ est la solution du problème de Cauchy associé au système affine par morceaux : $\dot{X}_h(t) = f_h(X_h(t), u(t))$ et à la condition initiale : $X_h(0) = X_0$. $X_h(\cdot)$ est donc une fonction continue, dérivable par morceaux. De plus, en tout point t où la dérivée existe, par définition de l'erreur d'interpolation $\varepsilon(h)$, on a :

$$\left\| \dot{X}_h(t) - f(X_h(t), u(t)) \right\| = \|f_h(X_h(t), u(t)) - f(X_h(t), u(t))\| \leq \varepsilon(h).$$

□

Nous avons ainsi vérifié que toute solution du système affine par morceaux est une solution approchée du système non linéaire initial. Il s'agit donc maintenant d'évaluer la convergence de cette approximation.

Convergence de l'approximation

Considérons une fois de plus l'équation différentielle ordinaire non autonome associée au système (2.1) : $\dot{X}(t) = F(t, X(t))$. Dans le cas où F est une fonction continue, le lemme fondamental [19, lemme 1.5.1], [30] nous permet de majorer la différence entre la solution $X(\cdot)$ et la solution approchée $X_h(\cdot)$, ce qui nous permet de conclure quant à la convergence de notre approximation (cf [44, th 11.2.1]).

Dans le contexte plus général des systèmes de contrôle, nous avons souligné le fait que la fonction F n'est a priori que mesurable et le lemme ne s'applique donc pas. Nous allons cependant montrer que, sous les différentes hypothèses évoquées au 2.3.1, nous obtenons des résultats similaires.

Sous les hypothèses a ou b, on peut montrer que le théorème de convergence C^1 de l'approximation des systèmes non contrôlés de [44, th 11.2.1] se généralise aux systèmes de contrôle non linéaires tels que (2.1) :

Théorème 2.3.2 (Convergence C^1 de l'approximation). *On suppose f globalement L -lipschitzienne par rapport à X , uniformément par rapport à u . Pour tout temps $t \geq 0$, pour lequel $X(t)$ et $X_h(t)$ sont définies,*

$$\|X(t) - X_h(t)\| \leq \frac{\varepsilon(h)}{L}(e^{Lt} - 1) \quad \text{et} \quad \left\| \dot{X}(t) - \dot{X}_h(t) \right\| \leq \varepsilon(h)e^{Lt}.$$

Pour démontrer le théorème 2.3.2, nous avons besoin du lemme auxiliaire suivant, énoncé et démontré dans [19, page 117] :

Lemme 2.3.3 (Lemme auxiliaire). *Soit $\varphi(\cdot)$ une fonction continue définie sur un intervalle de temps $[0, T]$, ($T > 0$), à valeurs positives, vérifiant l'inégalité :*

$$\forall t \in [0, T], \quad \varphi(t) \leq at + k \int_0^t \varphi(s) ds.$$

On a alors :

$$\forall t \in [0, T], \quad \varphi(t) \leq \frac{a}{k}(e^{kt} - 1).$$

Preuve du théorème 2.3.2. Soit $t \geq 0$. On cherche dans un premier temps à évaluer la différence entre la solution du système non linéaire et la solution approchée.

Par définition, $X(\cdot)$ (resp. $X_h(\cdot)$) est solution du problème de Cauchy $\dot{X}(t) = f(X(t), u(t))$ (resp. $\dot{X}_h(t) = f_h(X_h(t), u(t))$) pour la condition initiale $X(0) = X_0$ (resp. $X_h(0) = X_0$). On peut alors écrire les solutions sous forme intégrale (2.13) :

$$X(t) = X_0 + \int_0^t f(X(s), u(s)) ds, \quad X_h(t) = X_0 + \int_0^t f_h(X_h(s), u(s)) ds.$$

On en déduit ainsi :

$$\begin{aligned} \|X(t) - X_h(t)\| &= \left\| \int_0^t [f(X(s), u(s)) - f_h(X_h(s), u(s))] ds \right\| \\ &\leq \int_0^t \|f(X(s), u(s)) - f_h(X_h(s), u(s))\| ds. \end{aligned}$$

De plus, d'après l'inégalité triangulaire, pour tout $s \in [0, t]$, on a :

$$\begin{aligned} \|f(X(s), u(s)) - f_h(X_h(s), u(s))\| &\leq \|f(X(s), u(s)) - f(X_h(s), u(s))\| \\ &\quad + \|f(X_h(s), u(s)) - f_h(X_h(s), u(s))\|. \end{aligned}$$

Par hypothèse, f est L -lipschitzienne par rapport à X , d'où :

$$\|f(X(s), u(s)) - f(X_h(s), u(s))\| \leq L \|X(s) - X_h(s)\|,$$

et, par définition de l'erreur d'interpolation, on a également :

$$\|f(X_h(s), u(s)) - f_h(X_h(s), u(s))\| \leq \varepsilon(h).$$

En réinjectant ces inégalités dans le calcul de $\|X(t) - X_h(t)\|$, on obtient :

$$\|X(t) - X_h(t)\| \leq \int_0^t [L \|X(s) - X_h(s)\| + \varepsilon(h)] ds = \varepsilon(h)t + L \int_0^t \|X(s) - X_h(s)\| ds.$$

La fonction $s \mapsto \|X(s) - X_h(s)\|$ étant continue sur $[0, t]$, on lui applique le lemme auxiliaire 2.3.3, ce qui nous permet de conclure :

$$\|X(t) - X_h(t)\| \leq \frac{\varepsilon(h)}{L} (e^{Lt} - 1).$$

La seconde inégalité s'obtient alors simplement à partir des même inégalités intermédiaires :

$$\begin{aligned} \left\| \dot{X}(t) - \dot{X}_h(t) \right\| &= \|f(X(t), u(t)) - f_h(X_h(t), u(t))\| \\ &\leq \varepsilon(h) + L \|X(t) - X_h(t)\| \leq \varepsilon(h)e^{Lt}. \end{aligned}$$

□

D'autre part, sous l'hypothèse d, on obtient un résultat similaire au théorème 2.3.2 grâce à la compacité des ensembles atteignables à partir de X_0 en temps inférieur ou égal à t . Le résultat de compacité énoncé ci-après a été démontré et justifié en détails dans [1, chapitre 10, paragraphe 10.3]. Le soin est laissé au lecteur de s'y reporter pour davantage de précisions.

Théorème 2.3.3 (Compacité des ensembles atteignables). *Soit $\mathcal{A}(X_0, t)$ l'ensemble des points atteignables à partir du point X_0 en un temps inférieur ou égal à t du système $\dot{X} = f(X, u)$. On suppose :*

- i. Il existe un réel positif M tel que toute trajectoire issue de X_0 selon un contrôle admissible u et notée $X[X_0, u]$, est uniformément bornée par M sur $[0, t]$, i.e. :*

$$\exists M > 0, \forall u \in L^\infty([0, T], \mathbb{U}_m), \forall s \in [0, t], \|X[X_0, u](s)\| \leq M.$$

- ii. Pour tout $X \in \mathbb{R}^n$, l'ensemble $\{f(X, u) ; u \in \mathbb{U}_m\}$ est convexe.*

Alors l'ensemble $\mathcal{A}(X_0, t)$ est compact quels que soient $X_0 \in \mathbb{R}^n$ et $t > 0$.

Par une démonstration tout à fait similaire à celle du théorème 2.3.2, on prouve la convergence C^1 de l'approximation pour des fonctions f satisfaisant l'hypothèse d :

Proposition 2.3.2. *On suppose f lipschitzienne sur tout compact par rapport à X , uniformément par rapport à u (hypothèse d). On suppose également que l'ensemble atteignable $\mathcal{A}(X_0, s)$ est compact pour tout $s > 0$.*

Alors pour tout temps $t > 0$,

$$\|X(t) - X_h(t)\| \leq \frac{\varepsilon_\Omega(h)}{L_\Omega}(e^{L_\Omega t} - 1) \quad \text{et} \quad \left\| \dot{X}(t) - \dot{X}_h(t) \right\| \leq \varepsilon_\Omega(h)e^{L_\Omega t},$$

où Ω est le compact $\mathcal{A}(X_0, t) \cup \mathcal{A}_h(X_0, t)$.

Dans ce chapitre, nous avons développé une méthode d'approximation des dynamiques contrôlées non linéaires par des systèmes affines par morceaux. Nous avons également fourni une étude complète de la convergence de cette approximation selon différents jeux d'hypothèses qui garantissent l'existence et l'unicité de solutions du système non linéaire. Il apparaît alors que le point clé de cette étude est la définition du maillage de l'espace état-contrôle et de sa taille h . Dans le prochain chapitre, nous allons voir qu'il n'est pas obligatoire de mailler tout l'espace état-contrôle mais qu'au contraire, on peut le calculer à la volée.

Chapitre 3

Construction d'un modèle hybride de contrôle

Dans ce chapitre, nous nous intéressons à la modélisation des systèmes de contrôle non linéaires par les systèmes hybrides. Étant donné un système non linéaire général :

$$\dot{X}(t) = f(X(t), u(t)), \quad (3.1)$$

nous voulons construire un modèle suffisamment élaboré afin d'approcher la dynamique (3.1) avec une bonne précision mais aussi suffisamment simple pour en permettre l'analyse mathématique.

La notion de calcul hybride est une notion récente apparue en 2001 dans [29]. L'idée du calcul hybride est d'approcher des dynamiques non linéaires par des modèles hybrides. Une classe de modèles très utilisée est la classe des systèmes dynamiques affines par morceaux introduite par E.D. Sontag dans [75]. Ces systèmes sont en effet devenus un outil très pertinent et performant dans l'étude des systèmes non linéaires : ils parviennent généralement à reproduire les caractéristiques intrinsèques des systèmes physiques, tout en permettant une analyse mathématique relativement souple [52].

Un aperçu des différents types de problèmes pouvant être traités par le calcul hybride est présenté dans [65]. Citons en particulier les travaux d'A. Girard qui appliquent les idées du calcul hybride à la résolution d'équations différentielles ordinaires autonomes : $\dot{X}(t) = F(X(t))$ sur un domaine compact de l'espace d'état [43, 44].

Le principe général est le suivant : on construit un maillage de l'espace état-contrôle sur lequel on calcule une approximation affine par morceaux du champ de vecteurs non linéaire f en utilisant les résultats du chapitre 2. Le maillage, associé à la dynamique affine par morceaux, nous permet de définir un automate hybride contrôlé modélisant le système initial.

Contre toute attente, la classe habituelle des systèmes hybrides affines par morceaux (dont la dynamique dans un mode q donné est affine) se révèle inadaptée pour l'approximation de dynamiques contrôlées non linéaires. Nous proposons en conséquence une nouvelle classe de systèmes affines par morceaux plus appropriée compte tenu du problème qui nous intéresse.

Ce chapitre s'articule autour de la construction algorithmique d'une nouvelle classe de systèmes hybrides polyédraux affines par morceaux. Ces systèmes introduits au chapitre 1 sont définis en deux temps de la façon suivante :

1. On construit un maillage $\mathcal{D} = (D_q)_{q \in \mathcal{Q}}$ de l'espace d'état ce qui nous permet alors d'introduire la dynamique discrète $(\mathcal{Q}, \mathcal{E})$ de l'automate associé : l'ensemble \mathcal{Q} des états discrets du système correspond à l'ensemble dénombrable des indices du maillage \mathcal{D} . Une transition $e = (q, q') \in \mathcal{E}$ entre deux états q et q' est possible si et seulement si les cellules associées D_q et $D_{q'}$ ont une frontière en commun. Cette frontière est appelée garde du système et est notée G_e .
2. On introduit ensuite l'approximation du champ non linéaire f définie sur le maillage \mathcal{D} , décrivant ainsi la dynamique continue $(\mathcal{D}, \mathcal{U}, \mathcal{F})$ de l'automate.

Le couplage des deux composantes - discrète et continue - se fait par l'intermédiaire des gardes du système : une transition e entre deux modes q et q' est possible à l'instant t si la variable continue $X(t)$ appartient à la garde G_e .

Dans la partie 3.1, nous proposons donc une méthode explicite de construction d'un maillage semi-implicite de l'espace état-contrôle. Ensuite, en utilisant les outils développés au chapitre 2, nous définissons dans la partie 3.2 un automate hybride modélisant le système de contrôle non linéaire (3.1). Nous proposons en particulier des algorithmes permettant le calcul à la volée de l'état discret et des contraintes du modèle à un instant donné.

3.1 Maillage implicite de l'espace état-contrôle

Soit $h > 0$ le pas de discrétisation. Dans cette partie, nous voulons construire un maillage implicite de taille h de l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$. Il existe une infinité de façons de construire un tel maillage. Cependant le choix du maillage est un élément déterminant dans l'approximation des dynamiques non linéaires. En effet, à un maillage donné correspond une unique approximation affine par morceaux (2.2) calculée par interpolation du champ f . Il est donc important de choisir un maillage adapté à la configuration dynamique du système que l'on modélise.

3.1.1 Caractérisation du maillage implicite

A partir de maintenant, on note $\Delta = (\Delta_i)_{i \in I}$ le maillage en simplexes de l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$. I désigne donc l'ensemble dénombrable des indices des cellules de Δ .

Le premier critère de choix du maillage Δ est un critère dynamique. En effet, le problème de contrôle non linéaire défini au chapitre 1 consiste à prouver qu'il existe des contrôles admissibles $u(\cdot)$ permettant d'atteindre une cible donnée (ici $X_f = 0$). Une fois la cible atteinte, le système ne doit plus évoluer ce qui se traduit par l'existence d'un point fixe du système non linéaire en $(0, 0)$.

Par conséquent, parmi l'ensemble des maillages possibles de l'espace état-contrôle, nous sélectionnons ceux pour lesquels le système affine par morceaux associé admet un point fixe en $(0, 0)$ (i.e. pour lesquels : $f_h(0, 0) = 0$). Une façon simple de garantir l'existence de ce point fixe est d'imposer la propriété suivante au maillage cherché :

Propriété 3.1.1. *Si $0 \in \Delta_i$, alors 0 est un sommet de Δ_i .*

Lemme 3.1.1. *Soit Δ un maillage de $\mathbb{R}^n \times \mathbb{U}_m$ vérifiant la propriété 3.1.1. Alors $(0, 0)$ est un point d'équilibre du système : $\dot{X}(t) = f_h(X(t), u(t))$.*

Preuve. En tout sommet σ du maillage Δ , les contraintes d'interpolation du paragraphe 2.1.1 s'écrivent : $f(\sigma) = f_h(\sigma)$. De plus, $(0, 0)$ est par définition un point fixe du système non linéaire (3.1). On en déduit donc : $f_h(0, 0) = f(0, 0) = 0$.

□

Le second critère de choix de Δ est une contrainte géométrique sur la position des cellules du maillage dans $\mathbb{R}^n \times \mathbb{U}_m$. On impose arbitrairement au maillage Δ de se projeter sur un maillage en simplexes, noté $\mathcal{D} = (D_i)_{i \in \mathcal{Q}}$, de l'espace d'état \mathbb{R}^n :

Propriété 3.1.2. *Soit $\mathcal{D} = (D_i)_{i \in \mathcal{Q}}$ la projection orthogonale de $\Delta = (\Delta_i)_{i \in I}$ sur \mathbb{R}^n .*

1. \mathcal{D} est un maillage simplicial de \mathbb{R}^n

2. Soit $(i, j) \in I^2$. On pose : $D_i = p_{\mathbb{R}^n}^\perp(\Delta_i)$ et $D_j = p_{\mathbb{R}^n}^\perp(\Delta_j)$. On a alors :

$$D_i = D_j \text{ ou } \overset{\circ}{D}_i \cap \overset{\circ}{D}_j = \emptyset$$

Cette propriété nous permet d'introduire la relation d'équivalence \sim définie sur l'ensemble I des indices du maillage Δ :

$$i \sim j \Leftrightarrow p_{\mathbb{R}^n}^\perp(\Delta_i) = p_{\mathbb{R}^n}^\perp(\Delta_j).$$

D'après la propriété 3.1.2, deux simplexes Δ_i et Δ_j sont en relation, si et seulement s'ils se projettent exactement sur un même simplexe D_q du maillage \mathcal{D} de \mathbb{R}^n . Les classes d'équivalence de la relation \sim sont donc définies de la façon suivante :

Lemme 3.1.2. *Les classes d'équivalence pour la relation \sim sont les ensembles $\mathcal{K}(q)$ définis comme suit :*

$$\mathcal{K}(q) = \{i \in I ; p_{\mathbb{R}^n}^\perp(\Delta_i) = D_q\}$$

où q est un indice du maillage \mathcal{D} de \mathbb{R}^n . De plus : $\forall q, \text{card } \mathcal{K}(q) < +\infty$.

Ainsi, à chaque cellule D_i du maillage de \mathbb{R}^n correspond une "colonne" de cellules de Δ qui se projettent exactement sur D_i , cf figure 3.1-(a).

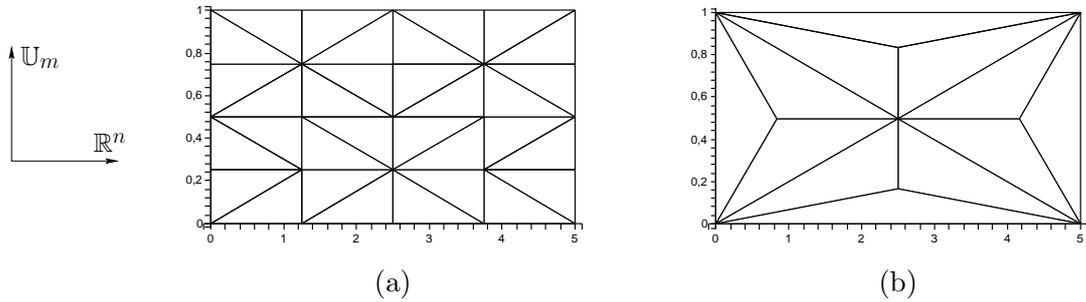


FIG. 3.1 – Exemples de maillages, pour $n=m=1$, vérifiant la propriété 1. La figure (a) vérifie également la propriété 2. Ce n'est pas le cas de (b).

D'après la structure du maillage Δ , le domaine $D_q \times \mathbb{U}_m$ se décompose en $\text{card } \mathcal{K}(q)$ simplexes selon la formule :

Lemme 3.1.3. $D_q \times \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} \Delta_{q'}$.

Cette propriété nous permettra dans la suite de définir les mêmes conditions de sorties pour toutes les cellules Δ_i ayant même projeté D_j sur \mathbb{R}^n . De ce fait, les conditions de sortie ne dépendent plus que de la position dans l'espace d'état. Nous reviendrons plus en détails sur ces notions dans le paragraphe 3.2.

Nous sommes maintenant en mesure de définir un algorithme de maillage de l'espace état-contrôle. Cependant, dans le cas de la résolution d'un problème de contrôle, la triangulation i.e. le découpage en simplexes, du domaine de contrôle ne peut se faire à la volée. En effet, la différence essentielle avec la résolution à la volée des systèmes différentiels ordinaires réside dans le fait qu'en théorie du contrôle, nous ne disposons généralement pas de condition initiale sur le contrôle. De ce fait, pour une position initiale $X_0 \in \mathbb{R}^n$ donnée dans l'espace d'état, nous calculons la cellule du maillage de l'espace d'état qui contient X_0 . Nous sommes ensuite obligés de calculer toutes les cellules de contrôle possibles ce qui revient à calculer une fois pour toutes une triangulation complète du domaine de contrôle.

3.1.2 Triangulation de $\mathbb{R}^n \times \mathbb{U}_m$ et taille du maillage

Le maillage implicite de l'espace état-contrôle est défini comme suit : on commence par calculer une fois pour toutes une triangulation explicite du domaine de contrôle \mathbb{U}_m . On construit ensuite un maillage implicite $\mathcal{D} = (D_q)_{q \in \mathcal{Q}}$ de l'espace d'état, ce qui nous permet d'en déduire un maillage semi-implicite $(\Delta_i)_{i \in I}$ de l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$.

Triangulation du domaine de contrôle

Le domaine de contrôle \mathbb{U}_m est un polyèdre convexe borné, défini comme l'enveloppe convexe de ses sommets. La subdivision simpliciale de tels polyèdres a été très largement étudiée ces dernières années et nous disposons aujourd'hui d'outils performants pour les calculer (par exemple, triangulation de Delaunay du logiciel *Qhull*¹[6]).

Pour pouvoir construire un maillage de l'espace état-contrôle, nous avons besoin de calculer un maillage régulier (autant que possible) de taille h du polytope \mathbb{U}_m . Pour l'instant, nous utilisons un algorithme combinant une discrétisation du type Sierpinski avec une triangulation de Delaunay : l'idée est de rajouter de points répartis régulièrement dans le polytope et d'appliquer une triangulation de Delaunay grâce au logiciel *Qhull*.

Maillage implicite de l'espace d'état

Pour cette étape, nous reprenons l'algorithme décrit dans [44, §11.1.2]. Le principe général est le suivant : l'espace \mathbb{R}^n est implicitement découpé en n -cubes d'arêtes de longueur h . Chaque cube est ensuite maillé en $n!$ simplexes et la partition résultante est un maillage de \mathbb{R}^n . Dans ce paragraphe, nous complétons cette construction par le calcul de la taille du maillage obtenu et par une démonstration complète de l'algorithme présenté.

¹Outil pour le calcul d'enveloppes convexes, de triangulation de Delaunay ou encore de diagrammes de Voronoï en dimension 2, 3 ou plus.

Tout cube C en dimension n est défini par la donnée d'un point origine $a = (a_1, \dots, a_n)$ de \mathbb{R}^n et de la longueur h de ses arêtes :

$$C = [a_1, a_1 + h] \times \dots \times [a_n, a_n + h]$$

On note : $C = a + [0, h]^n$. De la même façon, nous définissons le maillage de \mathbb{R}^n en n -cubes :

Définition 3.1.1 (Maillage de \mathbb{R}^n en n -cubes). Soit $a = (a_1, \dots, a_n)$ un point de \mathbb{R}^n et $h > 0$. Alors $(a + kh + [0, h]^n)_{k=(k_1, \dots, k_n) \in \mathbb{Z}^n}$ est un maillage de \mathbb{R}^n en n -cubes.

D'après la propriété 3.1.1, le point $(0, 0)$ doit être un sommet du maillage cherché, ce qui revient à choisir $a = 0$ comme origine de ce maillage. Maintenant que nous savons découper l'espace d'état en n -cubes de côté h , la difficulté est de construire une triangulation implicite de chacun de ces cubes.

Considérons un n -cube $C = b + [0, h]^n$ quelconque de \mathbb{R}^n (ici $b \in h\mathbb{Z}^n$) et \mathcal{S}_n l'ensemble des permutations de $\{1, \dots, n\}$. Pour toute permutation $\varphi \in \mathcal{S}_n$,

$$D_{b,\varphi} = \{(x_1, \dots, x_n) \in \mathbb{R}^n; 0 \leq x_{\varphi(1)} - b_{\varphi(1)} \leq \dots \leq x_{\varphi(n)} - b_{\varphi(n)} \leq h\} \quad (3.2)$$

est un simplexe de \mathbb{R}^n dont les sommets, notés $\sigma_{\varphi,p}$, sont définis par :

$$\begin{cases} \forall i = 1, \dots, p, & x_{\varphi(i)} = b_{\varphi(i)} \\ \forall i = p + 1, \dots, n, & x_{\varphi(i)} = b_{\varphi(i)} + h \end{cases}, p = 0, \dots, n \quad (3.3)$$

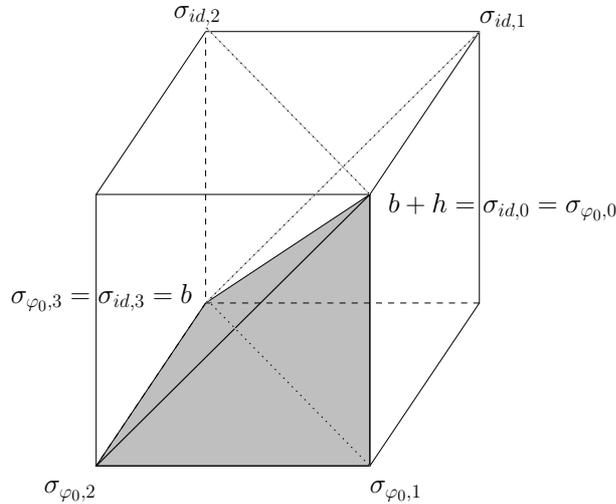


FIG. 3.2 – Découpage du cube $b + [0, h]^3$ en simplexes : sur la figure ci-dessus, on représente les simplexes $D_{b,\varphi} = Conv(\sigma_{\varphi,0}, \sigma_{\varphi,1}, \sigma_{\varphi,2}, \sigma_{\varphi,3})$, pour $\varphi = id$ et $\varphi = \varphi_0 = (3, 2, 1)$.

On a alors les résultats suivants :

Proposition 3.1.1 ([40]). $(D_{b,\varphi})_{\varphi \in \mathcal{S}_n}$ est un maillage du n -cube $C = b + [0, h]^n$.

Corollaire 3.1.1. $(D_{b,\varphi})_{\varphi \in \mathcal{S}_n}$ est un maillage de taille \sqrt{nh} .

Preuve. On considère un simplexe $D_{b,\varphi}$, $\varphi \in \mathcal{S}_n$, du maillage de C . On rappelle que la taille de $D_{b,\varphi}$ est égale à : $\sup_{(x,y) \in D_{b,\varphi}} \|x - y\|_2$ où $\|\cdot\|_2$ désigne la norme euclidienne définie sur \mathbb{R}^n .

Soit $(x, y) \in D_{b,\varphi}$. Calculons la distance euclidienne $\|x - y\|_2$ de x à y :

$$\|x - y\|_2 = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} = \sqrt{\sum_{i=1}^n (x_{\varphi(i)} - y_{\varphi(i)})^2}$$

De plus, par définition de $D_{b,\varphi}$, pour tout $i \in \{1, \dots, n\}$, on a : $0 \leq x_{\varphi(i)} - b_{\varphi(i)} \leq h$ et $0 \leq y_{\varphi(i)} - b_{\varphi(i)} \leq h$ d'où :

$$\forall i \in \{1, \dots, n\}, |x_{\varphi(i)} - y_{\varphi(i)}| \leq h, \text{ ce qui implique : } \|x - y\|_2 \leq \sqrt{\sum_{i=1}^n h^2} = \sqrt{n}h.$$

La taille du simplexe $D_{b,\varphi}$ est donc au moins égale à $\sqrt{n}h$. De plus, d'après les formules (3.3), on vérifie que les points b et $b + h$ appartiennent à $D_{b,\varphi}$ quelle que soit la permutation φ choisie :

$$\forall \varphi \in \mathcal{S}_n, b \in D_{b,\varphi} \text{ et } b + h \in D_{b,\varphi} \quad (3.4)$$

La distance de b à $b + h$ étant exactement égale à $\sqrt{n}h$, on en déduit que $D_{b,\varphi}$ est de taille $\sqrt{n}h$ pour toute permutation φ de \mathcal{S}_n . Le maillage $(D_{b,\varphi})_{\varphi \in \mathcal{S}_n}$ du n -cube C est donc exactement de taille $\sqrt{n}h$. □

En résumé, l'espace d'état est découpé à partir de l'origine 0 en n -cubes de côté h . Ensuite, chaque n -cube est à son tour subdivisé en $n!$ simplexes. Le point clé de cette construction est donc de vérifier que l'ensemble des simplexes qui maillent les n -cubes de la partition de \mathbb{R}^n forme un maillage de l'espace \mathbb{R}^n tout entier. Par construction, la réunion de tous ces simplexes est bien égale à \mathbb{R}^n ; il s'agit donc de vérifier que les simplexes de cubes adjacents coïncident sur la face d'intersection des deux cubes (cf figure 3.3).

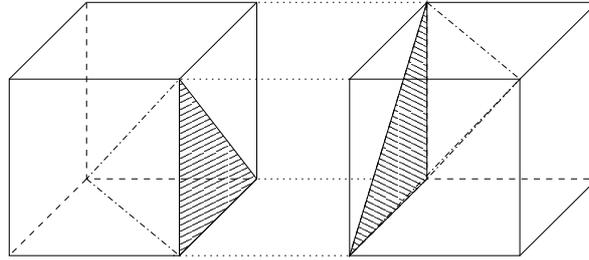


FIG. 3.3 – Exemple de deux simplexes qui ne se recollent pas à l'intersection de deux cubes adjacents

On note $\mathcal{C} = (C_k)_{k \in \mathbb{Z}^n}$ le maillage de \mathbb{R}^n en n -cubes, d'origine 0. Dans un premier temps, on montre que les simplexes $(D_{k,\varphi})_{k \in \mathbb{Z}^n}$ associés à une permutation $\varphi \in \mathcal{S}_n$ donnée, se déduisent les uns des autres par translation :

Lemme 3.1.4 (lemme auxiliaire). Soient $C_k = kh + [0, h]^n$ et $C_l = lh + [0, h]^n$ deux cubes distincts quelconques du maillage \mathcal{C} . Alors, pour toute permutation $\varphi \in \mathcal{S}_n$, le simplexe $D_{l,\varphi}$ est l'image du simplexe $D_{k,\varphi}$ par une translation de vecteur $(l - k)h$:

$$\forall \varphi \in \mathcal{S}_n, D_{l,\varphi} = D_{k,\varphi} + (l - k)h$$

Preuve. Soit $\varphi \in \mathcal{S}_n$ une permutation de l'ensemble $\{1, \dots, n\}$. On considère la translation $t_{\vec{v}}$ de vecteur $\vec{v} = (l - k)h$. L'image d'un simplexe par une translation est un simplexe, dont les sommets sont les translatés des sommets de départ. Pour démontrer le lemme, il suffit donc de prouver que l'image d'un sommet de $D_{k,\varphi}$ par $t_{\vec{v}}$ est un sommet de $D_{l,\varphi}$.

Soit $X = (x_1, \dots, x_n)$ un sommet de $D_{k,\varphi}$. D'après la définition des sommets donnée par (3.3), il existe $p \in \{0, \dots, n\}$, tel que les coordonnées de X vérifient :

$$\begin{cases} \forall i = 1, \dots, p, & x_{\varphi(i)} = k_{\varphi(i)}h \\ \forall i = p + 1, \dots, n, & x_{\varphi(i)} = (k_{\varphi(i)} + 1)h \end{cases} .$$

On définit alors $Y \in \mathbb{R}^n$ le translaté de X par $t_{\vec{v}}$. On a donc :

$$Y = t_{\vec{v}}(X) = X + (l - k)h = (x_1 + (l_1 - k_1)h, \dots, x_n + (l_n - k_n)h)$$

d'où, pour tout $i \in \{1, \dots, n\}$, $y_{\varphi(i)} = x_{\varphi(i)} + (l_{\varphi(i)} - k_{\varphi(i)})h$, ce qui nous donne :

$$\begin{cases} \forall i = 1, \dots, p, & y_{\varphi(i)} = k_{\varphi(i)}h + (l_{\varphi(i)} - k_{\varphi(i)})h = l_{\varphi(i)}h \\ \forall i = p + 1, \dots, n, & y_{\varphi(i)} = (k_{\varphi(i)} + 1)h + (l_{\varphi(i)} - k_{\varphi(i)})h = (l_{\varphi(i)} + 1)h \end{cases} .$$

On en déduit alors : $Y \in D_{l,\varphi}$.

□

Grâce au lemme auxiliaire ci-dessus, il nous suffit alors de démontrer que les maillages induits sur deux faces parallèles d'un même cube se superposent, ce qui nous permet de démontrer le résultat suivant :

Proposition 3.1.2. $(D_{k,\varphi})_{k \in \mathbb{Z}^n, \varphi \in \mathcal{S}_n}$ est un maillage simplicial de \mathbb{R}^n .

Preuve. D'après la définition 3.1.1, l'espace \mathbb{R}^n est découpé en n -cubes C_k , $k \in \mathbb{Z}^n$, eux-mêmes subdivisés en simplexes $D_{k,\varphi}$, $\varphi \in \mathcal{S}_n$, ce qui nous donne donc :

$$\mathbb{R}^n = \bigcup_{k \in \mathbb{Z}^n} C_k = \bigcup_{k \in \mathbb{Z}^n} \bigcup_{\varphi \in \mathcal{S}_n} D_{k,\varphi}$$

D'après la définition A.2.1 d'un maillage, il reste donc à démontrer que l'intersection de deux simplexes $D_{k,\varphi}$ et $D_{l,\psi}$ est soit vide, soit égale à l'enveloppe convexe de leurs sommets communs.

La démonstration que nous proposons s'appuie sur deux remarques : premièrement, si les simplexes $D_{k,\varphi}$ et $D_{l,\psi}$ sont dans des cubes non adjacents, leur intersection est nécessairement vide. Nous nous ramenons donc à des simplexes contenus dans deux cubes adjacents et

le problème illustré sur la figure 3.3 est de vérifier que les maillages respectifs de ces deux cubes se recollent sur la face commune aux deux cubes.

De plus, d'après le lemme 3.1.4, pour une permutation φ donnée, les simplexes associés à φ dans deux cubes adjacents se déduisent l'un de l'autre par translation. Il suffit donc de démontrer que les triangulations des faces parallèles du cube se superposent.

On considère le cube unité $C = [0, h]^n$ et une permutation φ de l'ensemble $\{1, \dots, n\}$. Le simplexe D_φ associé à φ est donc défini comme l'enveloppe convexe de ses sommets σ_p , $p = 0, \dots, n$, donnés par la formule (3.3) :

$$\sigma_p : \begin{cases} x_{p,\varphi(i)} = 0, & i = 1, \dots, p, \\ x_{p,\varphi(i)} = h, & i = p+1, \dots, n, \end{cases}, \quad p = 0, \dots, n$$

Avec ces notations, on a donc : $\sigma_0 = (h, \dots, h)$ et $\sigma_n = (0, \dots, 0)$.

On considère la face $F = \text{Conv}(\sigma_1, \dots, \sigma_n)$ du simplexe D_φ (i.e. : $\sigma_0 = (h, \dots, h) \notin F$). Par définition des sommets de F , nous avons donc la relation :

$$\forall p = 1, \dots, n, \quad x_{p,\varphi(1)} = 0, \quad (3.5)$$

ce qui signifie que la face F est incluse dans l'hyperplan d'équation $(x_{\varphi(1)} = 0)$.

On applique alors la translation de vecteur $h \vec{e}_{\varphi(1)}$ à la face F : c'est la translation qui permet de passer de l'hyperplan $(x_{\varphi(1)} = 0)$ à l'hyperplan parallèle $(x_{\varphi(1)} = h)$; on définit alors les points translatés $\gamma_p = (y_{p,1}, \dots, y_{p,n})$ des sommets σ_p , pour $p = 1, \dots, n$:

$$\forall p = 1, \dots, n, \quad \forall j = 1, \dots, n, \quad y_{p,j} = x_{p,j} + h\delta_{\varphi(1)j}, \quad (3.6)$$

où δ_{ij} est le symbole de Kronecker, égal à 1 si $i = j$, à 0 sinon.

On obtient ainsi un $(n-1)$ -simplexe image contenu dans l'hyperplan d'équation : $(x_{\varphi(1)} = h)$. Le problème est maintenant de vérifier que ce simplexe est une face d'un simplexe du maillage du cube. D'après (3.4), le point $(0, \dots, 0)$ appartient nécessairement à tout simplexe de la subdivision du cube C . On pose en conséquence :

$$D = \text{Conv}(\sigma_n, \gamma_1, \dots, \gamma_n)$$

Nous voulons maintenant démontrer que D est un simplexe et qu'il appartient au maillage du cube C , i.e. qu'il existe une permutation $\psi \in \mathcal{S}_n$, telle que : $D = D_\psi$.

On pose : $\omega_0 = \gamma_1$, $\omega_1 = \gamma_2$, \dots , $\omega_{n-1} = \gamma_n$, $\omega_n = (0, \dots, 0)$. Nous cherchons donc une permutation $\psi \in \mathcal{S}_n$, telle que les points ω_p vérifient une relation du type (3.3), i.e. :

$$\forall p = 1, \dots, n, \quad \begin{cases} y_{p,\psi(j)} = 0, & j = 1, \dots, p-1 \\ y_{p,\psi(j)} = h, & j = p, \dots, n \end{cases}$$

On introduit l'application ψ définie de la façon suivante :

$$\psi(j) = \varphi(j+1), \quad j = 1, \dots, n-1$$

$$\psi(n) = \varphi(1)$$

Puisque φ est une permutation de \mathcal{S}_n , ψ l'est aussi. D'après la relation (3.6), on sait que :

$$\forall p, j = 1, \dots, n, \quad y_{p,\psi(j)} = x_{p,\psi(j)} + h\delta_{\varphi(1)\psi(j)},$$

d'où, par définition de ψ , pour tout $p = 1, \dots, n$:

$$\begin{aligned} \forall j = 1, \dots, n-1, \quad y_{p,\psi(j)} &= x_{p,\varphi(j+1)} + h\delta_{\varphi(1)\varphi(j+1)} = x_{p,\varphi(j+1)} \\ y_{p,\psi(n)} &= x_{p,\varphi(1)} + h\delta_{\varphi(1)\varphi(1)} = x_{p,\varphi(1)} + h = h \end{aligned}$$

puisque $x_{p,\varphi(1)} = 0$ d'après (3.5). On peut alors en déduire les relations cherchées :

$$\begin{cases} y_{p,\psi(j)} = x_{p,\varphi(j+1)} = 0, & j = 1, \dots, p-1 \\ y_{p,\psi(j)} = x_{p,\varphi(j+1)} = h, & j = p, \dots, n-1 \\ y_{p,\psi(n)} = x_{p,\varphi(1)} + h = h \end{cases},$$

ce qui nous permet de conclure : $D = D_\psi$.

□

Ainsi, nous avons démontré par recollement que le découpage de \mathbb{R}^n en n -cubes puis en simplexes, est un maillage de \mathbb{R}^n . Ce dernier résultat nous permet d'affirmer que le maillage de l'espace d'état ainsi construit est de taille \sqrt{nh} .

Maillage de $\mathbb{R}^n \times \mathbb{U}_m$

Soient $(D_i)_{i \in J}$ et $(U_j)_j$ maillages respectifs de \mathbb{R}^n et \mathbb{U}_m . On construit un maillage Δ de $\mathbb{R}^n \times \mathbb{U}_m$ via la triangulation de Delaunay des domaines $D_i \times (U_j)_j$, $i \in J$ sans ajouter de sommets (en utilisant *Qhull*). De ce fait, le point $(0, 0) \in \mathbb{R}^n \times \mathbb{U}_m$ est bien l'un des noeuds du maillage Δ et la propriété 3.1.2 est satisfaite.

Au chapitre 2, nous avons insisté sur le rôle de la taille du maillage considéré pour la convergence des solutions approchées du système linéaire. Il est donc important de calculer la taille du maillage semi-implicite de $\mathbb{R}^n \times \mathbb{U}_m$ obtenu :

Lemme 3.1.5. *Soient D_i et U_j deux polytopes respectivement de \mathbb{R}^n et \mathbb{R}^m . On a alors :*

$$\text{diam}(D_i \times U_j) = \sqrt{\text{diam}(D_i)^2 + \text{diam}(U_j)^2}$$

Preuve. Soient (x, u) et (y, v) deux points quelconques du domaine $D_i \times U_j$. Par définition de la distance euclidienne, on observe que :

$$\|(x, u) - (y, v)\|_2^2 = \|x - y\|_2^2 + \|u - v\|_2^2. \quad (3.7)$$

D'où la relation : $\|(x, u) - (y, v)\|_2^2 \leq \text{diam}(D_i)^2 + \text{diam}(U_j)^2$, i.e. :

$$\text{diam}(D_i \times U_j) \leq \sqrt{\text{diam}(D_i)^2 + \text{diam}(U_j)^2} \quad (3.8)$$

De plus d'après la proposition A.1.2, on sait qu'il existe deux sommets s_1 et s_2 (resp. u_1 et u_2) du polytope D_i (resp. U_j), tels que :

$$\text{diam}(D_i) = d(s_1, s_2) \quad \text{et} \quad \text{diam}(U_j) = d(u_1, u_2)$$

En appliquant la relation (3.7) à $(x, u) = (s_1, u_1)$ et $(y, v) = (s_2, u_2)$, on obtient :

$$d((s_1, u_1), (s_2, u_2))^2 = d(s_1, s_2)^2 + d(u_1, u_2)^2 = \text{diam}(D_i)^2 + \text{diam}(U_j)^2.$$

Or les points (s_1, u_1) et (s_2, u_2) sont des sommets du produit cartésien $D_i \times U_j$. On en déduit donc que le diamètre de $D_i \times U_j$ est au moins égal à $\text{diam}(D_i)^2 + \text{diam}(U_j)^2$. D'après (3.8), on a donc bien égalité. □

On déduit du lemme 3.1.5 que le maillage Δ obtenu est de taille $\sqrt{n+1}h$.

Dans cette première partie, nous avons décrit les principales étapes de la construction d'un maillage simplicial implicite de l'espace état-contrôle. Nous allons maintenant montrer que ce maillage, combiné à l'approximation affine par morceaux du chapitre 2, nous permet de définir une approximation hybride du système non linéaire initial.

3.2 Définition du modèle hybride

Au paragraphe précédent, nous avons défini un maillage semi-implicite Δ de l'espace état-contrôle. Grâce aux techniques d'approximation affine par morceaux développées au chapitre 2, on construit une approximation du système différentiel non linéaire (3.1) :

$$\dot{X}(t) = f_h(X(t), u(t))$$

où f_h est une approximation affine par morceaux de f définie sur le maillage Δ . Nous nous intéressons maintenant à la définition d'un automate hybride modélisant le système non linéaire initial.

L'état d'un système hybride est décrit par deux variables : l'une discrète notée $q(t)$ à valeurs dans un ensemble dénombrable \mathcal{Q} , la seconde continue notée $X(t)$ à valeurs dans l'espace des phases \mathbb{R}^n considéré. Dans chaque état $q \in \mathcal{Q}$, l'évolution de la variable $X(\cdot)$ est soumise à la dynamique f_q du mode considéré. De ce fait, du choix de l'ensemble d'état \mathcal{Q} choisi, dépend le type de dynamique considérée et donc le choix des méthodes de résolution appropriées.

3.2.1 Choix des états discrets et définition de l'automate hybride

Au chapitre 2, paragraphe 2.3.1, nous avons insisté sur le fait qu'en théorie du contrôle, nous sommes généralement amenés à considérer des fonctions de contrôle mesurables, non continues (au mieux continues par morceaux). Ainsi, toute trajectoire $(X(\cdot), u(\cdot))$ dans l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$ est a priori discontinue, alors que la trajectoire $X(\cdot)$ dans l'espace d'état est continue (cf figure 3.4).

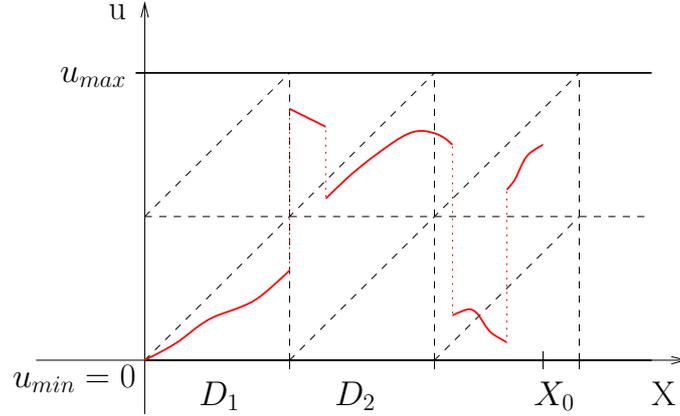


FIG. 3.4 – Exemple de trajectoire discontinue dans l'espace état-contrôle $\mathbb{R} \times [0, u_{max}]$

Par ailleurs, la figure 3.4 met également en évidence le fait que la notion de transition entre deux cellules du maillage Δ n'a aucun sens par rapport à la variable de contrôle. En effet, toute fonction de contrôle admissible peut passer à n'importe quel moment d'une cellule du maillage Δ à une autre, non nécessairement adjacente et surtout, que la trajectoire $(X(\cdot), u(\cdot))$ ait ou non atteint une garde du système.

Pour toutes ces raisons, le choix de la variable continue décrivant l'état du système hybride à définir est limité à la variable X à valeurs dans \mathbb{R}^n . En conséquence, l'ensemble des états discrets choisi est celui des indices du maillage simplicial de l'espace d'état \mathbb{R}^n .

On introduit alors les notations suivantes :

- \mathcal{Q} l'ensemble dénombrable des indices d'un maillage simplicial $(D_q)_q$ de taille h de l'espace d'état comme défini au paragraphe 3.1.2.
- $\mathcal{E} = \{(q, q') \in \mathcal{Q} ; \partial D_q \cap \partial D_{q'} \neq \emptyset\}$ l'ensemble des transitions : une transition est possible entre deux états q et q' si les cellules correspondantes D_q et $D_{q'}$ du maillage de \mathbb{R}^n sont adjacentes.
- $\mathcal{D} = \{D_q ; q \in \mathcal{Q}\}$ la collection des domaines du maillage de \mathbb{R}^n . Par construction, pour tout $q \in \mathcal{Q}$, D_q est un n -simplexe de \mathbb{R}^n et est donc d'intérieur non vide ($\overset{\circ}{D}_q \neq \emptyset$). Le domaine D_q définit un ensemble de contraintes affines sur l'état :

$$D_q = \{X \in \mathbb{R}^n ; N_q X + L_q \leq O\}$$

- $\mathcal{U} = \{U_m ; q \in \mathcal{Q}\}$ la collection des domaines de contrôles. Dans chaque état discret q , le contrôle est à valeurs dans le domaine initial U_m .
- $\mathcal{F} = \{f_q ; q \in \mathcal{Q}\}$ la collection de champs de vecteurs affines par morceaux, définis par interpolation du champ non linéaire f (cf chapitre 2) :

$$\forall q \in \mathcal{Q}, f_q : D_q \times U_m \rightarrow \mathbb{R}^n \\ (X, u) \mapsto f_h(X, u)$$

D'après le lemme 3.1.3, on sait que : $D_q \times \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} \Delta_{q'}$. Le champ f_q est donc défini

localement par :

$$\forall q' \in \mathcal{K}(q), \forall (X, u) \in \Delta_{q'}, f_q(X, u) = A_{q'}X + B_{q'}u + c_{q'}$$

– $\mathcal{G} = \{G_e ; e \in \mathcal{E}\}$ l'ensemble des gardes :

$$\forall e = (q, q') \in \mathcal{E}, G_e = \partial D_q \cap \partial D_{q'} \subset D_q$$

– $\mathcal{R} = \{R_e ; e \in \mathcal{E}\}$ l'ensemble des fonctions Reset. On pose : $\forall e \in \mathcal{E}, R_e(X) = \{X\}$, ce qui traduit le fait que la variable continue $X(\cdot)$ n'est pas ré-initialisée au moment d'une transition e du système.

D'après la définition générale des systèmes hybrides donnée en introduction (cf paragraphe 1.3.1), nous avons ainsi construit un système hybride approchant la dynamique non linéaire initiale :

Proposition 3.2.1. *Le septuple $\mathcal{H} = (\mathcal{Q}, \mathcal{E}, \mathcal{D}, \mathcal{U}, \mathcal{F}, \mathcal{G}, \mathcal{R})$ défini ci-dessus est un système hybride.*

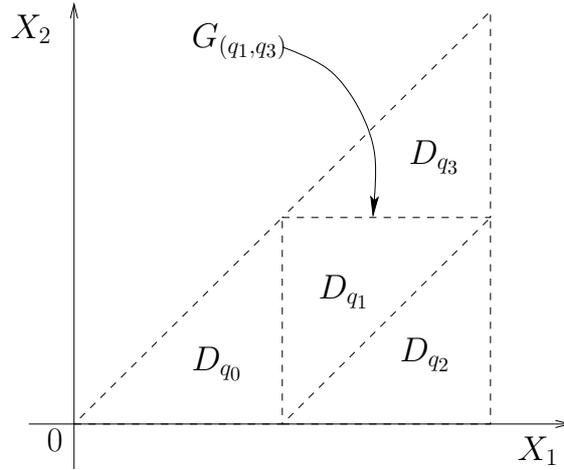


FIG. 3.5 – Illustration de la définition d'un automate hybride

Par définition de l'automate \mathcal{H} , l'évolution de la variable continue $X(\cdot)$ détermine à elle seule celle de la variable discrète $q(\cdot)$ au cours du temps. En effet, tout état discret $q \in \mathcal{Q}$ correspond à une unique cellule D_q de l'espace d'état et est donc lié à la position du système dans l'espace d'état. Si l'on considère une trajectoire $X(\cdot)$ dans \mathbb{R}^n , celle-ci traverse nécessairement un certain nombre de cellules D_q du maillage \mathcal{D} de \mathbb{R}^n , définissant ainsi la suite discrète des valeurs prise par la variable $q(\cdot)$ au cours du temps.

À tout instant t , l'état du système hybride \mathcal{H} est décrit par sa position $X(t)$ dans l'espace d'état et le mode $q(t) = q$ associé. À l'instant $t = 0$, le système se trouve à la position X_0 . On en déduit le mode q_0 dans lequel se trouve l'automate \mathcal{H} à l'instant $t = 0$. Dans le mode

q_0 , la variable continue $X(\cdot)$ est soumise à la dynamique : $\dot{X}(t) = f_{q_0}(X(t), u(t))$ définie sur le domaine $D_{q_0} \times \mathbb{U}_m$. Dès que le système atteint une garde $G_{(q_0, q_1)}$ à un instant t_1 , il peut effectuer une transition discrète de l'état q_0 vers l'état q_1 . On applique alors à nouveau le même processus à partir de la position $X(t_1)$ dans l'état q_1 .

Remarque 3.2.1. *Dans la suite de ce manuscrit, nous supposons que le modèle hybride ainsi construit n'accepte pas de trajectoire solution dite Zénon qui effectue une infinité de transitions en un temps fini. En effet, l'existence de trajectoires Zénon pose parfois des problèmes de simulation puisque les instants de transition sont de plus en plus proches jusqu'à devenir numériquement indifférentiables [87],[44, §2.3].*

3.2.2 Construction à la volée de l'état courant

Soit $X \in \mathbb{R}^n$ un point donné de l'espace d'état. Dans ce paragraphe, nous proposons un algorithme de calcul de l'état q du système hybride \mathcal{H} , i.e. du simplexe $D_q \in \mathcal{D}$, dans lequel se trouve le système à la position X .

Le principe de l'algorithme 3 ci-après basé sur la définition du maillage implicite de \mathbb{R}^n défini au paragraphe 3.1.2, est le suivant : on calcule tout d'abord le n -cube de \mathbb{R}^n qui contient X_0 , puis le simplexe du maillage de ce cube qui contient X_0 .

1. Calcul du n -cube qui contient X_0 .

D'après la définition 3.1.1, le n -cube qui contient X_0 est défini par un point origine $kh = (k_1h, \dots, k_nh) \in h\mathbb{Z}^n$, la longueur h de ses arêtes et les relations suivantes :

$$\forall i = 1, \dots, n, k_i h \leq x_i \leq (k_i + 1)h \text{ i.e. : } \forall i = 1, \dots, n, \frac{x_i}{h} - 1 \leq k_i \leq \frac{x_i}{h}$$

Par définition de la partie entière, on en déduit alors l'origine $kh = (k_1h, \dots, k_nh) \in \mathbb{Z}^n$ du n -cube cherché :

- Si $\frac{x_i}{h}$ n'est pas entier, alors : $k_i = \left\lfloor \frac{x_i}{h} \right\rfloor$.
- Si $\frac{x_i}{h}$ est un entier alors X_0 est à l'intersection de deux n -cubes. Par convention, on choisira le cube le plus proche de 0, soit :

$$k_i = \begin{cases} \frac{x_i}{h} - 1, & \text{si } x_i \geq 0 \\ \frac{x_i}{h}, & \text{si } x_i < 0 \end{cases}$$

2. Calcul du simplexe qui contient X_0 .

Nous avons calculé le n -cube $C = kh + [0, h]^n$ qui contient X_0 . D'après la proposition 3.1.1, C peut être subdivisé en $n!$ n -simplexes définis par :

$$\forall \varphi \in \mathcal{S}_n, D_\varphi = \{X \in \mathbb{R}^n ; 0 \leq X_{\varphi(1)} - k_{\varphi(1)}h \leq \dots \leq X_{\varphi(n)} - k_{\varphi(n)}h \leq h\} \quad (3.9)$$

Pour d'éviter d'avoir à calculer explicitement les $n!$ simplexes qui décomposent le n -cube C , l'idée est d'utiliser la définition (3.9) afin de calculer directement la permutation φ qui définit le simplexe cherché. Le principe est le suivant :

- i. On forme la liste : $L := [X_1 - k_1h, \dots, X_n - k_nh]$;
- ii. On applique un algorithme du type tri par sélection à la liste L , ce qui nous donne la liste triée \tilde{L} et la permutation φ sur les indices $\{1, \dots, n\}$ qui permet de passer de la liste L à la liste \tilde{L} .
- iii. On en déduit la liste des sommets du simplexe D_φ qui contient X_0 .

Remarque 3.2.2. *Il existe des algorithmes de tri plus performants que le tri par sélection : entre autres, le tri fusion ou le tri rapide - QuickSort en anglais. Actuellement, de nombreux problèmes issus de l'industrie, notamment de l'aéronautique, ne dépassent pas la dimension 6 ou 7. La liste à trier donnée en entrée étant de taille n i.e. la dimension de l'espace d'état, nous ne chercherons pas pour l'instant à optimiser l'algorithme de tri choisi.*

Algorithme 3 CurrentMode Calcul du simplexe contenant X

Données : $h > 0$ le pas de discrétisation, X un point de \mathbb{R}^n

Sortie : le simplexe de \mathbb{R}^n de taille h qui contient X

- 1: {Calcul du n -cube C de côté h qui contient X }
 - 2: **pour tout** i variant de 1 à n **faire**
 - 3: $k_i := \lceil \frac{x_i}{h} \rceil$;
 - 4: **fin pour**
 - 5: {Calcul du simplexe de C qui contient X_0 }
 - 6: On pose : $L = [X_1 - k_1h, \dots, X_n - k_nh]$; $\varphi := [1, 2, \dots, n]$;
 - 7: **pour tout** i variant de 1 à n **faire**
 - 8: $i_{max} := i$; {On parcourt la liste L }
 - 9: **pour tout** j variant de $i+1$ à n **faire**
 - 10: {On parcourt L à partir de l'indice i ; on cherche le premier indice $j > i$ dont l'élément est plus petit que $X_i - k_ih$ }
 - 11: **si** $X_j - k_jh < X_{i_{max}} - k_{i_{max}}h$ **alors**
 - 12: $i_{max} := j$;
 - 13: **fin si**
 - 14: **fin pour**
 - 15: Échanger $\varphi[i]$ et $\varphi[i_{max}]$;
 - 16: **fin pour**
 - 17: {Calcul des $n + 1$ sommets V^1, \dots, V^{n+1} du simplexe cherché grâce à (3.3)}
 - 18: **pour tout** p variant de 1 à $n+1$ **faire**
 - 19: **pour tout** i variant de 1 à p , **faire**
 - 20: $V_{\varphi[i]}^{p+1} := k_{\varphi[i]}h$;
 - 21: **fin pour**
 - 22: **pour tout** i variant de $p+1$ à n , **faire**
 - 23: $V_{\varphi[i]}^{p+1} := (k_{\varphi[i]} + 1)h$;
 - 24: **fin pour**
 - 25: **fin pour**
 - 26: Retourner $\{V^1, \dots, V^{n+1}\}$.
-

Le calcul de l'état courant nécessite donc $\mathcal{O}(n)$ opérations pour le calcul du cube qui contient X_0 , puis $\mathcal{O}(n)$ opérations pour former la liste L . Par application de l'algorithme de tri dont la complexité est en $\mathcal{O}(n \log n)$, nous avons alors le résultat suivant :

Proposition 3.2.2. *L'algorithme 3 de calcul de l'état courant du modèle hybride est correct et nécessite $\mathcal{O}(n \log n)$ opérations arithmétiques.*

3.2.3 Calcul des contraintes locales sur le contrôle

Soit $q \in \mathcal{Q}$ un mode quelconque de l'automate \mathcal{H} défini précédemment et $D_q \in \mathcal{D}$ la cellule associée dans l'espace d'état. Le domaine de contrôle dans le mode q est le polytope \mathbb{U}_m de dimension m dans \mathbb{R}^m . D'après le lemme 3.1.3, le domaine $D_q \times \mathbb{U}_m$ se décompose en simplexes $\Delta_{q'}$ de l'espace état-contrôle :

$$D_q \times \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} \Delta_{q'}$$

Toute cellule $\Delta_{q'}$, $q' \in \mathcal{K}(q)$, induit des contraintes mixtes sur l'état et le contrôle, ce qui signifie que, localement, les contraintes sur le contrôle dépendent de la position du système dans l'espace de contrôle. Dans ce paragraphe, nous voulons calculer les contraintes locales explicites sur le contrôle en fonction de la position. Pour cela, on définit $U_{q'}(X)$ l'ensemble des contraintes sur le contrôle dans la cellule $\Delta_{q'}$ à X fixé (voir figure 3.6) :

$$\forall X \in D_q, \forall q' \in \mathcal{K}(q), U_{q'}(X) = \{u \in \mathbb{R}^m / (X, u) \in \Delta_{q'}\} \quad (3.10)$$

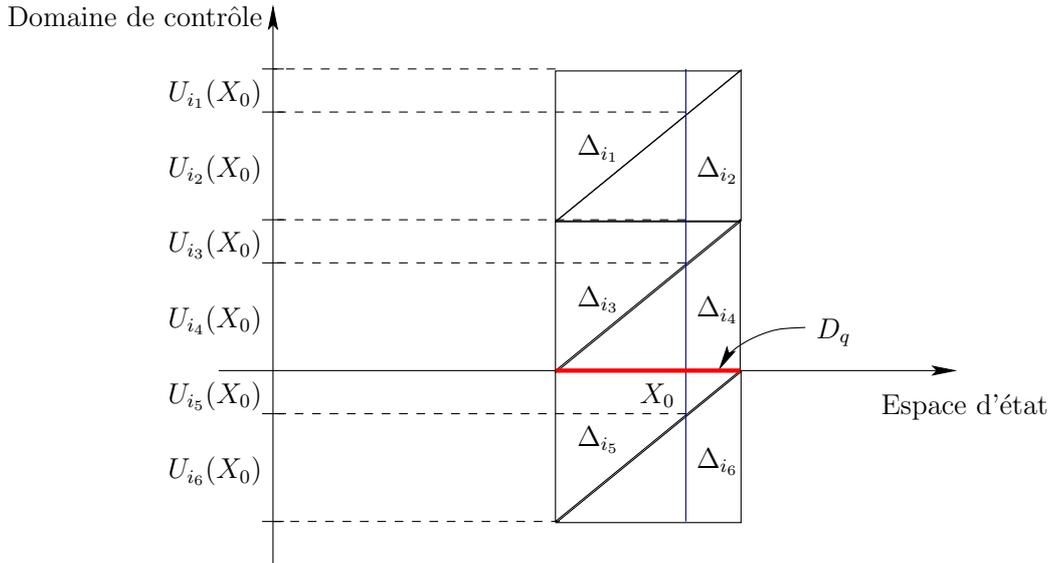


FIG. 3.6 – Définition de l'ensemble $U_{q'}(X_0)$ des contraintes locales sur le contrôle dans une cellule D_q

Par construction, le domaine de contrôle \mathbb{U}_m se décompose alors de la façon suivante :

Lemme 3.2.1. $\forall X \in D_q, \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} U_{q'}(X).$

Le problème est maintenant de déterminer la structure géométrique de $U_{q'}(X)$ dans \mathbb{R}^m .

Exemple 3.2.1. On se place dans le cas où l'état et le contrôle sont de dimension 1 ($n = m = 1$). Considérons une cellule $\Delta_{q'}$ de \mathbb{R}^2 définie par :

$$\Delta_{q'} = \text{Conv}((a_k, u_i), (a_k + h, u_i), (a_k + h, u_i + h))$$

Les contraintes mixtes sur l'état et le contrôle s'expriment sous la forme d'un système affine :

$$x \leq a_k + h, \quad u \geq u_i, \quad u - x \leq u_i - a_k$$

On en déduit facilement :

$$(x, u) \in \Delta_{q'} \Leftrightarrow \begin{cases} x \in D_q = [a_k, a_k + h] \\ u \in U_{q'}(x) = \{u \in \mathbb{R} / 0 \leq u - u_i \leq x - a_k\} \end{cases}$$

On remarque ainsi sur cet exemple que le domaine de contrôle est un segment (1-simplexe) dans \mathbb{R} si $x \in]a_k, a_k + h]$, ou est réduit à un point $\{u_i\}$ si $x = a_k$. Dans la suite, nous allons voir que ce résultat est généralisable en dimension quelconque.

Soient $X \in D_q$ et $q' \in \mathcal{K}(q)$. Par construction, $\Delta_{q'}$ est un simplexe de \mathbb{R}^{n+m} et peut donc être défini par un système de $(n + m + 1)$ inégalités affines indépendantes :

$$MY + d \geq 0 \text{ avec } M \in \mathbb{M}_{n+m+1, n+m}(\mathbb{R}) \text{ et } d \in \mathbb{R}^{n+m+1}$$

La matrice M se découpe en deux matrices :

$$M_1 = M \begin{bmatrix} I_n \\ 0 \end{bmatrix} \in \mathbb{M}_{n+m+1, n}(\mathbb{R}), \quad M_2 = M \begin{bmatrix} 0 \\ I_m \end{bmatrix} \in \mathbb{M}_{n+m+1, m}(\mathbb{R}),$$

de façon à avoir : $M = [M_1 \mid M_2]$. Le système d'inégalités devient donc : $M_1 X + M_2 X + d \geq 0$, ce qui nous donne une nouvelle caractérisation de $U_{q'}(X)$:

$$\begin{aligned} \forall X \in D_q, U_{q'}(X) &= \{u \in \mathbb{R}^m / (X, u) \in \Delta_{q'}\} \\ &= \{u \in \mathbb{R}^m / M_2 u + (M_1 X + d) \geq 0\} \end{aligned}$$

Ainsi, l'ensemble $U_{q'}(X)$ est caractérisé par un nombre fini d'inégalités affines sur le contrôle ; c'est donc un polyèdre de \mathbb{R}^m . De plus, $\Delta_{q'}$ étant borné, $U_{q'}(X)$ l'est aussi. $U_{q'}(X)$ est donc un polyèdre borné, i.e. un polytope de \mathbb{R}^m . On peut alors énoncer les résultats suivants :

Proposition 3.2.3. Soit $X_0 \in \overset{\circ}{D}_q$ et $q' \in \mathcal{K}(q)$. Alors $U_{q'}(X_0)$ est un m -simplexe de \mathbb{R}^m .

Corollaire 3.2.1. Soit $X_0 \in \partial D_q$ et $q' \in \mathcal{K}(q)$. Alors $U_{q'}(X_0)$ est soit un m -simplexe, soit un point de \mathbb{R}^m .

Preuve de la proposition 3.2.3. On se donne $X_0 \in \overset{\circ}{D}_q$ et $q' \in \mathcal{K}(q)$. Le problème revient en fait à calculer l'intersection du sous-espace affine :

$$P = \{(X_0, u) / u \in \mathbb{R}^m\} \subset \mathbb{R}^{n+m}$$

avec le simplexe $\Delta_{q'}$ (cf figure 3.7) :

$$U_{q'}(X_0) = \{(X_0, u) / u \in \mathbb{R}^m\} \cap \Delta_{q'} = P \cap \Delta_{q'}$$

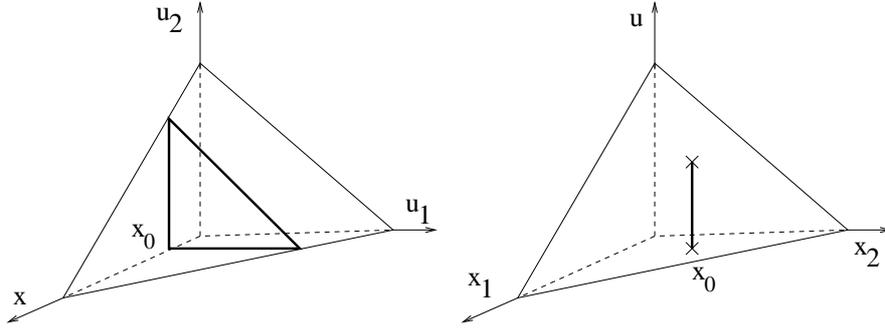


FIG. 3.7 – Intersection $P \cap \Delta_{q'}$ pour $(n, m) = (1, 2)$ à gauche, $(n, m) = (2, 1)$ à droite.

D'après [88, théorème 1.1], l'intersection d'un sous-espace affine avec un simplexe est un polytope. Donc $U_{q'}(X_0)$ est un polytope de \mathbb{R}^m . D'après la définition A.1.2 donnée en annexe, un m -simplexe est un polytope à $(m+1)$ sommets. La question est donc de calculer le nombre de sommets de $U_{q'}(X_0)$.

L'idée est d'intersecter le sous-espace P de dimension m avec les n -faces de $\Delta_{q'}$. Ces intersections, si elles existent, sont de dimension 0 et définissent les sommets de $U_{q'}(X_0)$. Il reste ensuite à prouver qu'il existe exactement $(m+1)$ n -faces différentes qui intersectent P pour pouvoir conclure.

Soit $\{X_1, \dots, X_{n+1}\}$ les sommets de D_q . Par définition, on a : $p_{\mathbb{R}^n}^\perp(\Delta_{q'}) = D_q$. On introduit alors les sommets de $\Delta_{q'}$:

$$V_k = \begin{bmatrix} X_{i_k} \\ u_{j_k} \end{bmatrix}, \quad k = 1, \dots, n+m+1$$

avec $i_k \in \{1, \dots, n+1\}$, $j_k \in \{1, \dots, m+1\}$ et $\{X_{i_k}\}_{k=1, \dots, n+m+1} = \{X_1, \dots, X_{n+1}\}$.

On suppose que $X_0 \in D_q$; X_0 s'écrit donc comme combinaison convexe stricte des sommets du simplexe D_q , i.e. :

$$\exists! (\alpha_k)_{k=1 \dots n+1} \in]0, 1[^{n+1}, \quad \sum_{k=1}^{n+1} \alpha_k = 1 \quad \text{et} \quad X_0 = \sum_{k=1}^{n+1} \alpha_k X_k \quad (3.11)$$

Première étape : toute n -face de $\Delta_{q'}$ admet au plus un point d'intersection avec le sous-espace affine P .

Soit F une n -face de $\Delta_{q'}$ définie par la liste (renumérotée pour faciliter la lecture) de ses sommets : $\left(\begin{bmatrix} X_{i_k} \\ u_{j_k} \end{bmatrix} \right)_{k=1, \dots, n+1}$ où $i_k \in \{1, \dots, n+1\}$, $j_k \in \{1, \dots, m+1\}$.

1^{er} cas : les i_k sont tous différents.

On a alors nécessairement : $\{X_{i_k} / k = 1, \dots, n+1\} = \{X_1, \dots, X_{n+1}\}$, ce qui nous permet de réécrire la relation convexe (3.11) sous la forme : $X_0 = \sum_{k=1}^{n+1} \alpha_{i_k} X_{i_k}$. On introduit alors le point X_F défini comme suit :

$$X_F = \sum_{k=1}^{n+1} \alpha_{i_k} \begin{bmatrix} X_{i_k} \\ u_{j_k} \end{bmatrix} = \begin{bmatrix} X_0 \\ \sum_{k=1}^{n+1} \alpha_{i_k} u_{j_k} \end{bmatrix}.$$

Par définition du sous-espace affine P , on a : $X_F \in P$. On a de plus supposé : $X_0 \in \overset{\circ}{D}_q$, ce qui implique que la relation convexe définissant X_F est également une relation stricte ; on en déduit alors : $X_F \in \overset{\circ}{F}$, soit : $X_F \in P \cap F$.

Par unicité de la décomposition convexe dans un simplexe, on peut conclure :

$$P \cap F = \{X_F\} \text{ et } X_F \in \overset{\circ}{F}$$

2ème cas : les i_k ne sont pas tous distincts.

On a alors : $\{X_{i_k} / k = 1, \dots, n+1\} \subsetneq \{X_1, \dots, X_{n+1}\}$. Soit Y un point quelconque de la face F : Y s'écrit comme combinaison convexe des sommets de F :

$$\exists! (\gamma_{i_k})_{k=1 \dots n+1} \in [0, 1]^{n+1}, \sum_{k=1}^{n+1} \gamma_{i_k} = 1 \text{ et } Y = \sum_{k=1}^{n+1} \gamma_{i_k} \begin{bmatrix} X_{i_k} \\ u_{j_k} \end{bmatrix}$$

Par ailleurs, on a supposé : $X_0 \in \overset{\circ}{D}_q$; X_0 dépend donc de tous les X_i , $i = 1, \dots, n+1$.

On a donc nécessairement : $\sum_{k=1}^{n+1} \gamma_{i_k} X_{i_k} \neq X_0$, soit : $Y \notin P$. On en déduit : $F \cap P = \emptyset$.

Deuxième étape : il existe exactement $(m+1)$ n -faces différentes qui intersectent le sous-espace affine P .

Par construction, on sait que : $\{X_{i_k} / k = 1, \dots, n+m+1\} = \{X_1, \dots, X_{n+1}\}$; quitte à renuméroter les sommets de $\Delta_{q'}$, nous pouvons donc supposer :

$$\forall l \in \{1, \dots, n+1\}, V_l = \begin{bmatrix} X_l \\ u_{j_l} \end{bmatrix}$$

- On pose : $F_0 = \text{Conv}(\{V_1, \dots, V_{n+1}\})$. D'après l'indépendance affine des sommets X_i , $i = 1, \dots, n+1$, de D_q , les sommets V_1, \dots, V_{n+1} de F_0 sont affinement indépendants. F_0 est donc une n -face de $\Delta_{q'}$ qui intersecte P .
- Soit $n+2 \leq l \leq n+m+1$ (i.e. m choix possibles). On définit :

$$F_l = \text{Conv}(\{V_1, \dots, V_{n+1}\} - \{V_{i_l}\} \cup \{V_l\}), \quad 1 \leq i_l \leq n+1$$

en remarquant que : $V_l = (X_{i_l}, u_{j_l})$ et $V_{i_l} = (X_{i_l}, u_{j_{i_l}})$. De même que pour F_0 , la face F_l est une n -face de $\Delta_{q'}$ qui intersecte P .

En résumé, nous avons trouvé exactement $(m + 1)$ n -faces différentes de $\Delta_{q'}$, qui intersectent P . Chacune de ses faces admet un unique point d'intersection avec P . Ces points étant à l'intérieur de chacune de ces n -faces, on a donc $(m + 1)$ points affinement indépendants appartenant à $P \cap \Delta_{q'} = U_{q'}(X_0)$. $U_{q'}(X_0) \subset \mathbb{R}^m$ est par conséquent un polytope de \mathbb{R}^m à $m + 1$ sommets, i.e. un m -simplexe de \mathbb{R}^m . □

Preuve du corollaire 3.2.1. Soit $X_0 \in \partial D_q$ et $q' \in \mathcal{K}(q)$. On suppose que le point X_0 appartient exactement à une k -face de D_q (avec $k < n$) : il existe donc $(k + 1)$ sommets de D_q , que l'on note X_1, \dots, X_{k+1} , tels que :

$$X_0 \in \overset{\circ}{F}_k,$$

en posant : $F_k = \text{Conv}(X_1, \dots, X_{k+1})$. On définit maintenant les ensembles :

$$\tilde{\Delta}_{q'} = \{(X, u) \in \Delta_{q'} ; X \in F_k\} \quad \text{et} \quad \tilde{U}_{q'}(X) = \{u \in \mathbb{U}_m ; (X, u) \in \tilde{\Delta}_{q'}\}.$$

On vérifie alors que :

$$\bigcup_{q' \in \mathcal{K}(q)} \tilde{\Delta}_{q'} = F_k \times \mathbb{U}_m \quad \text{et} \quad \tilde{U}_{q'}(X) = U_{q'}(X).$$

Par construction, F_k étant une k -face de D_q , $\tilde{\Delta}_{q'}$ est une face de $\Delta_{q'}$ de dimension au moins égale à k . On distingue alors deux cas :

Si $\tilde{\Delta}_{q'}$ est une k -face de $\Delta_{q'}$,

alors le domaine $U_{q'}(X_0)$ est nécessairement réduit à un point (démonstration par l'absurde basée sur le fait que X_0 s'écrit de façon unique comme combinaison convexe des sommets de F_k).

Si $\tilde{\Delta}_{q'}$ est de dimension supérieure à $k + 1$,

alors $\tilde{\Delta}_{q'}$ est nécessairement de dimension $k + m$. On peut donc maintenant appliquer la proposition 3.2.3 où F_k et $\tilde{\Delta}_{q'}$ jouent respectivement les rôles de D_q et $\Delta_{q'}$ et on conclut que $U_{q'}(X_0) = \tilde{U}_{q'}(X_0)$ est un m -simplexe de \mathbb{R}^m . □

Pour terminer, remarquons que par construction l'ensemble des domaines de contrôle $U_{q'}(X)$ pour $X \in D_q$ et $q' \in \mathcal{K}(q)$, est convexe :

Lemme 3.2.2. *Soient $(q_1, q_2) \in \mathcal{K}(q)$. Alors :*

$$\begin{aligned} u_1 \in U_{q_1}(X_1) \quad \text{et} \quad u_2 \in U_{q_2}(X_2) \\ \Downarrow \\ \forall \alpha \in [0, 1], \exists q' \in \mathcal{K}(q), \alpha u_1 + (1 - \alpha)u_2 \in U_{q'}(\alpha X_1 + (1 - \alpha)X_2) \end{aligned}$$

Preuve. Soit $\alpha \in [0, 1]$. D'après la définition (3.10) des domaines de contrôle, on sait que :

$$u_1 \in U_{q_1}(X_1), \quad u_2 \in U_{q_2}(X_2) \Rightarrow (X_1, u_1) \in \Delta_{q_1}, \quad (X_2, u_2) \in \Delta_{q_2}$$

Par hypothèses, $q_1 \in \mathcal{K}(q)$ et $q_2 \in \mathcal{K}(q)$, d'où : $X_1 \in D_q$ et $X_2 \in D_q$. La convexité de D_q implique : $\alpha X_1 + (1 - \alpha)X_2 \in D_q$ et celle de \mathbb{U}_m donne : $\alpha X_1 + (1 - \alpha)X_2 \in D_q$ et $\alpha u_1 + (1 - \alpha)u_2 \in \mathbb{U}_m$. D'où :

$$(\alpha X_1 + (1 - \alpha)X_2, \alpha u_1 + (1 - \alpha)u_2) \in D_q \times \mathbb{U}_m$$

De plus, par construction du maillage Δ , on sait que : $D_q \times \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} \Delta_{q'}$. Il existe donc un simplexe du maillage Δ qui contient le couple état-contrôle défini ci-dessus :

$$\exists q' \in \mathcal{K}(q), (\alpha X_1 + (1 - \alpha)X_2, \alpha u_1 + (1 - \alpha)u_2) \in \Delta_{q'},$$

ce qui prouve que : $\alpha u_1 + (1 - \alpha)u_2 \in U_{q'}(\alpha X_1 + (1 - \alpha)X_2)$.

□

Ainsi, nous avons construit un automate hybride modélisant le système non linéaire initial. La construction de cet automate se fait en pratique à la volée : à tout instant, nous sommes désormais capables de calculer l'état dans lequel se trouve le système ainsi les contraintes sur l'état et le contrôle auxquelles il est soumis.

Dans la partie suivante, nous allons voir comment déterminer un ensemble approché des conditions initiales à partir desquelles le système hybride \mathcal{H} peut atteindre la cible $0 \in \mathbb{R}^n$ en temps fini pour un contrôle bien choisi.

Deuxième partie

**Approximation du domaine
contrôlable**

Dans cette partie, nous nous intéressons à l'existence de solutions approchées du problème de contrôle non linéaire.

On considère un système de contrôle non linéaire très général :

$$\dot{X}(t) = f(X(t), u(t)).$$

La question de l'existence de solutions (non nécessairement optimales) est la suivante : Étant donné un point $X_0 \in \mathbb{R}^n$, existe-t-il un contrôle admissible u tel que la trajectoire associée à ce contrôle joigne X_0 à la cible $0 \in \mathbb{R}^n$ en temps fini ? Si oui, le point X_0 est dit *contrôlable* (cf figure 3.8).

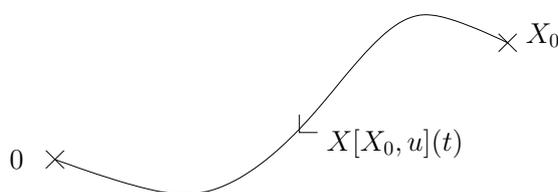


FIG. 3.8 – Problème de contrôlabilité

L'ensemble des points contrôlables de \mathbb{R}^n par le système considéré est appelé *domaine contrôlable* du système.

Cette partie est consacrée à la définition et l'approximation du domaine contrôlable des systèmes non linéaires. Dans le chapitre 4, nous définissons les concepts et les enjeux du problème de contrôlabilité des systèmes non linéaires. Après avoir étudié l'approximation du domaine contrôlable non linéaire par le domaine contrôlable hybride, nous proposons une approche constructive de la contrôlabilité d'un point initial donné par le biais des systèmes hybrides. Le chapitre 5 est consacré à l'étude et au calcul algorithmique du domaine contrôlable du modèle hybride proposé au chapitre 3 dans un domaine borné de l'espace d'état.

Chapitre 4

Différents niveaux de contrôlabilité

Dans cette partie, nous nous intéressons à la contrôlabilité à l'origine 0 de l'espace d'état \mathbb{R}^n des systèmes non linéaires du type :

$$\dot{X}(t) = f(X(t), u(t)). \quad (4.1)$$

Les contrôles admissibles u du système (4.1) sont des fonctions mesurables bornées à valeurs dans un polytope \mathbb{U}_m . Étant donné un point X_0 de \mathbb{R}^n , la question est de savoir s'il existe un contrôle u admissible tel que la trajectoire associée à u joigne X_0 à la cible 0 en temps fini.

La contrôlabilité à l'origine est un problème issu du concept plus général de la contrôlabilité complète. Un système dynamique est dit complètement contrôlable s'il existe une trajectoire admissible de ce système qui relie deux points donnés de l'espace d'état en temps fini.

La notion de contrôlabilité est apparue dans les années soixante avec les travaux de R. E. Kalman [56, 57] dans le cadre des systèmes linéaires autonomes du type : $\dot{X}(t) = AX(t) + Bu(t)$. Ces systèmes ont suscité de nombreux travaux de recherche et nous disposons aujourd'hui de résultats puissants, à commencer par le fameux critère de contrôlabilité de Kalman [58, 61, 28] en l'absence de contraintes sur l'état et le contrôle. Ces résultats jouent un rôle essentiel dans l'étude du domaine de contrôlabilité nulle, i.e. de l'ensemble des points de \mathbb{R}^n contrôlables jusqu'à l'origine 0 [61, paragraphe 2.3].

La contrôlabilité des systèmes non linéaires tels que (4.1), est un sujet extrêmement complexe. Une première approche de ce problème consiste à étendre les résultats de contrôlabilité obtenus pour les systèmes linéaires aux systèmes non linéaires. Par des arguments de linéarisation locale autour d'un point (X_0, u_0) donné de l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$, le système (4.1) est remplacé localement par le système linéaire $\dot{X}(t) = AX(t) + Bu(t)$ en posant :

$$A = \frac{\partial f}{\partial X}(X_0, u_0), \quad B = \frac{\partial f}{\partial u}(X_0, u_0).$$

L'étude de la contrôlabilité du système linéarisé permet d'en déduire des résultats de *contrôlabilité locale* du système non linéaire initial [61, 81]. Une autre approche consiste à relier la contrôlabilité à des propriétés de crochets de Lie du système non linéaire. Mais les résultats restent locaux [1, 51, 55]. Pour la classe particulière des systèmes affines par rapport au contrôle, des critères de contrôlabilité globale peuvent être obtenus [55, 62].

Dans le cas général, la contrôlabilité des systèmes non linéaires demeure un problème difficile. L'objectif de cette partie est de développer une approche constructive de la contrôlabilité des systèmes non linéaires. Nous ne nous intéressons donc pas uniquement à l'existence d'un critère de contrôlabilité des systèmes non linéaires mais à la mise en oeuvre de méthodes et d'algorithmes pour l'approximation du domaine contrôlable non linéaire.

La méthode proposée dans cette partie est basée sur l'approximation hybride du domaine contrôlable du système initial (4.1). On introduit le système hybride $\mathcal{H} = (\mathcal{Q}, \mathcal{E}, \mathcal{D}, \mathcal{U}, \mathcal{F}, \mathcal{G}, \mathcal{R})$ construit au chapitre 3 qui modélise le système non linéaire (4.1). La dynamique non linéaire est donc approchée par le système suivant :

$$\dot{X}(t) = f_h(X(t), u(t)), \quad (4.2)$$

où f_h est l'approximation affine par morceaux du champ de vecteurs non linéaire f calculée au chapitre 2. Le but de ce chapitre est d'étudier la contrôlabilité et le domaine contrôlable du système non linéaire via son approximation hybride.

Après avoir introduit les notions de contrôlabilité et de domaine contrôlable pour un système dynamique quelconque, nous nous intéressons dans ce chapitre à l'approximation hybride de la contrôlabilité. La partie 4.2 est donc consacrée à l'étude de l'approximation hybride du domaine contrôlable non linéaire. Nous proposons ensuite dans la partie 4.3, une approche constructive du problème de contrôlabilité à travers l'étude du domaine contrôlable hybride défini dans un ensemble non ordonné de cellules de l'espace d'état.

4.1 Contrôlabilité et domaine contrôlable

Intuitivement, la contrôlabilité d'un point X_0 vers la cible 0 correspond à l'existence d'une trajectoire admissible du système de contrôle considéré, qui relie X_0 à 0 en temps fini. Dans ce paragraphe, nous allons formaliser cette notion pour les systèmes de contrôle non linéaires.

4.1.1 Contrôlabilité des systèmes non linéaires ou atteignabilité ?

On considère le système de contrôle non linéaire suivant :

$$\dot{X}(t) = f(X(t), u(t)).$$

On rappelle que les contrôles admissibles u sont des fonctions mesurables à valeurs dans un polytope \mathbb{U}_m , défini comme l'enveloppe convexe de ses sommets : $\mathbb{U}_m = \text{Conv}(s_1, \dots, s_p)$. La contrôlabilité à l'origine d'un point X_0 de l'espace d'état est définie de la façon suivante :

Définition 4.1.1 (Domaine contrôlable). *Le point $X_0 \in \mathbb{R}^n$ est contrôlable vers 0 pour le système (4.1) si et seulement s'il existe un temps fini $T \geq 0$ et un contrôle admissible $u : [0, T] \rightarrow \mathbb{U}_m$, tels que le problème :*

$$\begin{cases} \dot{X}(t) &= f(X(t), u(t)) \\ X(0) &= X_0 \end{cases}, \quad X(T) = 0$$

admette une solution $X(\cdot)$ définie sur $[0, T]$.

L'ensemble des points contrôlables de \mathbb{R}^n est appelé domaine contrôlable du système (4.1).

Remarque 4.1.1. *Par abus de langage, le terme contrôlabilité désigne, à partir de maintenant la contrôlabilité à l'origine du système considéré. De même, sauf mention contraire, on parlera du domaine contrôlable d'un système pour désigner l'ensemble des points de l'espace d'état, contrôlables jusqu'à la cible 0.*

Or le champ de vecteurs f vérifie les hypothèses du théorème de Cauchy-Lipschitz (cf théorème 2.3.1) ce qui, pour un contrôle admissible donné, garantit l'existence et l'unicité d'une solution du problème de Cauchy formé du système (4.1) associé à la condition initiale : $X(0) = X_0$. En revanche, pour tout point contrôlable de l'espace d'état, il peut exister plusieurs trajectoires admissibles du système (4.1) le reliant à la cible.

Exemple 4.1.1. *On considère le système :*

$$\dot{X}(t) = \begin{bmatrix} -1 & -2 \\ -3 & -1 \end{bmatrix} u(t), \quad (4.3)$$

sachant que le contrôle u est à valeurs dans le simplexe unité \mathbb{U}_2 de \mathbb{R}^2 , soit :

$$\forall t, u(t) \in \mathbb{U}_2 = \text{Conv}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right)$$

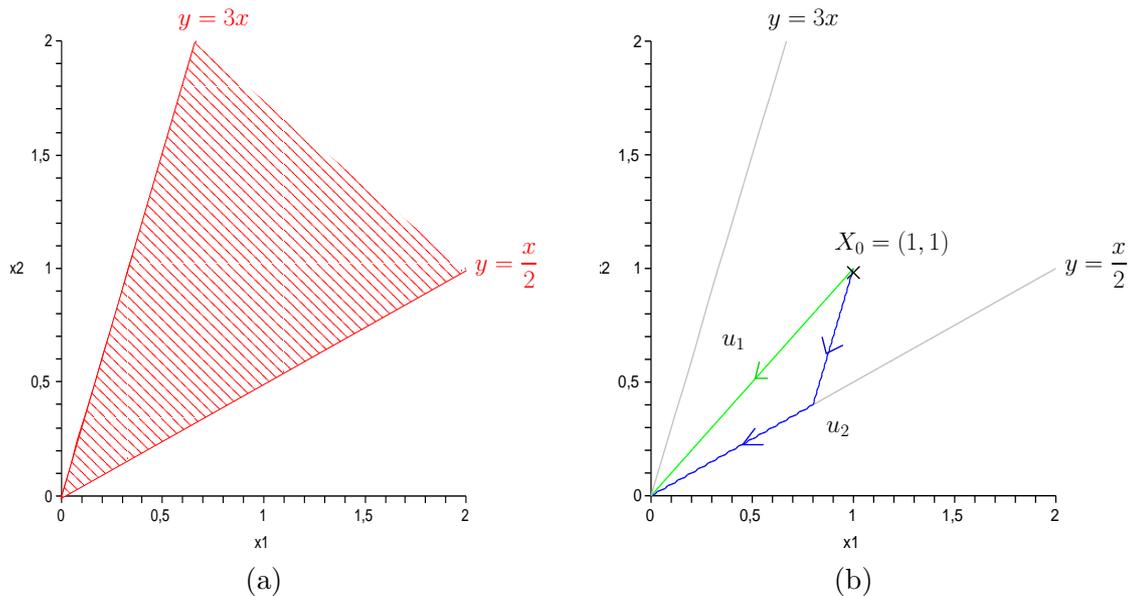


FIG. 4.1 – (a) Domaine contrôlable et (b) Trajectoires admissibles reliant X_0 à 0 en temps fini, du système (4.3).

Le domaine contrôlable du système (4.3) est le cône de \mathbb{R}^2 , défini par les demi-droites d'équations respectives $(y = \frac{x}{2})$ et $(y = 3x)$ dans le quart de plan positif (cf figure 4.1-(a)). La figure 4.1-(b) donne plusieurs exemples de trajectoires admissibles reliant le point $X_0 = (1, 1)$ à $(0, 0)$ pour les contrôles :

$$\forall t, u_1(t) = \left(\frac{1}{5}, \frac{2}{5}\right) \quad \text{et} \quad u_2(t) = \begin{cases} (-1, -3) & \text{si } x(t) - 2y(t) < 0 \\ (-2, -1) & \text{si } x(t) - 2y(t) = 0 \end{cases}$$

Par ailleurs, la notion de contrôlabilité d'un système dynamique est étroitement liée à la notion d'atteignabilité.

Considérons un point $X_0 \in \mathbb{R}^n$ supposé contrôlable jusqu'en 0 par le système (4.1). D'après la définition 4.1.1, il existe donc une trajectoire admissible du système (4.1) qui relie X_0 à 0 en temps fini. Ainsi, pour atteindre X_0 à partir de 0, l'idée est tout simplement de parcourir cette même trajectoire par renversement du temps en partant cette fois-ci de la cible :

Proposition 4.1.1. *Soit $X_0 \in \mathbb{R}^n$ un point donné de l'espace d'état. Le point X_0 est contrôlable à l'origine par le système non linéaire :*

$$\dot{X}(t) = f(X(t), u(t)), \quad u(t) \in \mathbb{U}_m,$$

si et seulement si X_0 est atteignable à partir de 0 en temps fini par le système :

$$\dot{X}(t) = -f(X(t), v(t)), \quad v(t) \in \mathbb{U}_m.$$

Preuve. Supposons X_0 contrôlable jusqu'en 0 par le système $\dot{X}(t) = f(X(t), u(t))$. D'après la définition 4.1.1, il existe un temps fini $T > 0$ et un contrôle admissible $u : [0, T] \rightarrow \mathbb{U}_m$ tels que le système (4.1) associé à la condition initiale $X(0) = X_0$, admette une solution $X(\cdot)$ vérifiant : $X(T) = 0$, ou de manière équivalente :

$$\begin{cases} X(t) &= X_0 + \int_0^t f(X(s), u(s)) ds \\ X(T) &= 0 \end{cases}$$

ce qui nous permet d'écrire au temps T la relation suivante :

$$X_0 = - \int_0^T f(X(s), u(s)) ds.$$

Nous voulons maintenant prouver que le point X_0 est atteignable à partir de 0 par renversement du temps. Pour tout temps $t \in [0, T]$, on introduit alors la trajectoire :

$$Y(t) = X(T - t),$$

et le contrôle :

$$v(t) = u(T - t).$$

On vérifie alors sans peine que la trajectoire Y est solution du système :

$$\begin{cases} \dot{Y}(t) &= -f(Y(t), v(t)) \\ Y(0) &= 0 \end{cases}, \quad Y(T) = 0.$$

La réciproque se démontre selon le même procédé : on suppose X_0 atteignable à partir de 0 en temps T pour le système : $\dot{Y}(t) = -f(Y(t), v(t))$. Grâce au même changement de variable que précédemment, on vérifie que X_0 est contrôlable en temps T pour le système (4.1).

□

En particulier, si, quel que soit le contrôle $u \in \mathbb{U}_m$, l'application $X \rightarrow f(X, u)$ est paire, i.e. :

$$\forall u \in \mathbb{U}_m, \forall X \in \mathbb{R}^n, f(-X, u) = f(X, u),$$

alors, d'après la proposition précédente, la contrôlabilité du système (4.1) depuis un point $X_0 \in \mathbb{R}^n$ est équivalente, par renversement du temps, à l'atteignabilité de X_0 depuis la cible 0 pour le système :

$$-\dot{Y}(t) = f(Y(t), v(t)), \text{ soit : } (-\dot{Y})(t) = f(-Y(t), v(t)).$$

Par le changement de variable $Z(t) = -Y(t)$, on se ramène alors au système initial : $\dot{Z}(t) = f(Z(t), v(t))$. Dans ces conditions, le problème de contrôlabilité à l'origine du système (4.1) est donc équivalent à l'étude des ensembles atteignables depuis l'origine 0 pour le même système.

4.1.2 Cas des systèmes linéaires : étude de la partie non contrôlable

Considérons le cas particulier où la dynamique f est affine :

$$\dot{X}(t) = AX(t) + Bu(t), \quad (4.4)$$

sans contraintes sur l'état X du système. La forme canonique du système linéaire (4.4) calculée au paragraphe 2.2.2, met en évidence l'existence d'une partie contrôlable :

$$\dot{Y}_1(t) = A_1 Y_1(t) + A_2 Y_2(t) + B_1 \tilde{u}(t), \quad (4.5)$$

et d'une partie non contrôlable (i.e. indépendante du contrôle u) :

$$\dot{Y}_2(t) = A_3 Y_2(t) \quad (4.6)$$

en posant : $Y = (Y_1, Y_2) = T_i^{-1} X \in \mathbb{R}^r \times \mathbb{R}^{n-r}$ où T_i est la matrice de passage du théorème de décomposition canonique (cf théorème 2.2.1).

La question de la contrôlabilité du système canonique se ramène ainsi à la résolution de deux problèmes de contrôlabilité successifs :

1. Calculer l'ensemble des points $Y_2 \in \mathbb{R}^{n-r}$ tels que la cible $0 \in \mathbb{R}^{n-r}$ soit atteignable à partir de Y_2 .
2. Calculer l'ensemble des points $Y_1 \in \mathbb{R}^r$ tels qu'il existe un contrôle (non optimal a priori) qui amène le système de l'état Y_1 à la cible $0 \in \mathbb{R}^r$ en temps fini.

Dans ce paragraphe, nous nous intéressons au premier problème, i.e. au calcul de l'ensemble des points de l'espace d'état, qui peuvent être amenés en 0 en temps fini. Nous proposons ensuite une étude de l'atteignabilité de la cible 0 en temps infini. Dans tous les cas, l'étude de la partie non contrôlable nous permet de ramener le problème de contrôlabilité du système canonique à celui d'un système de contrôle linéaire de rang plein, soumis ou non à des perturbations.

Contrôlabilité à l'origine d'un système linéaire canonique

Nous nous intéressons pour le moment à la partie non contrôlable du système (4.4). Le système associé est une équation différentielle autonome d'ordre 1, que l'on sait résoudre :

$$\dot{Y}_2(t) = A_3 Y_2(t) \iff Y_2(t) = e^{A_3 t} Y_2(0)$$

Or, d'après la proposition 4.1.1, tout problème de contrôlabilité est équivalent par renversement du temps à un problème d'atteignabilité en temps fini. On note $R(I)$ l'ensemble atteignable par le système (4.6) à partir d'un ensemble de conditions initiales I (cf [44, chapitre 6]) :

$$R(I) = \{e^{A_3 t} Y_2, Y_2 \in I, t \in \mathbb{R}^+\} \subset \mathbb{R}^{n-r}.$$

Avec ces notations, l'ensemble des points contrôlables du système (4.6) est également l'ensemble atteignable $R(\{0\})$ du système pour la condition initiale 0. Par ailleurs le point $0 \in \mathbb{R}^{n-r}$ est un point d'équilibre du système (4.6), ce qui implique nécessairement :

$$R(\{0\}) = \{0\}.$$

La cible $0 \in \mathbb{R}^{n-r}$ n'est donc pas atteignable en temps fini, sauf à partir de 0.

Le domaine de contrôlabilité du système canonique est donc nécessairement inclus dans le sous-espace $V = \{(Y_1, 0) \in \mathbb{R}^n / Y_1 \in \mathbb{R}^r\}$ de dimension r de \mathbb{R}^n . Dans ce sous-espace, on a de plus : $\forall t \geq 0, Y_2(t) = 0$ et le système canonique s'écrit alors :

$$\dot{Y}_1(t) = A_1 Y_1(t) + B_1 \tilde{u}(t)$$

la paire (A_1, B_1) étant contrôlable (d'après le théorème de Kalman). Nous sommes donc ramené à un système de contrôle linéaire de rang plein.

Atteignabilité de la cible en temps infini

Nous avons vu au paragraphe précédent, que la cible 0 est un point d'équilibre du système non contrôlable (4.6). Pour que 0 soit atteignable, cette fois-ci en temps infini, de n'importe quel point de \mathbb{R}^{n-r} , il suffit que 0 soit un foyer stable du système. En d'autres termes, il suffit que la matrice A_3 soit stable (i.e. toutes ses valeurs propres ont une partie réelle strictement négative).

Dans le cas contraire, l'idée est de décomposer la matrice A_3 en une matrice stable et une matrice instable et d'en déduire les conditions initiales adéquates pour que la cible puisse être atteinte.

Théorème 4.1.1 ([45] Décomposition de Schur). *Soit A une matrice carrée complexe d'ordre n . Il existe une matrice unitaire $Q \in \mathbb{M}_n(\mathbb{C})$, telle que :*

$$Q^* A Q = D + N$$

où $D = \text{diag}(\alpha_1, \dots, \alpha_n)$ et $N \in \mathbb{M}_n(\mathbb{C})$ est strictement triangulaire supérieure.

Par ailleurs, Q peut être choisie de telle façon que les valeurs propres α_i apparaissent dans un ordre donné le long de la diagonale.

Appliquons la décomposition de Schur à la matrice A_3 , en choisissant d'ordonner les valeurs propres selon le critère suivant :

$$\begin{aligned} \forall i = 1, \dots, k, & \quad Re(\alpha_i) < 0 \\ \forall i = k + 1, \dots, n - r, & \quad Re(\alpha_i) \geq 0 \end{aligned}$$

où k dénote le nombre de valeurs propres de A_3 de partie réelle strictement négatives.

Le système $Y_2(t) = e^{A_3 t} Y_2(0)$ s'écrit :

$$\begin{aligned} Q^* Y_2(t) &= e^{Q^* A_3 Q t} Q^* Y_2(0) = e^{(D+N)t} Q^* Y_2(0) \\ &= \begin{bmatrix} e^{\alpha_1 t} & \star & \star \\ & & \star \\ & & e^{\alpha_{n-r} t} \end{bmatrix} Q^* Y_2(0) \end{aligned}$$

On cherche les solutions de $\lim_{t \rightarrow +\infty} Q^* Y_2(t) = 0$. Remarquons qu'il n'est pas nécessaire de calculer la matrice $e^{(D+N)t}$, seuls les coefficients diagonaux nous intéressent. Le calcul de ces coefficients est immédiat : $e^{\alpha_i t}$. Récursivement de $(n - r)$ à $k + 1$, on démontre que toute valeur propre α_i de partie réelle positive ou nulle, correspond une condition initiale égale à 0, soit :

$$[0 \mid I_{n-r-k}] Q^* Y_2(0) = 0$$

On peut alors définir un nouveau changement de variable : $\tilde{Y}_2 = [I_k \mid 0] Q^* Y_2$ et le système canonique devient :

$$\begin{cases} \dot{\tilde{Y}}_1(t) = A_1 Y_1(t) + \tilde{A}_2 \tilde{Y}_2(t) + B_1 \tilde{u}(t) \\ \dot{\tilde{Y}}_2(t) = \tilde{A}_3 \tilde{Y}_2(t) \end{cases}$$

en posant : $\tilde{A}_2 = A_2 Q \begin{bmatrix} I_k \\ 0 \end{bmatrix}$ et $\tilde{A}_3 = (D + N) \begin{bmatrix} I_k \\ 0 \end{bmatrix}$, où \tilde{A}_3 est stable.

Le domaine de contrôlabilité du système canonique en temps infini est donc inclus dans le sous-espace $\{(Y_1, \tilde{Y}_2, 0) \in \mathbb{R}^r \times \mathbb{R}^k \times \mathbb{R}^{n-r-k}\}$. Dans ce sous-espace, les trajectoires convergent en temps infini vers la cible 0 et on se ramène à l'étude de contrôlabilité d'un système de contrôle linéaire de rang plein en dimension r soumis à des perturbations :

$$\dot{Y}_1(t) = A_1 Y_1(t) + B_1 \tilde{u}(t) + r(t) \quad \text{avec : } r(t) = \tilde{A}_2 e^{\tilde{A}_3 t} \tilde{Y}_2(0)$$

sachant que : $\lim_{t \rightarrow +\infty} r(t) = 0$ (puisque la matrice \tilde{A}_3 est stable).

4.1.3 Contrôlabilité de l'automate hybride

La définition 4.1.1 est une définition très générale, qui s'applique à tout système de contrôle et donc en particulier au modèle hybride \mathcal{H} construit au chapitre 3.

Un point X_0 donné de l'espace d'état est contrôlable jusqu'à la cible 0 par le système \mathcal{H} si et seulement s'il existe une trajectoire admissible de \mathcal{H} qui relie X_0 à 0 en temps fini. Cette définition fait intervenir le concept de solution (on dit aussi *exécution*) d'un système hybride que nous précisons ci-après.

Notion de solution d'un système hybride

L'étude des systèmes hybrides et de leurs solutions ont suscité de nombreux travaux sur le sujet [15, 63, 44]. Nous proposons ici une définition synthétique inspirée de ces travaux :

Définition 4.1.2 (Exécution d'un système hybride). Soit $\mathcal{H}_g = (\mathcal{Q}, \mathcal{E}, \mathcal{D}, \mathcal{U}, \mathcal{F}, \mathcal{G})$ un automate hybride quelconque. Une solution (on dit aussi exécution) du système hybride \mathcal{H}_g est une trajectoire hybride (τ, γ, X) , où :

- $\tau = ([t_i, t_{i+1}])_{i=0\dots r}$ ($t_i \leq t_{i+1}$ pour $i < r$) est une suite finie ou infinie d'intervalles de \mathbb{R} .
- $\gamma : \tau \rightarrow \mathcal{Q}$ décrit l'état discret du système hybride,
- $X : \tau \rightarrow \mathbb{R}^n$ décrit l'évolution continue du système hybride,

et vérifiant, pour tout $i \in \{0, \dots, r\}$, tel que $t_i < t_{i+1}$,

- $t \mapsto \gamma(t)$ est constante sur l'intervalle $]t_i, t_{i+1}[$.
- $t \mapsto X(t) \in D_{\gamma(t)}$ sur l'intervalle $]t_i, t_{i+1}[$.
- il existe une application mesurable $u :]t_i, t_{i+1}[\rightarrow U_{\gamma(t)}$ telle que :

$$\forall t \in]t_i, t_{i+1}[, \dot{X}(t) = f_{\gamma(t)}(X(t), u(t)),$$

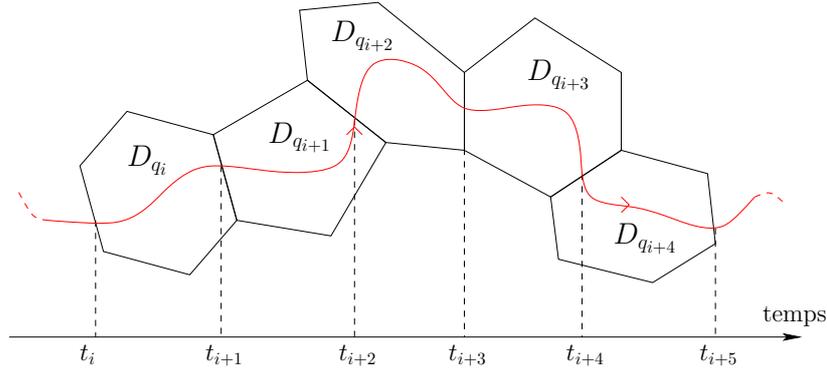


FIG. 4.2 – Représentation d'une exécution d'un système hybride affine par morceaux.

Une illustration de cette définition est donnée sur la figure 4.2. La fonction γ décrit l'état discret et la variable X l'évolution continue du système hybride dans l'espace d'état \mathbb{R}^n au cours du temps. Pour tout intervalle de temps $]t_i, t_{i+1}[$ non vide, le système se trouve dans le mode $\gamma(t) = q_i$, à la position $X(t)$ dans la cellule D_{q_i} . Dans le mode q_i , il existe un contrôle admissible u tel que l'évolution continue du système soit régie par l'équation différentielle : $\dot{X}(t) = f_{q_i}(X(t), u(t))$. Les temps t_i et t_{i+1} correspondent respectivement aux temps d'entrée et de sortie de la trajectoire $X(\cdot)$ de la cellule D_{q_i} .

Contrôlabilité des systèmes hybrides

Maintenant que le concept de solution d'un système hybride a été bien précisé, nous sommes en mesure de redéfinir la contrôlabilité d'un point donné de l'espace d'état pour le modèle hybride \mathcal{H} :

Définition 4.1.3. *Le point $X_0 \in \mathbb{R}^n$ est contrôlable vers 0 par \mathcal{H} s'il existe une exécution finie $\chi = (\tau, \gamma, X)$ de \mathcal{H} satisfaisant :*

- i. $(\gamma(t_0), X(t_0)) = (q_0, X_0)$ et $(\gamma(t_{r+1}), X(t_{r+1})) = (q_r, 0)$
- ii. Pour tout $i \in \{0, \dots, r\}$, tel que $t_i < t_{i+1}$:
 - $\forall t \in]t_i, t_{i+1}[$, $\gamma(t) = q_i$ et $X(t) \in D_{q_i}$.
 - il existe une fonction de contrôle $u(\cdot)$ mesurable, définie sur $]t_i, t_{i+1}[$ à valeurs dans \mathbb{U}_m telle que :

$$\forall t \in]t_i, t_{i+1}[, \dot{X}(t) = f_h(X(t), u(t))$$

- iii. $\forall i \in \{1, \dots, r\}$, $X(t_i) \in G_{(q_{i-1}, q_i)}$.

en posant : $\tau = ([t_i, t_{i+1}])_{i=0\dots r}$ et $\gamma = (q_i)_{i=0\dots r}$.

L'ensemble des points de \mathbb{R}^n contrôlables jusqu'en 0 par l'automate hybride \mathcal{H} , noté $\mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)$, est appelé domaine contrôlable de \mathcal{H} .

À toute suite $\gamma = (q_i)_{i=0\dots r}$ de modes de l'automate hybride considéré, on associe le domaine Ω_γ défini de la façon suivante :

$$\Omega_\gamma = \bigcup_{i=0}^r D_{q_i},$$

défini comme l'union d'un nombre fini de cellules du maillage \mathcal{D} de l'espace d'état. Deux modes q et q' sont dits adjacents si et seulement si : $D_q \cap D_{q'} \neq \emptyset$.

Un point X_0 donné de l'espace d'état est contrôlable jusqu'en 0 par l'automate hybride \mathcal{H} à condition qu'il existe un chemin γ fini et continu de cellules de \mathcal{D} et une trajectoire du système \mathcal{H} dans le domaine Ω_γ associé à γ , qui relie X_0 à la cible 0 en temps fini.

Remarque 4.1.2. *Pour tout point contrôlable X_0 de l'espace d'état, le système peut accepter plusieurs exécutions qui transfèrent \mathcal{H} de la position X_0 à la cible 0 et donc plusieurs chemins de cellules possibles entre 0 et X_0 .*

4.2 Approche hybride du domaine contrôlable non linéaire

Au chapitre 3, nous avons construit un modèle hybride approchant un système de contrôle non linéaire donné. Le problème est donc maintenant de vérifier que le domaine contrôlable du modèle hybride est bien une approximation du domaine contrôlable du système non linéaire initial et d'évaluer la qualité de cette approximation.

4.2.1 Approximation d'ensembles : la distance d'Hausdorff

Afin d'évaluer l'erreur commise entre les domaines contrôlables non linéaire et hybride, on introduit la notion de distance sur des ensembles :

Définition 4.2.1 (Distance de Hausdorff [37]). *Soit $\|\cdot\|$ une norme définie sur \mathbb{R}^n . L'application $d_H : \mathcal{P}(\mathbb{R}^n) \rightarrow \mathbb{R}_+$ définie par :*

$$d_H(E, F) = \max \left(\sup_{x \in E} \inf_{y \in F} \|x - y\|, \sup_{y \in F} \inf_{x \in E} \|x - y\| \right),$$

est une distance sur l'ensemble $\mathcal{P}(\mathbb{R}^n)$ des parties de \mathbb{R}^n .

La distance de Hausdorff vérifie la propriété suivante :

Proposition 4.2.1 (Propriétés de la distance d’Hausdorff [37]).

1. Soient E et F des parties de \mathbb{R}^n , telles que : $E \subset F$. Alors :

$$d_H(E, F) = \sup_{y \in F} \inf_{x \in E} \|x - y\|$$

2. Soient E_1, E_2, F_1 et F_2 des parties de \mathbb{R}^n .

$$\forall \delta > 0, [d_H(E_1, F_1) \leq \delta \text{ et } d_H(E_2, F_2) \leq \delta \Rightarrow d_H(E_1 \cup E_2, F_1 \cup F_2) \leq \delta]$$

Cette distance va nous permettre d’évaluer l’erreur d’approximation commise entre le domaine contrôlable non linéaire et le domaine contrôlable hybride.

4.2.2 Étude de l’erreur globale d’approximation

On considère les domaines contrôlables $\mathcal{C}_{NL}(\mathbb{R}^n)$ et $\mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)$ associés respectivement au système non linéaire : $\dot{X}(t) = f(X(t), u(t))$ et au modèle hybride \mathcal{H} de ce système.

On veut calculer (ou du moins majorer) la distance de Hausdorff

$$d_H(\mathcal{C}_{NL}(\mathbb{R}^n), \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)) = \max \left(\sup_{X \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \inf_{Y \in \mathcal{C}_{NL}(\mathbb{R}^n)} \|X - Y\|, \sup_{Y \in \mathcal{C}_{NL}(\mathbb{R}^n)} \inf_{X \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \|X - Y\| \right)$$

entre ces deux ensembles.

Soit $X_0 \in \mathcal{C}_{NL}(\mathbb{R}^n)$ un point contrôlable jusqu’en 0 du système non linéaire. Par définition, il existe donc un temps fini $T > 0$ et un contrôle $u_0 : [0, T] \rightarrow \mathbb{U}_m$ admissible tels que le système :

$$\begin{cases} \dot{X}(t) &= f(X(t), u_0(t)) \\ X(0) &= X_0 \end{cases} \quad (4.7)$$

admette une solution X vérifiant : $X(T) = 0$.

L’idée développée ci-après est d’introduire un point $Y_{h,0}$ qui soit contrôlable par le système hybride en temps T et pour le même contrôle u_0 que pour X_0 . L’astuce consiste alors à introduire le système :

$$\begin{cases} \dot{Y}(t) &= -f_h(Y(t), u_0(T-t)) \\ Y(0) &= 0 \end{cases}, \quad t \in [0, T]. \quad (4.8)$$

obtenu par renversement du temps à partir du système non linéaire initial pour le contrôle $u_0(T - \cdot)$. D’après le théorème de Cauchy-Lipschitz, on sait que ce système admet une unique solution \tilde{Y} sur l’intervalle de temps $[0, T]$. On pose alors :

$$Y_{h,0} = \tilde{Y}(T).$$

D’après la proposition 4.1.1, puisque $Y_{h,0}$ est atteignable en temps T à partir de 0 pour le couple $(\tilde{Y}, u_0(T - \cdot))$, $Y_{h,0}$ est de manière équivalente contrôlable en temps T pour le couple (Y_h, u_0) en posant $Y_h = \tilde{Y}(T - \cdot)$. On en déduit donc :

$$Y_{h,0} \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n),$$

ce qui nous permet d'obtenir la majoration suivante :

$$\inf_{Y \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \|X_0 - Y\| \leq \|X_0 - Y_{h,0}\|$$

De plus, au chapitre 2, nous avons établi différents résultats de convergence des trajectoires du système hybride vers celles du système non linéaire, suivant les hypothèses choisies sur le champ non linéaire f .

En particulier, si f est supposée globalement lipschitzienne par rapport à X et uniformément par rapport à u , le théorème 2.3.2 assure la convergence C^1 des trajectoires $Y_h(T - \cdot)$ du système hybride vers celles $X(T - \cdot)$ du système non linéaire, soit :

$$\|X(T - t) - Y_h(T - t)\| \leq \frac{\varepsilon(h)}{L}(e^{Lt} - 1),$$

ce qui implique en $t = T$:

$$\|X_0 - Y_{h,0}\| \leq \frac{\varepsilon(h)}{L}(e^{LT} - 1) \text{ soit : } \inf_{Y \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \|X_0 - Y\| \leq \frac{\varepsilon(h)}{L}(e^{LT} - 1)$$

où $\varepsilon(h)$ désigne l'erreur d'interpolation commise lors de la construction du modèle hybride \mathcal{H} . Nous venons donc de montrer le résultat suivant :

$$\forall X_0 \in \mathcal{C}_{NL}(\mathbb{R}^n), \exists T \geq 0, \inf_{Y \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n)} \|X_0 - Y\| \leq \frac{\varepsilon(h)}{L}(e^{LT} - 1) \quad (4.9)$$

En partant cette fois-ci d'un point $Y_{h,0}$ contrôlable en temps T_h par le système hybride, on obtient de façon similaire :

$$\forall Y_{h,0} \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n), \exists T_h \geq 0, \inf_{X \in \mathcal{C}_{NL}(\mathbb{R}^n)} \|X - Y_0\| \leq \frac{\varepsilon(h)}{L}(e^{LT_h} - 1) \quad (4.10)$$

Le temps T n'étant a priori pas borné, nous sommes amenés à définir la notion d'ensemble de points contrôlables en un temps inférieur ou égal à T , noté $\mathcal{C}_{NL,[0,T]}(\mathbb{R}^n)$ dans le cas non linéaire et $\mathcal{C}_{\mathcal{H},[0,T]}(\mathbb{R}^n)$ dans le cas hybride. Ceci nous permet donc de conclure :

Proposition 4.2.2. *Soit f un champ non linéaire supposé globalement lipschitzien par rapport à X et uniformément par rapport à u . Alors :*

$$d_H(\mathcal{C}_{NL,[0,T]}(\mathbb{R}^n), \mathcal{C}_{\mathcal{H},[0,T]}(\mathbb{R}^n)) \leq \varepsilon(h) \frac{e^{LT} - 1}{L},$$

sachant que l'erreur d'interpolation converge vers 0 quand le pas h tend vers 0 : $\lim_{h \rightarrow 0} \varepsilon(h) = 0$.

Ce résultat se généralise de façon immédiate au cas où le champ f est lipschitzien sur tout compact par rapport à X , uniformément par rapport à u , en remarquant que le domaine contrôlable en temps inférieur ou égal à T est également l'ensemble atteignable à partir de 0 en temps au plus égal à T :

Proposition 4.2.3. *Soit f un champ non linéaire supposé lipschitzien sur tout compact par rapport à X , uniformément par rapport à u . Alors, sous les hypothèses de la proposition 2.3.2, on a pour tout $T > 0$:*

$$\exists L_{\Omega} > 0, d_H(\mathcal{C}_{NL,[0,T]}(\mathbb{R}^n), \mathcal{C}_{\mathcal{H},[0,T]}(\mathbb{R}^n)) \leq \varepsilon_{\Omega}(h) \frac{e^{L_{\Omega}T} - 1}{L_{\Omega}},$$

où Ω est le compact formé par l'union de $\mathcal{C}_{NL,[0,T]}(\mathbb{R}^n)$ et $\mathcal{C}_{\mathcal{H},[0,T]}(\mathbb{R}^n)$.

Dans ce paragraphe, nous avons donc prouvé que pour tout point X_0 contrôlable vis-à-vis du modèle hybride \mathcal{H} , il existe un point contrôlable vis-à-vis du système non linéaire initial qui soit proche de X_0 . Cette proximité est quantifiée par les propositions 4.2.2 et 4.2.3.

De plus, le procédé de construction de points contrôlables particuliers grâce à l’astuce (4.8) nous permet d’obtenir le résultat suivant :

Proposition 4.2.4. *Soit X_0 un point contrôlable en temps T selon le contrôle u pour le système $\dot{X} = f(X, u)$, où f est supposée lipschitzienne en X , uniformément par rapport à u .*

Alors, quel que soit $h > 0$, il existe un point $Y_{h,0}$ de l’espace d’état tel que $Y_{h,0}$ est contrôlable en temps T par le système hybride \mathcal{H}_h pour le même contrôle u et :

$$\forall h > 0, \|X_0 - Y_{h,0}\| \leq \varepsilon(h) \frac{e^{LT} - 1}{L} \quad \text{et} \quad \lim_{h \rightarrow 0^+} Y_{h,0} = X_0$$

où les systèmes \mathcal{H}_h sont des approximations hybrides affines par morceaux du système non linéaire considéré sur un maillage de pas h .

4.2.3 Deux approches de la contrôlabilité d’un point initial donné

Soit X_0 un point donné de l’espace d’état dont on veut déterminer la contrôlabilité ; prouver la contrôlabilité d’un point donné de l’espace d’état revient à tester son appartenance au domaine contrôlable du système considéré. Deux approches peuvent alors être envisagées :

Une approche globale Le point X_0 est contrôlable de façon approchée vis-à-vis du système non linéaire s’il appartient au domaine contrôlable hybride. La question est donc d’arriver à tester l’appartenance d’un point X_0 donné au domaine contrôlable hybride.

Malheureusement, le domaine contrôlable d’un système hybride de même que les ensembles atteignables, est en pratique rarement calculable. Entre autres raisons, [44, chapitre 7] évoque le problème du nombre non borné de transitions nécessaires pour atteindre certains états du système. En effet, le principe du calcul du domaine contrôlable hybride serait parcourir, à partir de la cible 0, l’ensemble des cellules atteignables par le système hybride par renversement du temps. Si le point X_0 est “loin de la cible” i.e. atteint après un grand nombre d’itérations, le calcul du domaine cherché ne peut se faire en temps fini puisque l’on est amené à simuler une infinité de transitions.

Ainsi, même si l’approche globale nous permet d’obtenir des résultats pertinents sur l’approximation de la contrôlabilité des systèmes non linéaires, il semble qu’une approche plus propice à la simulation est nécessaire.

Une approche constructive pour réduire l’exploration D’après la définition de la contrôlabilité hybride présentée au paragraphe 4.1.3, le point X_0 est contrôlable vis-à-vis du système hybride \mathcal{H} s’il existe un chemin fini $\gamma = (q_i)_{i=0\dots r}$ de modes adjacents de l’automate hybride \mathcal{H} et une trajectoire dans le domaine borné Ω_γ associé qui relie X_0 à la cible en temps fini. Par conséquent, il suffit d’obtenir la contrôlabilité du point X_0 dans un chemin fini de cellules pour avoir la contrôlabilité hybride de X_0 .

L’idée est la suivante : on fixe une trajectoire discrète γ de l’automate hybride dans l’ensemble \mathcal{Q} des états discrets de sorte que le chemin de cellules associé à γ dans \mathbb{R}^n contienne

à la fois la cible 0 et le point X_0 . Le point X_0 est contrôlable dans le chemin donné si nous pouvons trouver une exécution de la forme (τ, γ, X) vérifiant les conditions de la définition 4.1.3 (cf paragraphe 4.3). La difficulté est alors de déterminer au moins un chemin de cellules dans lequel X_0 est contrôlable.

Une solution intéressante serait de proposer une approche multi-échelle de la contrôlabilité du point X_0 : l'idée est d'étudier sa contrôlabilité à travers des approximations hybrides à différentes échelles i.e. pour différents pas de maillage $h > 0$.

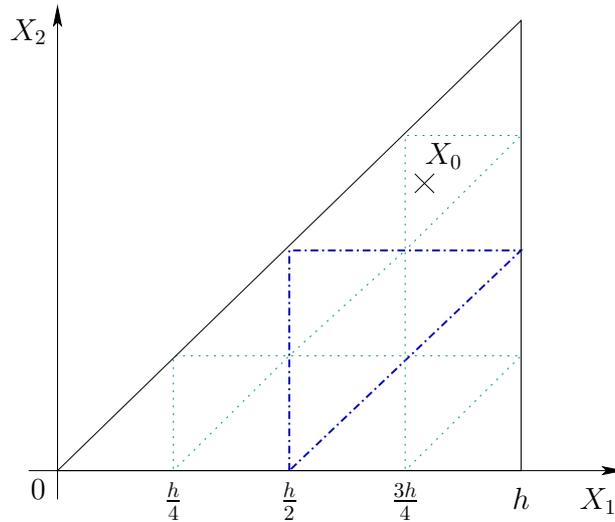


FIG. 4.3 – Maillage d'un domaine à différentes échelles : h (en traits pleins), $\frac{h}{2}$ (en tirets-pointillés) et $\frac{h}{4}$ (en pointillés)

Partant d'une échelle grossière i.e. d'un pas de maillage $h > 0$ suffisamment grand pour que le point initial et la cible soient dans une même cellule D_q de l'espace d'état, on calcule le domaine contrôlable dans le domaine considéré (une cellule D_q à la première itération). En fonction des résultats de contrôlabilité obtenus, on décide ou non de raffiner localement le maillage afin d'améliorer l'approximation. Un point clé pour valider cette approche est donc d'obtenir des relations entre les contrôlabilités du point X_0 à différentes échelles.

Le schéma de la figure 4.4 présente les deux approches de la contrôlabilité d'un point initial donné ainsi que les différents niveaux d'approximation du domaine contrôlable hybride.

À défaut de pouvoir calculer explicitement le domaine contrôlable complet d'un système hybride, nous proposons au paragraphe suivant une méthode permettant de calculer le domaine contrôlable sur des domaines bornés de l'espace d'état par le biais du calcul de l'ensemble atteignable à partir de la cible par renversement du temps.

Approche globale

$$\begin{array}{ccc} X_0 \text{ appartient au} & & X_0 \text{ appartient au} \\ \text{domaine contrôlable non linéaire} & \sim & \text{domaine contrôlable hybride} \\ X_0 \in \mathcal{C}_{NL}(\mathbb{R}^n) & & X_0 \in \mathcal{C}_{\mathcal{H}}(\mathbb{R}^n) \end{array}$$

Approche constructive : réduction de l'exploration (paragraphe 4.3)

$$\begin{array}{ccc} X_0 \text{ appartient au} & & \text{Il existe un ensemble } \Omega_\gamma \text{ de cellules} \\ \text{domaine contrôlable non linéaire} & \sim & \text{tel que } X_0 \text{ soit contrôlable dans } \Omega_\gamma : \\ X_0 \in \mathcal{C}_{NL}(\mathbb{R}^n) & & X_0 \in \mathcal{C}_{\mathcal{H}}(\Omega_\gamma) \\ & & \cup \\ & & \text{Il existe un chemin ordonné } \gamma \text{ de cellules} \\ & & \text{le long duquel } X_0 \text{ est contrôlable} \\ & & X_0 \in \mathcal{C}_{\mathcal{H}}(\Omega_\gamma) = \bigcup_{q \in \gamma} \mathcal{C}_{\mathcal{H}}(\tau_q, D_q) \\ & & \text{(Paragraphe 4.3.2)} \end{array}$$

FIG. 4.4 – Différents niveaux d'approximation de la contrôlabilité d'un point donné par les systèmes hybrides

4.3 Réduction de l'exploration : domaine contrôlable sur un ensemble de cellules

L'idée directrice de cette approche a été inspirée par l'étude de la contrôlabilité d'un point donné présentée au paragraphe 4.2.3. Un point X_0 donné de l'espace d'état est contrôlable de façon approchée par le système non linéaire (4.1) s'il existe un chemin $\gamma = (q_i)_{i=0\dots r}$ de modes discrets du modèle hybride et une trajectoire contenue dans le domaine $\Omega_\gamma = \bigcup_{i=0}^r D_{q_i}$ associé au chemin, qui relie X_0 à la cible 0 en temps fini. Ainsi, en supposant le chemin γ connu (cf paragraphe 4.2.3), il s'agit de savoir calculer le domaine contrôlable dans le domaine Ω_γ i.e. l'ensemble des points de Ω_γ qui peuvent atteindre la cible sans sortir de Ω_γ .

Dans ce paragraphe, nous considérons donc une suite $\gamma = (q_i)_{i=0\dots r}$ finie d'états discrets de l'automate \mathcal{H} telle que :

$$\begin{cases} 0 \in D_{q_0} \\ D_{q_i} \cap D_{q_{i+1}} \neq \emptyset, i = 0, \dots, r-1 \end{cases}$$

Par définition, $(D_{q_i})_{i=0\dots r}$ est donc une suite de cellules adjacentes de l'espace d'état.

Nous introduisons alors le domaine Ω_γ associé au chemin γ : c'est l'union finie des cellules D_{q_i} , $i = 0, \dots, r$, associées au chemin γ :

$$\Omega_\gamma = \bigcup_{i=0}^r D_{q_i}.$$

Par construction, Ω_γ est un ensemble compact de \mathbb{R}^n contenant l'origine et l'adjacence des cellules D_{q_i} du chemin γ implique également la connexité du domaine Ω_γ (voir figure 4.5).

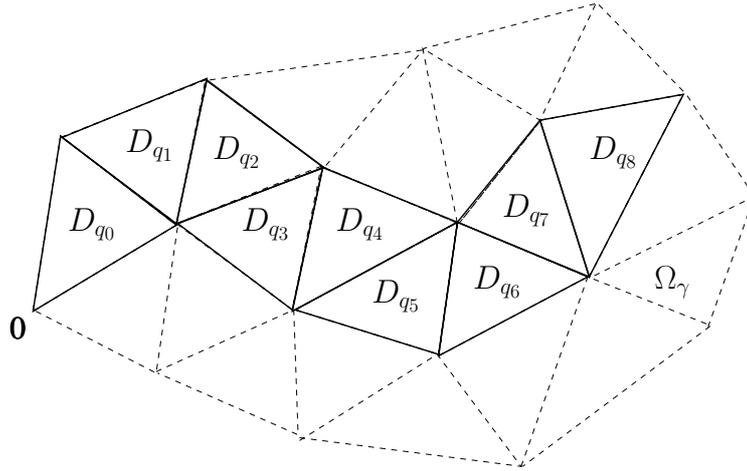


FIG. 4.5 – Définition du domaine Ω_γ associé à un chemin fini de cellules adjacentes contenant la cible 0.

Nous nous intéressons au domaine contrôlable défini dans Ω_γ . Commençons donc par introduire la notion de domaine contrôlable hybride dans une cellule D_q de l'espace d'état.

4.3.1 Domaine contrôlable à l'échelle d'une cellule

Considérons un mode $q \in \mathcal{Q}$ donné de l'automate hybride et τ_q une cible locale contenue dans la cellule D_q . Nous définissons maintenant une notion plus faible du domaine contrôlable dans un mode discret donné de l'automate :

Définition 4.3.1 (Contrôlabilité dans un mode q). Soit $q \in \mathcal{Q}$ un mode donné de l'automate hybride \mathcal{H} et τ_q un sous-ensemble de la cellule D_q .

Un point $X_0 \in D_q$ est contrôlable jusqu'à la cible τ_q , si et seulement s'il existe un temps $T \geq 0$ et un contrôle admissible u tels que le problème :

$$\begin{cases} \dot{X}_h(t) = f_h(X_h(t), u(t)) \\ X(0) = X_0 \end{cases}, \quad X_h(T) \in \tau_q$$

admette une solution X_h , vérifiant : $\forall t \in [0, T]$, $X_h(t) \in D_q$. On note $\mathcal{C}_{\mathcal{H}}(\tau_q, D_q)$ le domaine contrôlable du système dans la cellule D_q relativement à la cible τ_q .

Le domaine contrôlable $\mathcal{C}_{\mathcal{H}}(\tau_q, D_q)$ est donc l'ensemble des points de D_q , à partir desquels il est possible d'atteindre la cible τ_q en temps fini, sans sortir de la cellule D_q . On a alors le résultat suivant :

Proposition 4.3.1. $\tau_q \subset \tau_{q'} \Rightarrow \mathcal{C}_{\mathcal{H}}(\tau_q, D_q) \subset \mathcal{C}_{\mathcal{H}}(\tau_{q'}, D_q)$.

En effet, tout point de D_q à partir duquel on peut atteindre τ_q sans sortir de la cellule D_q , atteint par défaut $\tau_{q'}$ sans sortir de D_q .

4.3.2 Propagation le long d'un chemin de cellules

La définition 4.3.1 du domaine contrôlable dans une cellule D_q se généralise sans problème à toute partie de l'espace d'état et donc en particulier au domaine Ω_γ défini précédemment :

Définition 4.3.2 (Contrôlabilité dans le domaine Ω_γ). *Un point $X_0 \in \Omega_\gamma$ est contrôlable à l'origine par le système $\dot{X}(t) = f_h(X(t), u(t))$ dans le domaine Ω_γ si et seulement si il existe un temps $T \geq 0$ et un contrôle u admissible tels que le problème :*

$$\begin{cases} \dot{X}(t) &= f_h(X(t), u(t)) \\ X(0) &= X_0 \end{cases}, \quad X(T) = 0$$

admette une solution X , vérifiant : $\forall t \in [0, T], X(t) \in \Omega_\gamma$. On note $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$ le domaine contrôlable hybride dans le domaine Ω_γ .

En d'autres termes, un point X_0 est contrôlable jusqu'à la cible 0 dans le domaine Ω_γ s'il existe une trajectoire admissible du système qui relie X_0 à 0 en temps fini et sans sortir de Ω_γ .

Nous pouvons alors proposer une approximation du domaine contrôlable d'un système de contrôle quelconque dans Ω_γ , définie récursivement le long du chemin γ . L'astuce consiste à commencer par la cellule D_{q_0} qui contient la cible 0 : on calcule tout d'abord le domaine contrôlable $\mathcal{C}_{\mathcal{H}}(\{0\}, D_{q_0})$ i.e. l'ensemble des points contrôlables jusqu'en 0 sans sortir de D_{q_0} . L'intersection de ce domaine avec la garde $G_{(q_0, q_1)}$, si elle est non vide, définit une nouvelle cible à atteindre à partir, cette fois-ci, de la cellule D_{q_1} . On réitère le processus jusqu'à avoir parcouru les $r + 1$ cellules qui composent Ω_γ .

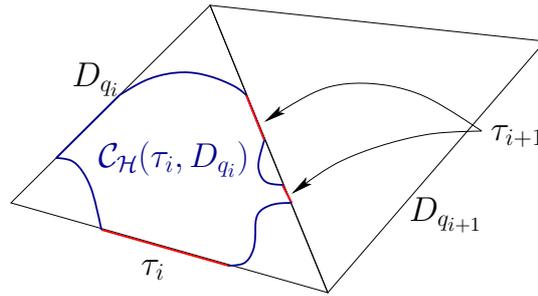


FIG. 4.6 – Propagation du domaine contrôlable le long d'un chemin de cellules

Lemme 4.3.1. *Pour tout $i \in \{0, \dots, r-1\}$, on note $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i} \cup D_{q_{i+1}})$ l'ensemble des points contrôlables jusqu'à la cible τ_i sans sortir de $D_{q_i} \cup D_{q_{i+1}}$. On a alors :*

$$\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cup \mathcal{C}_{\mathcal{H}}(\tau_{i+1}, D_{q_{i+1}}) \subset \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i} \cup D_{q_{i+1}}),$$

en posant : $\tau_{i+1} = \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}$.

Preuve. Soit $X_0 \in \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cup \mathcal{C}_{\mathcal{H}}(\tau_{i+1}, D_{q_{i+1}})$. Deux cas se présentent alors :

- $X_0 \in \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$: le point X_0 est donc contrôlable jusqu'à la cible τ_i sans sortir de $D_{q_i} \subset D_{q_i} \cup D_{q_{i+1}}$ et donc de $D_{q_i} \cup D_{q_{i+1}}$, soit :

$$X_0 \in \mathcal{C}_{\mathcal{H}}(\tau_{i+1}, D_{q_{i+1}}).$$

- $X_0 \in \mathcal{C}_{\mathcal{H}}(\tau_{i+1}, D_{q_{i+1}})$: il existe donc un point $Y_0 \in \tau_{i+1}$ tel que le système hybride \mathcal{H} relie X_0 à Y_0 en temps fini sans sortir de $D_{q_{i+1}}$. Or, par définition de τ_{i+1} , Y_0 est un point de $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$, ce qui signifie qu'il existe une trajectoire du système \mathcal{H} qui relie Y_0 à τ_i en temps fini et sans sortir de D_{q_i} .

Par transitivité, il existe donc une trajectoire de \mathcal{H} dans $D_{q_i} \cup D_{q_{i+1}}$ qui relie X_0 à τ_i en temps fini (et qui passe par Y_0).

□

Le domaine contrôlable dans Ω_γ est donc approché cellules après cellules de la façon suivante :

Proposition 4.3.2. *On pose :*

$$\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma) = \mathcal{C}_{\mathcal{H}}(\tau_0, D_{q_0}) \cup \mathcal{C}_{\mathcal{H}}(\tau_1, D_{q_1}) \cup \dots \cup \mathcal{C}_{\mathcal{H}}(\tau_r, D_{q_r}),$$

où $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$ désigne l'ensemble des points de D_{q_i} à partir desquels l'ensemble cible $\tau_i \subset D_{q_i}$ est accessible en temps fini tout en restant dans la cellule D_{q_i} . La cible courante τ_i est défini récursivement de la façon suivante :

$$\begin{cases} \tau_0 = \{0\} \\ \tau_{i+1} = \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}, \quad i = 0, \dots, r-1 \end{cases}$$

$\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma)$ est une sous-approximation de $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$, soit : $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma) \subset \mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$.

Remarque 4.3.1. *Dans la construction du domaine contrôlable $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma)$, nous avons imposé un ordre de parcours des cellules du chemin γ considéré. De ce fait, nous excluons entre autres les trajectoires qui rebouclent dans une même cellule (cf figure 4.7). C'est pourquoi le domaine $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma)$ est une sous-approximation du domaine contrôlable dans Ω_γ .*

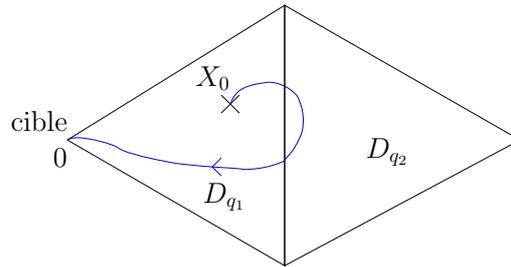


FIG. 4.7 – Exemple de trajectoire exclue du sous-domaine contrôlable $\underline{\mathcal{C}}_{\mathcal{H}}(D_{q_1} \cup D_{q_2})$

Cette proposition pose les fondements de notre étude du domaine contrôlable hybride : derrière cette décomposition en sous-domaines du domaine contrôlable dans Ω_γ se dessine l'algorithme de construction du domaine contrôlable hybride dans un chemin de cellules présenté au chapitre 5.

4.3.3 Propagation du domaine contrôlable en p itérations

Une autre approche du calcul du domaine contrôlable peut être envisagée avec les outils définis au paragraphe précédent. En effet, partant de la cellule contenant la cible choisie, l'idée est de propager le domaine contrôlable par renversement du temps en un nombre fixé d'itérations (cf figure 4.8).

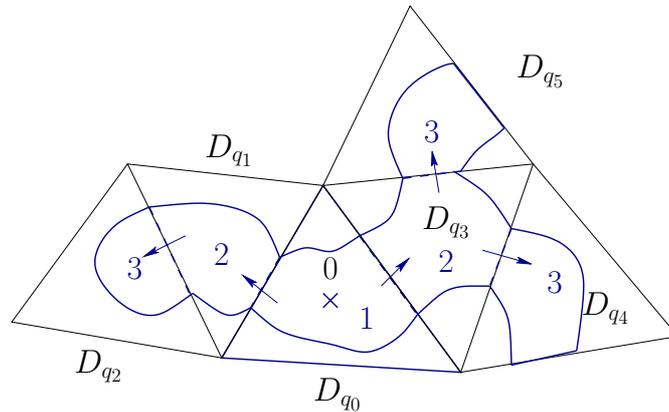


FIG. 4.8 – Propagation du domaine contrôlable à partir de la cellule D_{q_0} sur trois itérations

Le principe est le suivant : partant de la cellule initiale contenant la cible, on calcule le domaine contrôlable dans cette cellule puis les intersections de ce domaine avec les faces de la cellule. Chaque intersection définit une cible pour la prochaine itération et une cellule successeur vers laquelle on propage le domaine contrôlable selon le processus décrit au paragraphe précédent. Cette opération est répétée à l'itération p à partir des domaines contrôlables calculés à l'itération $(p - 1)$.

Dans ce chapitre, nous avons étudié l'approximation du domaine contrôlable non linéaire par le domaine contrôlable hybride. Grâce à la notion de domaine contrôlable dans un chemin de cellules données, nous avons également proposé une approche constructive du calcul du domaine contrôlable hybride permettant de réduire l'exploration de l'espace d'état et développé une méthode de calcul de ce domaine par propagation. Le prochain chapitre est consacré au calcul effectif du domaine contrôlable hybride sur des chemins de cellules.

Chapitre 5

Approximation convexe du domaine contrôlable hybride

Nous nous intéressons dans ce chapitre au calcul du domaine contrôlable des systèmes hybrides i.e. au calcul de l'ensemble des points de l'espace d'état pour lesquels il existe une trajectoire admissible du système qui les amène à la cible 0 en temps fini. La classe de systèmes considérée est celle des systèmes hybrides affines par morceaux définis au chapitre 3.

Dans sa généralité le problème de la contrôlabilité/atteignabilité des systèmes hybrides est indécidable. Certaines sous-classes de systèmes hybrides pour lesquels ce problème est décidable, ont toutefois pu être identifiées mais le calcul du domaine contrôlable reste difficile.

Une première classe de méthodes d'approximation des ensembles atteignables consiste à extraire par abstraction un modèle simplifié qui lui est équivalent vis-à-vis de la propriété à vérifier [2] : on citera entre autres les méthodes d'abstraction par projection [3] ou encore celles d'abstraction pour les systèmes hybrides [79]. Par exemple pour la vérification des systèmes hybrides, l'enjeu est de construire un modèle pour lequel le calcul d'atteignabilité est le plus simple possible et dont l'ensemble atteignable est une sur-approximation de celui du modèle initial. La difficulté de cette approche consiste à trouver une bonne abstraction qui permette de conclure avec une complexité raisonnable.

Une autre classe de méthodes a été développée dans le cadre de la vérification des systèmes des systèmes continus et hybrides et est basée sur l'approximation de l'ensemble atteignable par différents types d'ensembles tels que l'union de polytopes ou d'hyperrectangles orthogonaux [27, 4, 44]. Par exemple, pour la vérification d'une propriété de sûreté, [4] propose un algorithme qui calcule une sur-approximation de l'ensemble atteignable certifiant ainsi que le système ne peut s'échapper d'un ensemble admissible d'états.

Citons également une approche propre aux systèmes hybrides proposée dans [82, 47] : le problème de contrôlabilité est décomposé en deux problèmes complémentaires : un problème d'atteignabilité d'un automate à événements discrets et un ensemble fini de problèmes d'atteignabilité pour des systèmes affines définis sur des polytopes de l'espace d'état.

L'approche développée dans ce chapitre appartient à la deuxième classe de méthodes : partant de la cible, on cherche théoriquement à calculer l'ensemble des états atteignables

du système hybride \mathcal{H} par renversement du temps. Dans la continuité de l'étude menée au chapitre 4 et dans un souci de réduire l'exploration, nous nous intéressons ici au calcul du domaine contrôlable hybride dans une suite donnée de modes discrets de l'automate \mathcal{H} .

Contrairement à ce qui est usuellement proposé en vérification, notre idée est de garantir la contrôlabilité d'un point donné et donc de proposer un algorithme de sous-approximation du domaine contrôlable. Dans [35] nous avons présenté des techniques efficaces de sous-approximation du domaine contrôlable des systèmes linéaires $\dot{X} = AX + Bu$ basées sur la convexité de ce domaine et sur le calcul d'ensembles atteignables en temps fini. Dans [73], nous généralisons ces techniques aux systèmes hybrides polyédraux affines dans chaque mode. Dans ce chapitre, nous voulons adapter ces techniques aux systèmes hybrides construits au chapitre 3.

Ce chapitre est consacré à l'étude algorithmique du domaine contrôlable hybride défini sur des chemins donnés d'états discrets. Nous commençons donc par la mise en place d'outils pour la simulation des systèmes hybrides contrôlés construits au chapitre 3 indispensables dans la perspective du calcul du domaine contrôlable. Après avoir présenté des résultats de convexité du domaine contrôlable hybride dans une cellule, nous proposons un algorithme de calcul de ce domaine contrôlable hybride. Nous concluons ce chapitre par une étude de l'erreur d'approximation.

5.1 Adaptation des outils de simulation des systèmes hybrides au contrôle

La simulation d'un système hybride \mathcal{H} consiste à calculer une exécution (τ, q, X) acceptée par \mathcal{H} , connaissant une condition initiale (t_0, q_0, X_0) . Un algorithme de simulation a été proposé par A. Girard dans le contexte des automates hybrides linéaires par morceaux autonomes [44, chapitre 4, Simulation des systèmes hybrides]. Chaque itération de cet algorithme se décompose en trois phases :

1. Simulation de l'évolution continue, i.e. le calcul des trajectoires continues du système dans un état discret donné.
2. Détection des événements, i.e. le calcul du temps de sortie d'un mode $q \in \mathcal{Q}$ de l'automate hybride.
3. Simulation de l'évolution discrète, i.e. le calcul des transitions discrètes $e \in \mathcal{E}$ effectuées par le système.

Le principe de cet algorithme est le même que l'on considère des systèmes hybrides polyédraux affines (cf 1.3.2) ou bien des systèmes de la classe de ceux construits au chapitre 3. Cependant, la mise en oeuvre de l'étape 3. de l'algorithme de simulation s'avère très complexe dans le contexte des systèmes hybrides considérés dans ce manuscrit. En effet, dans chaque mode q du système hybride, il nous faut calculer la cellule de contrôle appropriée afin de pouvoir se ramener localement à une dynamique affine. En raison du manque de régularité de la fonction de contrôle, ceci n'est pas toujours possible.

C'est pourquoi nous proposons une adaptation des travaux d'A. Girard à la simulation des systèmes hybrides définis au chapitre 3. Il s'agit donc dans un premier temps de résoudre

symboliquement les équations de la dynamique hybride dans un état discret du système hybride \mathcal{H} . Puis nous nous intéressons aux conditions d'entrée d'une trajectoire donnée dans une cellule $\Delta_{q'}$ de l'espace état-contrôle, ce qui nous permet enfin de construire un algorithme qui teste les conditions d'entrée dans un mode q de l'automate \mathcal{H} . Pour terminer, nous évoquons succinctement le problème du calcul des temps de sortie d'un mode q dont nous rappelons les difficultés et les solutions existantes.

Nous nous intéressons tout particulièrement dans cette partie à la classe des contrôles affinement dépendants (éventuellement par morceaux) de la position du système i.e. de la forme :

$$u = FX + g.$$

Nous nous apercevons en effet dans la suite de ce manuscrit que ces contrôles jouent un rôle prépondérant non seulement pour le calcul du domaine contrôlable hybride (cf paragraphe 5.3.1), mais aussi pour la résolution des problèmes de contrôle optimal traités dans la partie III (cf paragraphe 7.1.1).

5.1.1 Calcul des trajectoires continues pour un contrôle $u = FX + g$

Dans ce paragraphe, nous nous intéressons au calcul des trajectoires admissibles de l'automate hybride \mathcal{H} pour une condition initiale $X(t_0) = X_0$ et un contrôle u donnés.

Soit q le mode discret dans lequel se trouve le système à l'instant initial t_0 , i.e. tel que le système reste dans la cellule D_q dans un intervalle de temps non réduit à $\{t_0\}$:

$$\exists \varepsilon > 0, \forall t \in [t_0, t_0 + \varepsilon[, X(t) \in D_q \quad (5.1)$$

Le problème est alors de calculer dans la cellule D_q , la trajectoire issue du point X_0 à l'instant t_0 selon le contrôle donné u , ce qui revient à résoudre le problème de Cauchy suivant :

$$\begin{cases} \dot{X}(t) = f_q(X(t), u(t)) \\ X(t_0) = X_0 \end{cases} \quad \text{tant que : } X(t) \in D_q, \quad (5.2)$$

sachant que la fonction $f_q : D_q \times \mathbb{U}_m \rightarrow \mathbb{R}^n$ est affine par morceaux. D'après l'étude menée au chapitre 2, la fonction de contrôle u est mesurable, non continue dans le cas général. Dans le cas général donc, la résolution formelle du problème (5.2) n'est pas toujours possible.

En revanche, considérons des contrôles affinement dépendants de la position du système au cours du temps de la forme :

$$\forall t, u(t) = FX(t) + g.$$

De tels contrôles possèdent la même régularité que la trajectoire X . Par conséquent, la trajectoire état-contrôle (X, u) est continue par rapport au temps. Elle évolue de plus par définition dans le sous-espace affine de \mathbb{R}^{n+m} d'équation $(u = FX + g)$ (voir figure 5.1).

Nous introduisons maintenant la cellule $\Delta_{q'}$, $q' \in \mathcal{K}(q)$, de la colonne des cellules état-contrôle possibles dans le mode q , vérifiant :

$$\exists \varepsilon' > 0, \forall t \in [t_0, t_0 + \varepsilon'], (X(t), FX(t) + g) \in \Delta_{q'}. \quad (5.3)$$

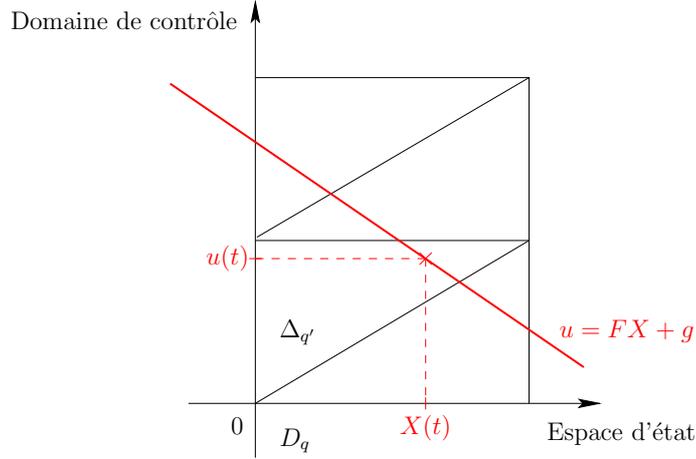


FIG. 5.1 – Sous-espace affine ($u = FX + g$) support de la trajectoire état-contrôle dans $\mathbb{R}^n \times \mathbb{U}_m$

Par analogie avec (5.1), la condition (5.3) signifie que la trajectoire (X, u) reste dans $\Delta_{q'}$ sur un intervalle de temps non réduit à $\{t_0\}$. Ainsi, dans cette cellule, la dynamique du système hybride \mathcal{H} est régie par le système différentiel affine suivant :

$$\begin{cases} \dot{X}(t) &= A_{q'}X(t) + B_{q'}u(t) + c_{q'} \\ X(t_0) &= X_0 \end{cases} \quad \text{soit :} \quad \begin{cases} \dot{X}(t) &= (A_{q'} + B_{q'}F)X(t) + (B_{q'}g + c_{q'}) \\ X(t_0) &= X_0 \end{cases}$$

dont la solution peut être calculée de manière formelle sur l'intervalle de temps $[t_0, t_0 + \epsilon']$:

$$X(t) = e^{(A_{q'} + B_{q'}F)(t-t_0)} X(t_0) + e^{(A_{q'} + B_{q'}F)t} \int_{t_0}^t e^{-(A_{q'} + B_{q'}F)s} (B_{q'}g + c_{q'}) ds.$$

En pratique, la résolution d'un système d'équations différentielles ordinaires comprend une première étape de triangularisation du système (soit $\mathcal{O}(n^3)$ opérations par une méthode de décomposition de Gauss), puis une seconde étape de résolution de n équations différentielles ordinaires à une variable. Nous en déduisons donc :

Proposition 5.1.1. *Avec les notations précédentes, le calcul des trajectoires continues pour un contrôle $u = FX + g$ dans une cellule $\Delta_{q'}$ s'effectue en $\mathcal{O}(n^3)$ opérations arithmétiques.*

Remarque 5.1.1. *Pour une évaluation numérique plus simple, une astuce consiste à poser le changement de variable suivant :*

$$Y(t) = \begin{bmatrix} X(t) \\ 1 \end{bmatrix}.$$

(idée donnée dans [44, paragraphe 4.1] pour les systèmes linéaires autonomes). Ce changement de variable nous permet d'obtenir un nouveau système :

$$\begin{cases} \dot{Y}(t) &= \begin{bmatrix} A_{q'} + B_{q'}F & B_{q'}g + c_{q'} \\ 0 & 0 \end{bmatrix} Y(t) \\ Y(t_0) &= \begin{bmatrix} X_0 \\ 1 \end{bmatrix} \end{cases},$$

dont la solution est : $Y(t) = e^{\bar{A}(t-t_0)}Y(t_0)$ en posant : $\bar{A} = \begin{bmatrix} A_{q'} + B_{q'}F & B_{q'}g + c_{q'} \\ 0 & 0 \end{bmatrix}$.

5.1.2 Conditions d'entrée dans la cellule $\Delta_{q'}$

Considérons maintenant un mode q donné de l'automate \mathcal{H} et une cellule $\Delta_{q'}$ quelconque de la colonne de cellules état-contrôle possibles dans le mode q (i.e. : $q' \in \mathcal{K}(q)$). On suppose également donné un contrôle $u = FX + g$ admissible dans le mode q i.e. vérifiant :

$$\forall X \in D_q, FX + g \in \mathbb{U}_m.$$

Dans ce paragraphe, nous nous intéressons à la trajectoire état-contrôle, notée $Z = (X, u)$, définie comme suit :

$$Z(t) = \begin{bmatrix} X(t) \\ u(t) \end{bmatrix} = \begin{bmatrix} X(t) \\ FX(t) + g \end{bmatrix} = \begin{bmatrix} I_n \\ F \end{bmatrix} X(t) + \begin{bmatrix} 0 \\ g \end{bmatrix},$$

et calculée dans la cellule $\Delta_{q'}$ comme solution du problème de Cauchy suivant :

$$\begin{cases} \dot{Z}(t) &= \begin{bmatrix} I_n \\ F \end{bmatrix} (A_{q'} + B_{q'}F)X(t) + \begin{bmatrix} I_n \\ F \end{bmatrix} (B_{q'}g + c_{q'}) \\ Z(t_0) &= \begin{bmatrix} I_n \\ F \end{bmatrix} X_0 + \begin{bmatrix} 0 \\ g \end{bmatrix} \end{cases} \quad (5.4)$$

Sous ces conditions, le problème est alors de construire un critère permettant de tester pour une condition initiale donnée si la trajectoire état-contrôle $Z = (X, u)$ calculée dans la cellule $\Delta_{q'}$ quitte ou non $\Delta_{q'}$ à l'instant initial t_0 .

Ce problème se formule de la façon suivante : existe-t-il un intervalle de temps $[t_0, t_0 + T[$ non réduit à $\{t_0\}$ sur lequel la trajectoire Z reste dans la cellule $\Delta_{q'}$ i.e. a-t-on :

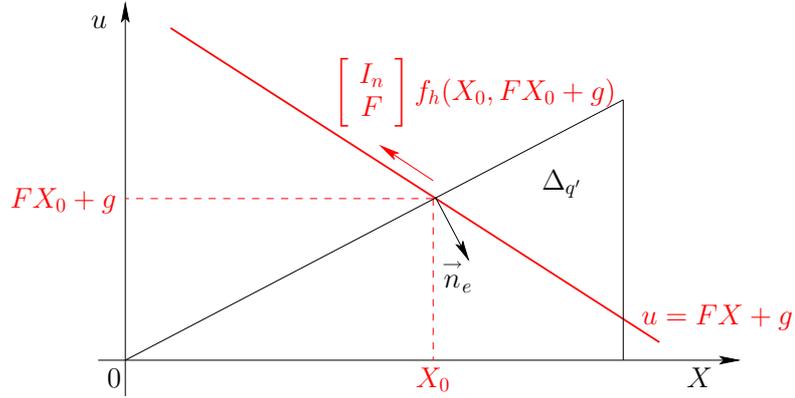
$$\exists T > 0, \forall t \in [t_0, t_0 + T[, Z(t) \in \Delta_{q'} ? \quad (5.5)$$

Si le point initial $Z(t_0)$ appartient à l'intérieur de $\Delta_{q'}$, d'après la continuité des trajectoires solutions du problème de Cauchy (5.4), la condition (5.5) est nécessairement satisfaite. Si, en revanche, $Z(t_0)$ appartient à la frontière de $\Delta_{q'}$, nous voulons construire un algorithme qui nous permette de déterminer **sans la calculer** si la trajectoire $Z = (X, u)$ issue du point $Z(t_0) = (X_0, FX_0 + g)$ se dirige ou non vers l'intérieur de $\Delta_{q'}$.

Dans la suite, nous proposons une adaptation de l'algorithme présenté dans [44, §4.2 Détection des événements] aux systèmes hybrides linéaires par morceaux autonomes.

Conditions d'entrée dans la cellule $\Delta_{q'}$

On introduit l'ensemble $E_{q'}$ des faces de la cellule $\Delta_{q'}$. A chaque face $e \in E_{q'}$ de $\Delta_{q'}$, on associe le vecteur normal à la face e , dirigé vers l'intérieur de $\Delta_{q'}$ et noté \vec{n}_e (cf figure 5.2).

FIG. 5.2 – Condition d'entrée dans une cellule $\Delta_{q'}$ de l'espace état-contrôle

On définit alors l'application :

$$g_e(t) = \langle Z(t) - Z_e, \vec{n}_e \rangle = \left\langle \begin{bmatrix} I_n \\ F \end{bmatrix} X(t) + \begin{bmatrix} 0 \\ g \end{bmatrix} - Z_e, \vec{n}_e \right\rangle,$$

où $\langle \cdot, \cdot \rangle$ désigne le produit scalaire euclidien usuel en dimension $n + m$ et Z_e un point quelconque de la face considérée (on peut par exemple prendre un sommet de la face e).

Remarque 5.1.2. Avec ces notations, l'appartenance d'un point $Y \in \mathbb{R}^{n+m}$ au polytope $\Delta_{q'}$ est caractérisée par les relations suivantes :

$$\forall e \in E_{q'}, \langle Y - Z_e, \vec{n}_e \rangle \geq 0.$$

L'appartenance du point $Z(t_0) = (X_0, FX_0 + g)$ à la frontière de $\Delta_{q'}$ se traduit ainsi par la relation :

$$\exists e \in E_{q'}, g_e(t_0) = 0.$$

Le problème (5.5) se reformule alors de la façon suivante :

$$\text{Existe-t-il } T > 0 \text{ tel que : } \forall e \in E_{q'}, \forall t \in [t_0, t_0 + T[, g_e(t) \geq 0 ?$$

Dans le contexte des systèmes hybrides linéaires par morceaux autonomes, [44, chapitre 4 : Simulation des systèmes hybrides] énonce et démontre le résultat suivant, applicable tel quel dans notre contexte :

Proposition 5.1.2 ([44, prop 4.2.1]). Soit $e \in E_{q'}$, telle que : $g_e(t_0) = 0$.

1. Si : $\forall k \in \{1, \dots, n\}, g_e^{(k)}(t_0) = 0$, alors : $\forall t \in \mathbb{R}, g_e(t) = 0$.
2. Si : $\exists K \in \{1, \dots, n\}, g_e^{(K)}(t_0) < 0$ et $\forall k \in \{1, \dots, K-1\}, g_e^{(k)}(t_0) = 0$, alors :

$$\exists \varepsilon > 0, \forall t \in]t_0, t_0 + \varepsilon[, g_e(t) < 0$$

3. Si : $\exists K \in \{1, \dots, n\}, g_e^{(K)}(t_0) > 0$ et $\forall k \in \{1, \dots, K-1\}, g_e^{(k)}(t_0) = 0$, alors :

$$\exists \varepsilon > 0, \forall t \in]t_0, t_0 + \varepsilon[, g_e(t) > 0$$

L'idée consiste à regarder la position du vecteur champ et, si nécessaire, de ses dérivées par rapport au vecteur normal \vec{n}_e (cf figure 5.2), sachant que les faces e qui importent sont celles qui sont susceptibles d'être traversées à l'instant $t = t_0$, i.e. pour lesquelles : $(X_0, FX_0 + g) \in e$. Ceci revient donc à étudier le signe des dérivées "utiles" de g_e en t_0 , sachant que :

$$\forall t \in \mathbb{R}, g_e^{(k)}(t) = \left\langle \begin{bmatrix} I_n \\ F \end{bmatrix} (A_{q'} + B_{q'}F)^{k-1} ((A_{q'} + B_{q'}F)X_0 + B_{q'}g + c_{q'}), \vec{n}_e \right\rangle .$$

D'après la proposition 5.1.2, nous pouvons conclure que :

- s'il existe une face $e \in E_{q'}$ satisfaisant les hypothèses de la première assertion, la trajectoire $Z = (X, u)$ reste dans l'hyperplan engendré par la face e , d'équation : $\langle Y - Z_e, \vec{n}_e \rangle = 0$. La contrainte imposée par la face e est donc toujours respectée.
- s'il existe une face telle que la deuxième assertion soit vérifiée, alors la trajectoire quitte la cellule $\Delta_{q'}$ à l'instant t_0 .

Ainsi, le problème (5.5) admet une solution à condition que toute face $e \in E_{q'}$ telle que : $g_e(t_0) = 0$ (i.e. $X_0 \in e$), la troisième assertion soit vérifiée.

L'avantage de cette méthode réside dans le fait qu'on ne calcule pas la trajectoire $Z = (X, u)$ mais uniquement le vecteur champ et éventuellement ses dérivées, à l'instant t_0 .

La proposition 5.1.2 nous permet alors de construire l'algorithme 4 qui teste si les conditions d'entrée dans une cellule $\Delta_{q'}$ donnée sont satisfaites ou non. Nous utilisons pour cela un algorithme auxiliaire, *TimeDerivative*, qui calcule la dérivée par rapport au temps d'une application $t \mapsto g(X(t))$ donnée, sous une contrainte différentielle du type : $\dot{X}(t) = h(X(t))$.

5.1.3 Simulation de l'évolution discrète

Jusqu'à maintenant nous avons présenté un algorithme permettant de tester pour un contrôle affinement dépendant de la position du système et une condition initiale donnés, si la trajectoire état-contrôle associée entre ou non dans une cellule $\Delta_{q'}$ de $\mathbb{R}^n \times \mathbb{U}_m$. Pour cela, nous avons supposé connus le mode q de l'automate hybride ainsi que la cellule $\Delta_{q'}$ dans lesquels se trouve le système à l'instant initial t_0 . Il reste donc à construire un algorithme qui calcule la position du système hybride \mathcal{H} à la fois dans l'espace d'état \mathbb{R}^n et dans l'espace état-contrôle $\mathbb{R}^n \times \mathbb{U}_m$.

Au chapitre 3, nous avons présenté un algorithme (cf algorithme 3) permettant de calculer la cellule D_q de l'espace d'état dans laquelle se trouve le système hybride \mathcal{H} à une position X_0 donnée. Dans le cas où X_0 se trouve à l'intersection de plusieurs cellules du maillage \mathcal{D} de \mathbb{R}^n , nous n'avions alors aucune information nous permettant de choisir l'une ou l'autre des cellules. Le problème est donc maintenant de construire un critère nous permettant de déterminer pour une condition initiale et un contrôle $u = FX + g$ donnés, la cellule D_q vers laquelle se dirige la trajectoire du système hybride \mathcal{H} .

Conditions d'entrée dans le mode q

Soit donc une condition initiale $X(t_0) = X_0$ et un contrôle affine $u = FX + g$ donnés. On introduit alors q un mode discret de l'automate \mathcal{H} tel que : $X_0 \in D_q$. Le problème est alors

Algorithme 4 CellEntry Conditions d'entrée dans une cellule $\Delta_{q'}$

Données : n dimension de l'espace d'état, \mathcal{H} automate hybride, $\Delta_{q'}$ une cellule du maillage de $\mathbb{R}^n \times \mathbb{U}_m$,

Données : X_0 un point initial et $u = FX + g$ un contrôle affine admissible.

Sortie : Teste **sans la calculer** si la trajectoire (X, u) issue du point $(X_0, FX_0 + g)$ reste dans la cellule $\Delta_{q'}$ (cf condition (5.5)).

- 1: {Initialisation :} $test := true$;
- 2: {On calcule l'approximation affine f_h de f dans $\Delta_{q'}$:}
 $[A, B, c] := HybridDynamic(n, f, \Delta_{q'})$; $f_h := (X, u) \mapsto AX + Bu + c$;
- 3: {Ensemble des faces dans la cellule $\Delta_{q'}$:} $E_{q'} := \{e; e \text{ face de } \Delta_{q'}\}$;
- 4: {Tant que les contraintes imposées par e sont respectées,}
- 5: **tant que** $test=true$ et $e \in E_{q'}$ **faire**
- 6: {On calcule un vecteur normal entrant à e :} $\vec{n}_e = NormalVector(e)$;
- 7: $g_e := X \mapsto \begin{bmatrix} I_n \\ F \end{bmatrix} X + \begin{bmatrix} 0 \\ g \end{bmatrix} - Z_e \langle \vec{n}_e \rangle$; (où Z_e est un sommet quelconque de e).
- 8: {Ordre de dérivation de g_e :} $k := 0$;
- 9: **tant que** $k \leq n$ et $g_e(X_0) = 0$ **faire**
- 10: {Calcul de la k ème dérivée en temps de $g_e(X(t))$ sachant que : $\dot{X}(t) = f_h(X(t), u(t))$:}
 $g_e := X \mapsto TimeDerivative(n, g_e(X), X \mapsto f_h(X, FX + g), X)$;
- 11: $k := k + 1$;
- 12: **fin tant que**
- 13: **si** $1 \leq k \leq n$ et $g_e(X_0) < 0$ **alors** $test := false$; **fin si**
- 14: **fin tant que**
- 15: Retourner $test$.

le suivant : existe-t-il un intervalle de temps non réduit à $\{t_0\}$ tel que la trajectoire issue de X_0 selon le contrôle u défini précédemment, reste dans la cellule D_q , soit :

$$\text{Existe-t-il } \varepsilon > 0 \text{ tel que : } \forall t \in [t_0, t_0 + \varepsilon[, X(t) \in D_q ? \quad (5.6)$$

Ce problème, analogue à celui traité dans le paragraphe précédent, peut se formuler de manière équivalente dans l'espace état-contrôle de la façon suivante :

$$\text{Existe-t-il } \varepsilon' > 0 \text{ et } q' \in \mathcal{K}(q) \text{ tels que : } \forall t \in [t_0, t_0 + \varepsilon'[, (X(t), FX(t) + g) \in \Delta_{q'} ? \quad (5.7)$$

Nous pouvons d'ores et déjà remarquer que, si le point initial $X(t_0)$ se trouve à l'intérieur de la cellule D_q , le problème (5.6) admet nécessairement une solution. Il ne reste alors qu'à déterminer la cellule état-contrôle $\Delta_{q'}$ vers laquelle se dirige la trajectoire (X, u) à partir du point $(X(t_0), FX(t_0) + g)$ selon le contrôle $u = FX + g$.

Dans tous les cas se pose donc la question du calcul de la cellule $\Delta_{q'}$, $q' \in \mathcal{K}(q)$, dans laquelle évolue la trajectoire (X, u) . Comme illustré sur la figure 5.3, deux configurations peuvent se produire :

- a. Il existe une unique cellule $\Delta_{q'}$, $q' \in \mathcal{K}(q)$, dans le mode q , telle que :

$$(X(t_0), FX(t_0) + g) \in \text{int}(\Delta_{q'}).$$

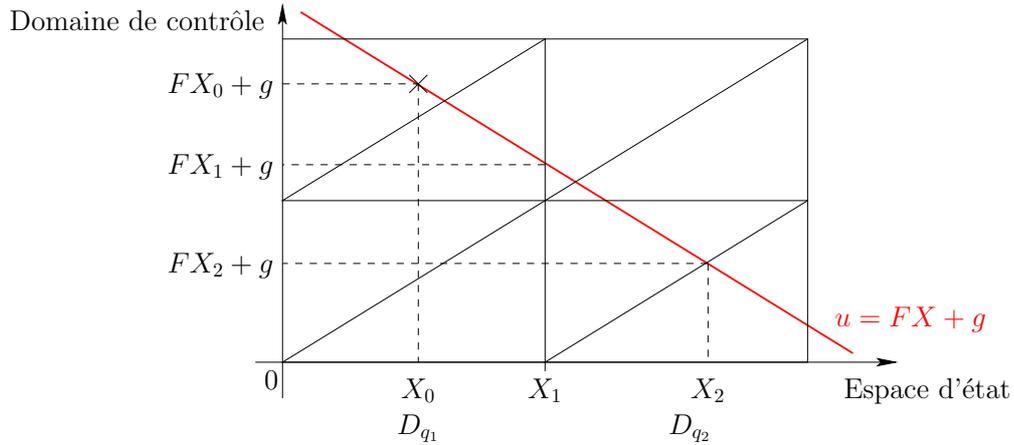


FIG. 5.3 – Position du système \mathcal{H} dans l'espace d'état et dans l'espace état-contrôle

Au quel cas les conditions d'entrée dans le mode q sont nécessairement satisfaites et $\Delta_{q'}$ est bien la cellule cherchée.

- b. Il existe plusieurs cellules $\Delta_{q'}$, $q' \in \mathcal{K}(q)$, dans le mode q qui contiennent le point $(X(t_0), FX(t_0) + g)$.

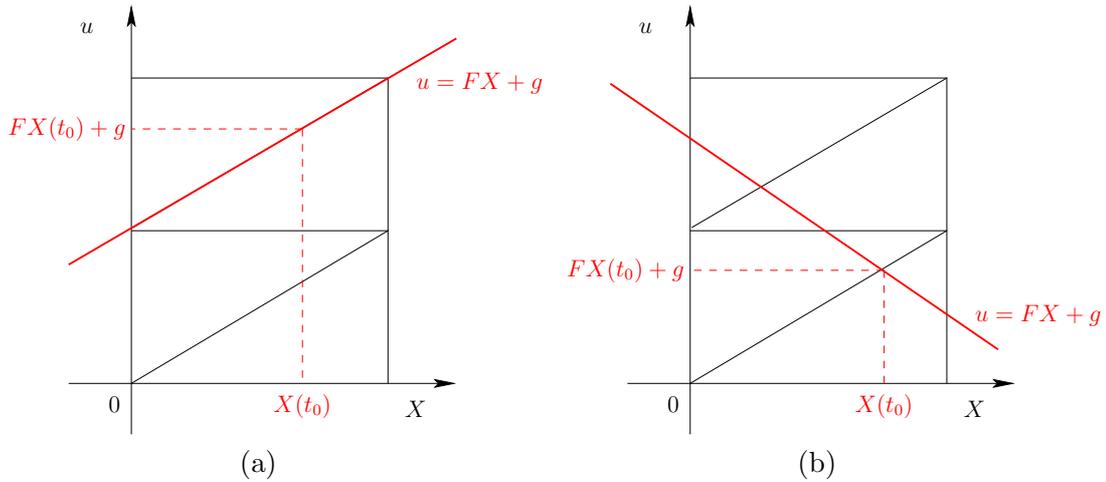


FIG. 5.4 – Choix de la cellule état-contrôle dans un mode q donné en fonction de la position du sous-espace affine ($u = FX + g$)

On note G l'intersection de ces cellules :

$$G = \bigcap_{\substack{q' \in \mathcal{K}(q) \\ (X(t_0), FX(t_0) + g) \in \Delta_{q'}}} \Delta_{q'}$$

Nous pouvons à nouveau distinguer deux configurations géométriques particulières :
 – G est incluse dans le sous-espace affine d'équation ($u = FX + g$) (cf figure 5.4-(a)) :

par continuité du champ de vecteur hybride f_h , peu importe la cellule $\Delta_{q'}$ choisie, la dynamique est la même.

- Dans le cas contraire (cf figure 5.4-(b)), nous devons déterminer vers quelle cellule se dirige la trajectoire (X, u) à l'instant t_0 : pour chaque cellule $\Delta_{q'}$, $q' \in \mathcal{K}(q)$, contenant le point $(X(t_0), FX(t_0) + g)$, nous appliquons donc l'algorithme 4 qui teste si les conditions d'entrée de la trajectoire dans $\Delta_{q'}$ sont satisfaites.

Nous sommes donc maintenant en mesure de définir l'algorithme *ModeEntry* qui teste pour une condition initiale et un contrôle affine $u + FX + g$ donné, si les conditions d'entrée dans un mode q sont satisfaites. L'algorithme renvoie le cas échéant la cellule $\Delta_{q'}$ de l'espace état-contrôle dans laquelle évolue la trajectoire à partir du temps t_0 .

Algorithme 5 ModeEntry Conditions d'entrée dans un mode q

Données : n dimension de l'espace d'état, \mathcal{H} automate hybride, q mode discret de \mathcal{H} , X_0 un point initial donné dans D_q et $u = FX + g$ un contrôle affine admissible.

Sortie : Teste si le système \mathcal{H} quitte ou non le mode q au point X_0 et si non, calcule la cellule état-contrôle dans laquelle évolue la trajectoire (X, u) **sans la calculer** explicitement.

- 1: {Ensemble des cellules état-contrôle existant dans le mode q :}
 $cells := \{\Delta_{q'}; q' \in \mathcal{K}(q)\};$
 - 2: $i := 1;$
 {On teste les conditions d'entrée dans les cellules $\Delta_{q'}$ qui contiennent $(X_0, FX_0 + g)$:}
 - 3: **tant que** $i \leq \text{card } \mathcal{K}(q)$
 et $[(X_0, FX_0 + g) \notin cells[i] \text{ ou } CellEntry(n, \mathcal{H}, cells[i], X_0, u) = false]$ **faire**
 - 4: $i := i + 1;$
 - 5: **fin tant que**
 - 6: **si** $i \leq \text{card } cells$ **alors** Retourner $cells[i]$; **sinon** Retourner []; **fin si**
-

Calcul du temps de sortie

Une fois le critère d'entrée dans une cellule $\Delta_{q'}$ donnée satisfait, se pose le problème du calcul du temps de sortie t_f de la trajectoire considérée, de la cellule $\Delta_{q'}$ qu'elle traverse. Qu'il s'agisse de systèmes linéaires autonomes ou de systèmes avec des termes de contrôle, le problème reste le même et demeure extrêmement complexe.

[44] propose un algorithme de calcul des temps de sortie d'un mode par sous- et sur-approximation du temps de sortie t_f avec une précision donnée. Nous réutilisons tel quel cet algorithme dans toute simulation du modèle hybride.

L'algorithme 6 présenté ci-après calcule pour une condition initiale $X(0) = X_0$ et un contrôle $u = FX + g$ donnés, le temps et la face de sortie de la trajectoire $X[X_0, u]$ issue de X_0 selon u d'une cellule du maillage \mathcal{D} de l'espace d'état.

5.2 Propriétés topologiques du domaine contrôlable hybride liées à la convexité

Dans ce paragraphe nous nous intéressons aux propriétés topologiques du domaine contrôlable du modèle hybride \mathcal{H} dans un mode discret.

Algorithme 6 OutCell Calcul de la sortie d'un mode q

Données : \mathcal{H} , q mode discret, X_0 point initial, $u = FX + g$ contrôle admissible.

Sortie : $\begin{cases} [t_f, X_f, F_f] & \text{temps, point et face de sortie, s'ils existent, de la trajectoire} \\ & X[X_0, u] \text{ de la cellule } D_q \\ [] & \text{sinon.} \end{cases}$

- 1: $cell := ModeEntry(n, \mathcal{H}, q, X_0, u)$
- 2: **si** $cell \neq []$ **alors**
- 3: {Calcul de la dynamique :}
 $[A, B, c] := HybridDynamic(n, f, cell)$;
- 4: {Calcul de la trajectoire issue de X_0 selon le contrôle u :}
 $X[X_0, u]$ solution de : $\dot{X}(t) = AX(t) + Bu(t) + c$, $X(0) = X_0$.
- 5: {Calcul du premier temps d'intersection de $X[X_0, u]$ avec ∂D_q :}
 $t_f := +\infty$;
- 6: **pour** chaque face F de la cellule D_q **faire**
- 7: $t_{temp} :=$ premier temps d'intersection de la trajectoire $X[X_0, u](t)$ avec l'hyperplan support de F ;
- 8: **si** $t_{temp} > 0$ et $t_{temp} < t_f$ **alors** $t_f := t_{temp}$; $X_f := X[X_0, u](t_f)$; $F_f := F$; **fin si**
- 9: **fin pour**
- 10: **fin si**
- 11: Retourner $[t_f, X_f, F_f]$

Dans le cas des systèmes linéaires $\dot{X}(t) = AX(t) + Bu(t)$, le domaine contrôlable global du système est convexe pourvu que le domaine de contrôle \mathbb{U}_m soit également convexe et contienne l'origine 0, cf [61]. Dans cette partie, nous souhaitons adapter ce résultat dans un premier temps aux systèmes hybrides polyédraux affines au sens de la définition 1.3.2, puis au modèle hybride d'un système non linéaire construit au chapitre 3.

Le premier résultat que nous obtenons (et démontré dans la suite) est la convexité du domaine contrôlable en temps $T > 0$ fixé :

Proposition 5.2.1. *Soit q un état discret de l'automate hybride \mathcal{H} et τ_q un sous-ensemble de la cellule D_q , supposé convexe.*

Alors l'ensemble des points de D_q contrôlables jusqu'à τ_q en temps T est convexe.

Ensuite sous les hypothèses du théorème 2.3.3 énoncé au chapitre 2 page 47, à savoir :

i. Il existe $M > 0$ tel que toute trajectoire associée est uniformément bornée par M sur $[0, T]$

ii. Pour tout $X \in \mathbb{R}^n$, l'ensemble $V(X) = \{f_h(X, u); u \in K\}$ est convexe.

sachant que les contrôles admissibles considérés sont des fonctions mesurables à valeurs dans un compact $K \subset \mathbb{R}^m$, et d'après le théorème 5.2.1 énoncé dans [81], l'ensemble atteignable en temps t est compact et évolue continûment en t sur l'intervalle de temps $[0, T]$. D'où :

Proposition 5.2.2. *Soient $X_{1,0}$ et $X_{2,0}$ deux points contrôlables en temps T jusqu'à la cible τ par le système \mathcal{H} . On note X_i , $i \in \{1, 2\}$, la trajectoire du système \mathcal{H} qui amène $X_{i,0}$ à la cible en temps T et sans sortir de D_q .*

Alors, sous les hypothèses du théorème 2.3.3, tout point appartenant au domaine délimité par les trajectoires X_1 et X_2 sur $[0, T]$, la cible τ et le segment $[X_{1,0}, X_{2,0}]$ est contrôlable (cf figure 5.5).

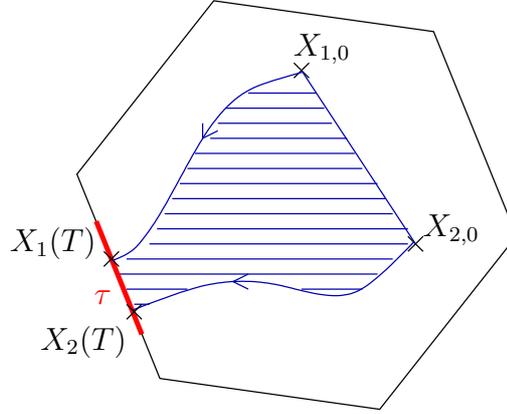


FIG. 5.5 – Les points situés dans le domaine hachuré sont contrôlables jusqu'à la cible τ

Les paragraphes 5.2.1 et 5.2.2 sont consacrés à la justification et la démonstration des propositions 5.2.1 et 5.2.2. Dans le dernier paragraphe, nous concluons par une discussion sur les propriétés topologiques, liées à la convexité, du domaine contrôlable hybride en temps quelconque dans une cellule.

5.2.1 Cas des systèmes hybrides polyédraux affines

Soit $\mathcal{H}_{\text{aff}} = (\mathcal{Q}, \mathcal{E}, \mathcal{D}, \mathcal{U}, \mathcal{F}, \mathcal{G}, \mathcal{R})$ un système hybride polyédral affine. Nous rappelons que dans chaque mode q du chemin γ considéré, la dynamique du système est affine. Tout au long de ce paragraphe, nous notons :

$$\forall X \in D_q, \forall u \in \mathbb{U}_m, f_q(X, u) = AX + Bu + c.$$

Commençons par démontrer la proposition 5.2.1, à savoir que l'ensemble des points contrôlables dans un mode q donné en temps T fixé est convexe.

Soient X_1 et X_2 deux points donnés de la cellule D_q supposés contrôlables en temps $T > 0$ jusqu'à la cible τ sans sortir de D_q . D'après la définition 4.3.2, il existe donc deux contrôles u_1 et u_2 admissibles tels que, pour $j \in \{1, 2\}$, le système :

$$\begin{cases} \dot{X}_j(t) &= AX_j(t) + Bu_j(t) + c \\ X_j(0) &= X_{j,0}, \quad X_j(T) \in \tau \end{cases}$$

admette une solution X_j , vérifiant : $\forall t \in [0, T], X_j(t) \in D_q$.

Nous démontrons alors la proposition 5.2.1, à savoir que si la cible τ est accessible en temps T à partir des points X_1 et X_2 de D_q tout en restant dans D_q , alors elle l'est aussi à partir de tout point du segment $[X_1, X_2]$.

Nous introduisons maintenant un point Y_0 quelconque du segment $]X_1, X_2[$, i.e. :

$$Y_0 = \alpha X_1 + (1 - \alpha)X_2, \quad \alpha \in]0, 1[$$

(On suppose en effet que Y_0 n'est égal ni à X_1 , ni à X_2 , car sinon, Y_0 est contrôlable par hypothèses). Montrons que Y_0 est contrôlable jusqu'à la cible τ dans la cellule D_q .

À tout instant $t \in [0, T]$, on définit :

$$u(t) = \alpha u_1(t) + (1 - \alpha)u_2(t) \quad (5.8)$$

Vérifions maintenant que la fonction u définie ci-dessus, est un contrôle admissible pour le système \mathcal{H}_{aff} .

Par définition, u_1 et u_2 sont des fonctions mesurables définies sur $[0, T]$. La fonction u est donc également mesurable sur $[0, T]$. De plus, la convexité du domaine de contrôle \mathbb{U}_m nous permet de conclure : $\forall t \in [0, T]$, $u(t) \in \mathbb{U}_m$. La fonction u définie par la formule (5.8), est donc bien une fonction de contrôle admissible pour le système \mathcal{H}_{aff} .

On introduit alors la trajectoire $Y(\cdot)$ de l'automate hybride \mathcal{H}_{aff} associée au contrôle u ainsi défini et à la condition initiale $Y(0) = Y_0$: en tout instant $t \in [0, T]$, auquel la dérivée $\dot{Y}(t)$ existe, on a donc :

$$\begin{cases} \dot{Y}(t) &= AY(t) + Bu(t) + c \\ Y(0) &= Y_0 \end{cases} \quad (5.9)$$

Pour prouver la contrôlabilité du point Y_0 , il reste donc à démontrer que la trajectoire $Y(\cdot)$ reste dans la cellule D_q sur l'intervalle de temps $[0, T]$ et qu'elle atteint la cible τ .

Soit $t \in [0, T]$ un instant auquel les dérivées $\dot{X}_1(t)$ et $\dot{X}_2(t)$ existent ; on démontre que la trajectoire $\alpha X_1(\cdot) + (1 - \alpha)X_2(\cdot)$ est une solution du système (5.9) :

$$\begin{aligned} \alpha X_1(0) + (1 - \alpha)X_2(0) &= \alpha X_{1,0} + (1 - \alpha)X_{2,0} = Y_0 \\ \alpha \dot{X}_1(t) + (1 - \alpha)\dot{X}_2(t) &= \alpha (AX_1(t) + Bu_1(t) + c) + (1 - \alpha)(AX_2(t) + Bu_2(t) + c) \\ &= A(\alpha X_1(t) + (1 - \alpha)X_2(t)) + Bu(t) + c \end{aligned}$$

D'après l'unicité de la solution du problème de Cauchy (5.9) (voir paragraphe 2.3.1), nous pouvons conclure :

$$\forall t \in [0, T], \quad Y(t) = \alpha X_1(t) + (1 - \alpha)X_2(t).$$

Or la cellule D_q est convexe, d'où : $\forall t \in [0, T]$, $Y(t) \in D_q$.

Calculons l'extrémité $Y(T)$ de la trajectoire $Y(\cdot)$: $Y(T) = \alpha X_1(T) + (1 - \alpha)X_2(T)$. Par hypothèses, on sait qu'à l'instant T , la trajectoire $X_j(\cdot)$ ($j = 1, 2$) atteint la cible τ , d'où : $X_1(T) \in \tau$ et $X_2(T) \in \tau$. De plus, par hypothèses, τ est un ensemble convexe ce qui nous permet de conclure :

$$Y(T) \in \tau.$$

Les hypothèses i. et ii. citées au début de cette partie et nécessaires pour avoir la continuité en temps de l'ensemble atteignable, sont vérifiées de façon immédiate par le système hybride \mathcal{H}_{aff} affine dans la cellule D_q .

5.2.2 Cas du modèle hybride \mathcal{H} construit au chapitre 3

On considère maintenant le modèle hybride \mathcal{H} défini au chapitre 3. On rappelle que, dans chaque cellule D_q du chemin γ , la dynamique est affine par morceaux, définie par :

$$\forall q' \in \mathcal{K}(q), \forall (X, u) \in \Delta_{q'}, f_q(X, u) = A_{q'}X + B_{q'}u + c_{q'},$$

sachant que : $D_q \times \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} \Delta_{q'}$. Soient X_1 et X_2 deux points contrôlables en temps T dans la cellule D_q relativement à une cible convexe τ . Il existe donc deux contrôles u_1 et u_2 admissibles tels que, pour $j \in \{1, 2\}$, le système :

$$\begin{cases} \dot{X}_j(t) &= f_q(X_j(t), u_j(t)) \\ X_j(0) &= X_{j,0}, \quad X_j(T) \in \tau \end{cases}$$

admette une solution X_j , vérifiant : $\forall t \in [0, T], X_j(t) \in D_q$.

On se donne un point Y_0 du segment $]X_1, X_2[: Y_0 = \alpha X_1 + (1 - \alpha)X_2$, $\alpha \in]0, 1[$, dont on veut montrer qu'il est contrôlable vers la cible τ dans la cellule D_q .

La difficulté de la démonstration réside dans le fait que l'approximation affine par morceaux f_h n'est pas nécessairement convexe sur $\mathbb{R}^n \times \mathbb{U}_m$ ce qui complique singulièrement le choix d'un contrôle u qui amènerait Y_0 à la cible sans sortir de la cellule. En revanche, il nous semble qu'une solution serait d'utiliser la proposition suivante (cf chapitre 8, théorème 8.2 ou la remarque page 124 du livre [1] de A.A. Agrachev et Y.L. Sachkov) :

Théorème 5.2.1 (Approximation du flot d'une EDO [1]). *Le flot du système :*

$$\dot{X}(t) = \alpha(t)f_h(X(t), u_1(t)) + [1 - \alpha(t)]f_h(X(t), u_2(t)), \quad \alpha(t) \in [0, 1]$$

peut être approché par le flot de systèmes différentiels de la forme :

$$\dot{X}(t) = f_h(X(t), v(t)) \quad \text{où } v(t) \in \{u_1(t), u_2(t)\}.$$

En effet, il suffit alors de définir la trajectoire Y par : $Y(t) = \alpha X_1(t) + (1 - \alpha)X_2(t)$, ce qui nous donne (d'après la convexité de la cible τ) les conditions aux bords cherchées :

$$Y(0) = Y_0 \text{ et } Y(T) = \alpha X_1(T) + (1 - \alpha)X_2(T) \in \tau.$$

Il reste donc à vérifier que Y est une trajectoire du modèle hybride \mathcal{H} ; par construction :

$$\dot{Y}(t) = \alpha \dot{X}_1(t) + (1 - \alpha) \dot{X}_2(t).$$

D'après le théorème 5.2.1, il existe un contrôle u tel que : $\forall t \in [0, T], u(t) \in \{u_1(t), u_2(t)\}$ et Y est une trajectoire approchée du modèle hybride \mathcal{H} associée au contrôle u . Le contrôle u ainsi caractérisé est bien mesurable en valeurs dans \mathbb{U}_m , donc admissible. Il s'en suit la contrôlabilité du point Y_0 par passage à la limite.

Remarque 5.2.1. *Afin d'être plus précis dans la démonstration, il faudrait obtenir à partir du théorème 5.2.1 une majoration de la distance entre le flot initial et son approximation, ce qui nous permettrait d'en déduire l'erreur commise au niveau des trajectoires.*

5.2.3 Discussion sur la convexité du domaine contrôlable hybride dans une cellule D_q

Remarquons tout d'abord que dans le cas où la cible locale est l'origine 0 de l'espace d'état, le domaine contrôlable hybride $\mathcal{C}_{\mathcal{H}}(\{0\}, D_q)$ est effectivement convexe.

En effet, considérons deux points X_1 et X_2 contrôlables dans la cellule D_q respectivement en temps T_1 et T_2 . Il existe donc un contrôle admissible u_1 (resp. u_2) qui permet d'amener le point X_1 (resp. X_2) à la cible en temps T_1 (resp. T_2).

Nous introduisons alors le contrôle :

$$\tilde{u}_1(t) = \begin{cases} u_1(t) & \text{si } t \in [0, T_1] \\ 0 & \text{si } t \in [T_1, T_2] \end{cases}$$

Le point X_1 est alors contrôlable en temps T_2 selon le contrôle \tilde{u}_1 . D'après la proposition 5.2.1, tout point du segment $[X_1, X_2]$ est contrôlable jusqu'à la cible τ dans la cellule D_q considérée en temps T_2 .

De plus, par extension, si tout point X_f de la cible τ considérée est un point d'équilibre du système i.e. :

$$\forall X_f \in \tau, \exists u_f \in \mathbb{U}_m, f_h(X_f, u_f) = 0$$

alors le domaine contrôlable hybride dans la cellule D_q est convexe.

Dans le cas général, le problème de la contrôlabilité à la cible évoque celui du contrôle dans un mode q donné d'un système hybride affine par morceaux vers une face de la cellule D_q associée [82, 47]. En effet, les auteurs de [82, 47] proposent des conditions nécessaires et suffisantes pour qu'une face donnée d'une cellule soit accessible à partir d'un point donné et fournit un contrôle en boucle de rétroaction qui amène le système à la position souhaitée.

Une idée pour étendre le domaine contrôlable obtenu par la proposition 5.2.2 serait de choisir une cible intermédiaire et de calculer le domaine contrôlable jusqu'à cette cible dans la cellule considérée.

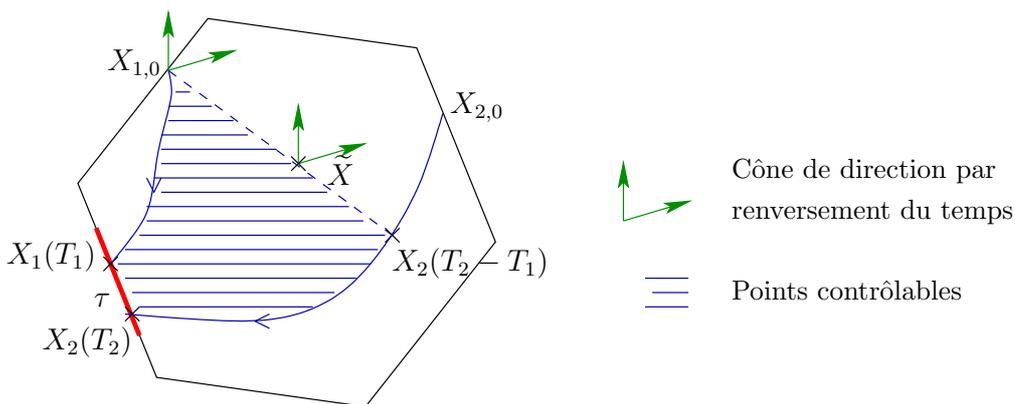


FIG. 5.6 – Idée pour prolonger le domaine contrôlable obtenu en temps fixé.

La figure 5.6 illustre cette idée : en effet, considérons deux points X_1 et X_2 , contrôlables respectivement en temps T_1 et T_2 avec $T_1 < T_2$, dans la cellule D_q . D'après les résultats

précédents, le segment $[X_{1,0}, X_2(T_2 - T_1)]$ est contrôlable en temps T_1 et le domaine hachuré est contrôlable par continuité.

Nous pouvons alors par exemple choisir une nouvelle cible locale $[\tilde{X}, X_2(T_2 - T_1)]$ à partir de laquelle on calcule un nouveau domaine contrôlable. Ce procédé répété sur toutes les cibles pourrait conduire à un algorithme de sous-approximation du domaine contrôlable hybride dans une cellule. Malheureusement la complexité d'une telle approche serait bien trop élevée.

En outre, bien que le domaine contrôlable hybride dans une cellule ne soit en général pas convexe, de nombreux exemples tels que celui du ressort (cf exemple 5.3.3) exhibent des domaines locaux convexes ce qui nous incite à considérer le cas particulier de ces systèmes.

Cas particulier d'un domaine contrôlable convexe par morceaux sur un chemin

Nous supposons dans ce paragraphe que la propriété de convexité est vérifiée dans chaque cellule de l'automate \mathcal{H} , soit :

$$\forall q \in \mathcal{Q}, [\tau \subset D_q \text{ convexe} \Rightarrow \mathcal{C}_{\mathcal{H}}(\tau, D_q) \text{ convexe}] . \quad (5.10)$$

Soit $\gamma = (q_i)_{i=0\dots r}$ un chemin discret de modes adjacents de l'automate hybride affine par morceaux \mathcal{H} considéré. D'après la proposition 4.3.2 appliquée au modèle hybride \mathcal{H} :

$$\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_{\gamma}) = \mathcal{C}_{\mathcal{H}}(\tau_0, D_{q_0}) \cup \mathcal{C}_{\mathcal{H}}(\tau_1, D_{q_1}) \cup \dots \cup \mathcal{C}_{\mathcal{H}}(\tau_r, D_{q_r})$$

sachant que les cibles successives sont définies par les relations :

$$\begin{cases} \tau_0 = \{0\} \\ \tau_{i+1} = \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}, \quad i = 0, \dots, r-1 \end{cases}$$

Le domaine contrôlable $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_{\gamma})$ du système hybride \mathcal{H} est donc décomposé en une union de sous-domaines contrôlables convexes définis dans les cellules du chemin γ :

Proposition 5.2.3. *Soit $\gamma = (q_i)_{i=0\dots r}$ une suite de modes adjacents de l'automate hybride \mathcal{H} . Alors, sous la condition (5.10), quel que soit $i = 0, \dots, r$, τ_i et $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$ sont convexes.*

Preuve. Soit P_i : “ τ_i et $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$ sont convexes” la propriété à démontrer.

Pour $i = 0$, la cible est réduite à un point : $\tau_0 = \{0\}$ et est donc convexe. D'après la remarque faite au début de ce paragraphe, $\mathcal{C}_{\mathcal{H}}(\tau_0, D_{q_0})$ est également convexe. Donc P_0 est vérifiée.

Supposons τ_i et $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$ convexes. Au rang $i + 1$, la cible τ_{i+1} est définie par :

$$\tau_{i+1} = \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}$$

d'après la proposition 4.3.2. Or, par hypothèse de récurrence, $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$ est convexe. La garde $G_{(q_i, q_{i+1})}$ est une face convexe du polyèdre D_{q_i} . La cible τ_{i+1} est donc l'intersection de deux ensembles convexes, ce qui implique sa convexité. D'après la condition (5.10), on peut alors conclure que si P_i est vraie, alors P_{i+1} l'est aussi.

□

Ainsi, dans le cas où les domaines contrôlables locaux successifs sont convexes, l'algorithme 9 présenté ci-après va nous permettre de calculer une sous-approximation convexe par morceaux du domaine contrôlable hybride le long du chemin γ considéré, garantissant ainsi la contrôlabilité des points de l'approximation.

5.3 Algorithme de calcul d'une approximation convexe du domaine contrôlable

On considère le modèle hybride \mathcal{H} construit au chapitre 3 dont la dynamique est régie par l'équation :

$$\dot{X}(t) = f_h(X(t), u(t)),$$

où f_h est une application affine par morceaux. On cherche à calculer une approximation du domaine contrôlable à l'origine de ce système i.e. de l'ensemble des points X_0 de l'espace d'état qui peuvent être amenés en 0 en temps fini par le système \mathcal{H} .

Au chapitre 4, nous avons vu que le domaine contrôlable d'un système hybride est rarement calculable. Afin de réduire l'exploration des états discrets du système, la partie 4.3 propose une approche constructive de la contrôlabilité par l'intermédiaire des domaines contrôlables hybrides définis sur des chemins donnés de cellules dans l'espace d'état. Le but est maintenant de proposer un algorithme de calcul du domaine contrôlable hybride défini sur un chemin donné. Plus précisément, nous développons dans la suite un algorithme calculant une approximation convexe de ce domaine.

L'élément de base d'un tel algorithme est donc le calcul d'une approximation du domaine contrôlable défini dans un mode q donné de l'automate hybride \mathcal{H} .

5.3.1 Approximation convexe contrôlable dans un mode q

Considérons un mode q du modèle hybride \mathcal{H} ou, de manière équivalente, une cellule D_q de l'espace d'état. On se donne une cible convexe, notée τ_q , dans la cellule D_q . On suppose de plus que la cible τ_q est un polytope de \mathbb{R}^n , défini par la donnée de ses sommets :

$$\tau_q = \text{Conv}(\tau_{q,1}, \dots, \tau_{q,k}) \subset D_q.$$

Maintenant nous voulons calculer une approximation de l'ensemble $\mathcal{C}_{\mathcal{H}}(\tau_q, D_q)$ des points contrôlables jusqu'à la cible τ_q dans la cellule D_q i.e. de l'ensemble des points de la cellule D_q à partir desquels la cible τ_q peut être atteinte en temps fini sans sortir de D_q .

L'idée générale développée ci-après, consiste à calculer dans la cellule D_q des points contrôlables particuliers ; nous introduisons l'enveloppe convexe de ces points comme approximation du domaine contrôlable dans le mode q considéré. Si le domaine contrôlable hybride défini dans le mode q est convexe (cf paragraphe 5.2), nous obtenons alors une sous-approximation contrôlable de ce domaine.

Méthode pour calculer des points contrôlables

D'après la proposition 4.1.1 démontrée au chapitre 4, un point X_0 est contrôlable jusqu'à la cible τ_q par le système \mathcal{H} dans la cellule D_q , si et seulement s'il est atteignable à partir de la cible τ_q en temps fini par le système :

$$\dot{X}(t) = -f_h(X(t), u(t)), \quad u(t) \in \mathbb{U}_m \quad (5.11)$$

Par conséquent, pour calculer simplement un point contrôlable du système \mathcal{H} dans la cellule D_q , il suffit de choisir un contrôle u , tel que la trajectoire du système (5.11) issue de la cible τ_q selon u ne quitte pas D_q à l'instant initial.

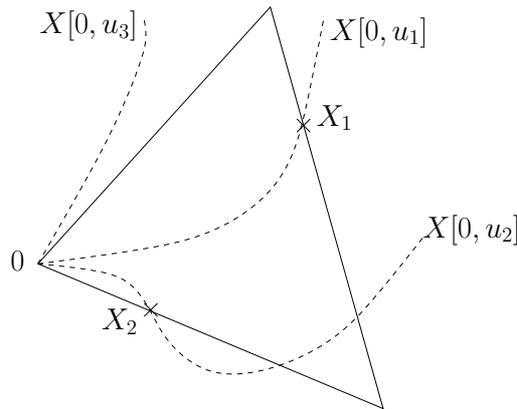


FIG. 5.7 – Principe de calcul d'une approximation convexe contrôlable dans un état q contenant la cible : $\tau_q = \{0\}$. Le long de la trajectoire $X[0, u_1]$ (resp. $X[0, u_2]$), seuls les points compris entre 0 et X_1 (resp. 0 et X_2), sont contrôlables ; par contre, aucun point de $X[0, u_3]$ n'est contrôlable dans D_q .

Plus précisément, on détermine les trajectoires, notées $X[\tau_{q,i}, u]$, issues des sommets $\tau_{q,i}$ de la cible τ_q . Si $X[\tau_{q,i}, u]$ quitte la cellule D_q à l'instant $t = 0$, alors aucun de ses points n'est contrôlable dans D_q . Dans le cas contraire, jusqu'à ce que $X[\tau_{q,i}, u]$ sorte "pour la première fois" de D_q , tout point de la trajectoire est contrôlable (cf figure 5.7).

La difficulté de cette approche réside donc dans le choix des contrôles, qui vont nous permettre de calculer une approximation convexe du domaine contrôlable hybride. En effet, l'expression des contrôles choisis doit être suffisamment simple pour permettre un calcul efficace des trajectoires associées, tout en étant assez significatifs afin d'obtenir une approximation du domaine contrôlable hybride de bonne qualité.

Choix des fonctions de contrôle D'après la construction du maillage de l'espace état-contrôle présentée au chapitre 3, on sait qu'il existe une colonne de cellules $\{\Delta_{q'}; q' \in \mathcal{K}(q)\}$ de $\mathbb{R}^n \times \mathbb{U}_m$ qui se projettent exactement sur la cellule D_q . Dans chaque cellule $\Delta_{q'}$ de la colonne, la dynamique du système \mathcal{H} est affine.

L'astuce consiste alors à choisir une fonction de contrôle u , de telle sorte que le couple $(X(t), u(t))$ reste dans une même cellule $\Delta_{q'}$ de la colonne de cellules définie par $\mathcal{K}(q)$ (cf

condition (5.13)), tant que $X(t)$ ne sort pas de D_q (cf condition (5.12)) :

$$\exists T > 0, \forall t \in [0, T], X(t) \in D_q \quad (5.12)$$

$$\forall t \in [0, T], (X(t), u(t)) \in \Delta_{q'}. \quad (5.13)$$

La solution que nous proposons ici consiste à considérer des contrôles affinement dépendants de la position du système :

$$u_j(t) = F_j X(t) + g_j, \quad (5.14)$$

tels que la trajectoire état-contrôle $(X(t), u(t))$ évolue dans le mode q , sur les bords d'une cellule $\Delta_{q'}$ de la colonne définie par $\mathcal{K}(q)$ (cf figure 5.8).

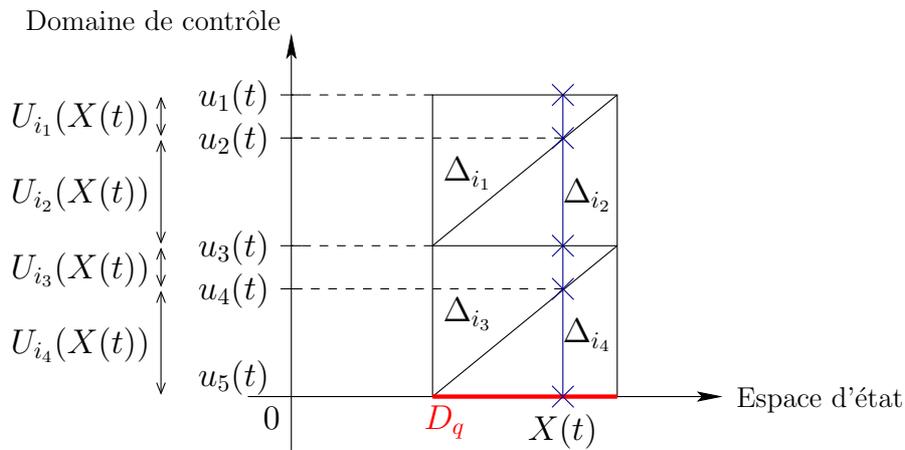


FIG. 5.8 – Choix des contrôles $u_i(t)$ en fonction de la position $X(t)$ pour le calcul d'une approximation convexe du domaine contrôlable

Concrètement, cela revient à déterminer l'ensemble des sommets $u_j(t)$ des domaines de contrôles induits $U_{q'}(X(t))$, ce qui est chose faite au paragraphe 3.2.3.

Remarque 5.3.1. *Deux domaines de contrôles adjacents non réduits à un point, possèdent exactement m sommets en commun, ce qui réduit considérablement le nombre de contrôles u_i à considérer.*

Ainsi, les conditions (5.12) et (5.13) sont satisfaites et la dynamique du système \mathcal{H} obéit à une équation différentielle affine par rapport à l'état :

$$\dot{X}(t) = A_{q'} X(t) + B_{q'} u_j(t) + c_{q'} \quad \text{soit : } \dot{X}(t) = (A_{q'} + B_{q'} F_j) X(t) + (B_{q'} g_j + c_{q'}).$$

que l'on sait résoudre (cf paragraphe 5.1.1).

Calcul d'une approximation convexe du domaine contrôlable hybride Nous sommes maintenant en mesure de définir un algorithme de calcul d'une approximation convexe du domaine contrôlable hybride, voir algorithme 7. Le principe illustré par la figure 5.9 est le suivant : pour chaque contrôle u_j défini précédemment et pour chaque sommet $\tau_{q,i}$ de la cible considérée,

1. on teste si la trajectoire issue de $\tau_{q,i}$ par renversement du temps selon le contrôle u_j reste ou non dans la cellule D_q (cf paragraphe 5.1.2).
2. Si oui, on calcule effectivement cette trajectoire, ce qui revient à résoudre le système :

$$\begin{cases} \dot{X}(t) &= (A_{q'} + B_{q'}F_j)X(t) + B_{q'}g_j + c_{q'} \\ X(0) &= \tau_{q,i} \end{cases}$$

où $q' \in \mathcal{K}(q)$ désigne l'indice d'une cellule $\Delta_{q'}$ dans laquelle évolue le couple (X, u_j) au cours du temps (cf paragraphe 5.1.1).

3. On détermine alors la première intersection, notée $X_{i,j}$, avec les frontières de D_q :

$$X_{i,j} = X[\tau_{q,i}, u_j](T_{i,j}), \text{ en notant : } T_{i,j} = \sup\{t < 0 ; X[\tau_{q,i}, u_j](t) \in \partial D_q\}$$

Le point d'intersection $X_{i,j}$ ainsi obtenu, est contrôlable jusqu'au point $\tau_{q,i}$ pour le contrôle u_j par le système \mathcal{H} .

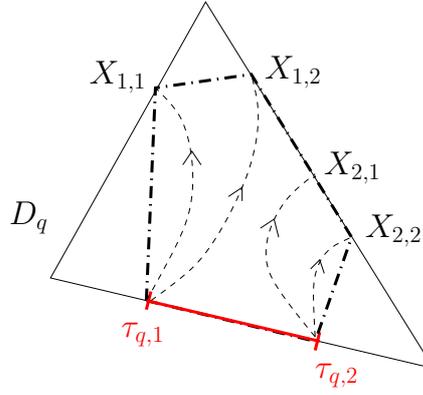


FIG. 5.9 – Principe de l'approximation convexe du domaine contrôlable dans un état q

Notons $\Lambda(\tau_q, D_q)$ l'enveloppe convexe des points $X_{i,j}$ contrôlables dans D_q , calculée par l'algorithme 7. Par construction et d'après l'étude menée au paragraphe 5.2, nous pouvons conclure :

Proposition 5.3.1. $\Lambda(\tau_q, D_q)$ est une approximation convexe du domaine contrôlable, noté $\mathcal{C}_{\mathcal{H}}(\tau_q, D_q)$, du système hybride \mathcal{H} dans le mode q .

De plus :

$$\mathcal{C}_{\mathcal{H}}(\tau_q, D_q) \text{ convexe} \Rightarrow \Lambda(\tau_q, D_q) \subset \mathcal{C}_{\mathcal{H}}(\tau_q, D_q).$$

Dans ce cas, $\Lambda(\tau_q, D_q)$ est une sous-approximation contrôlable.

5.3.2 Cas des automates hybrides polyédraux, affines par morceaux

Au paragraphe précédent, nous avons proposé un algorithme qui, pour un automate hybride du type introduit au chapitre 3, calcule une approximation convexe du domaine contrôlable dans une cellule D_q du maillage \mathcal{D} , vers une cible polyédrale donnée. Comme nous l'avons déjà fait dans [73], nous proposons maintenant une généralisation de cet algorithme aux automates hybrides polyédraux affines par morceaux introduits au chapitre 1.

Algorithme 7 ConvexApproximation : Approximation convexe contrôlable en mode q

Données : \mathcal{H} , q un état discret de l'automate hybride \mathcal{H} , $\tau_q = \text{Conv}(\tau_{q,1}, \dots, \tau_{q,k})$ la cible dans l'état q définie comme l'enveloppe convexe de ses sommets.

Sortie : une approximation convexe du domaine contrôlable dans D_q .

- 1: Approximation initiale : $\Lambda := \emptyset$;
 - 2: {On calcule l'ensemble des indices des cellules $\Delta_{q'}$ se projetant exactement sur D_q :}
 $\mathcal{K}(q) := \{q' \in I / p_{\mathbb{R}^n}^\perp(\Delta_{q'}) = D_q\}$
 - 3: {Ensemble des contrôles déjà testés :} $\Sigma := \emptyset$.
 - 4: **pour tout** $q' \in \mathcal{K}(q)$ **faire**
 - 5: {On calcule $U_{q'}(X)$ polytope des contraintes sur le contrôle (cf §3.2.3) :}
 $U_{q'}(X) := \text{ControlConstraints}(\Delta_{q'})$;
 - 6: {Pour chaque sommet de $U_{q'}(X)$ non testé,}
 - 7: **pour tout** i variant de 1 à $\text{card}(U_{q'}(X))$, tel que $U_{q'}(X)[i] \notin \Sigma$ **faire**
 - 8: ième sommet de Σ : $u := t \rightarrow U_{q'}(X(t))[i]$; $\Sigma_t := \Sigma \cup \{U_{q'}(X)[i]\}$;
 - 9: {Pour chaque sommet $\tau_{q,j}$ de la cible τ_q ,}
 - 10: **pour tout** j variant de 1 à k **faire**
 - 11: **si** $\text{ModeEntry}(n, \mathcal{H}, q, \tau_{q,j}, u)$ **alors**
 - 12: {Calcul de l'intersection de la trajectoire $X[\tau_{q,j}, u]$ avec le bord de D_q :}
 $(X_f, \text{facesortie}) := \text{OutCell}(\mathcal{H}, q, \tau_{q,j}, u)$;
 - 13: {Le point ainsi calculé, s'il existe, appartient à l'approximation contrôlable}
 - 14: **si** $X_f \neq \emptyset$ **alors** $\Lambda := \Lambda \cup \{X_f\}$; **fin si**
 - 15: **fin si**
 - 16: **fin pour**
 - 17: **fin pour**
 - 18: **fin pour**
 - 19: Retourner $\text{Conv}(\Lambda)$.
-

On introduit dans ce paragraphe l'automate hybride \mathcal{H}_{aff} polyédral, affine par morceaux. On rappelle que \mathcal{H}_{aff} est défini sur une partition \mathcal{D}_{aff} en polyèdres (quelconques) de l'espace d'état \mathbb{R}^n . Dans chaque mode $q \in \mathcal{Q}_{aff}$ de l'automate, la dynamique est régie par une équation différentielle affine :

$$\dot{X}(t) = A_q X(t) + B_q u(t) + c_q,$$

et le contrôle est à valeurs dans un polytope \mathbb{U}_m de \mathbb{R}^m indépendant de la position du système. On rappelle que le polytope \mathbb{U}_m est défini par la donnée de ses sommets : $\mathbb{U}_m = \text{Conv}(s_1, \dots, s_p)$.

Appliquer l'algorithme d'approximation convexe précédent à l'automate \mathcal{H}_{aff} dans un mode q donné, revient concrètement à chercher, s'ils existent, les points d'intersection des trajectoires $X[\tau_{q,i}, s_j]$ dans D_q , issues des sommets de la cible τ selon les contrôles $u = s_j$ avec la frontière de la cellule D_q . Cependant, les résultats obtenus de cette façon sont, même sur des exemples simples, de qualité médiocre.

Exemple 5.3.1. *On considère le système linéaire suivant :*

$$\dot{X}(t) = \begin{bmatrix} -1 & -2 \\ -3 & -1 \end{bmatrix} u(t), \quad X(t) \in D_q = \text{Conv}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right),$$

sous la contrainte : $\forall t, u(t) \in \mathbb{U}_2 = \text{Conv}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right)$.

Dans cet exemple, nous nous intéressons au calcul d'une approximation convexe du domaine contrôlable au point 0 dans la cellule D_q . Nous avons de plus choisi un système affine local simple dont on connaît le domaine contrôlable exact dans la cellule D_q , afin de le comparer à l'approximation calculée par l'algorithme 9.

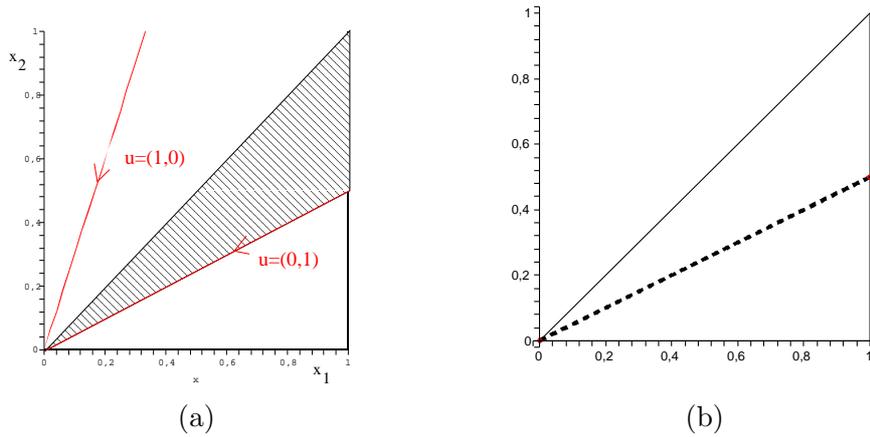


FIG. 5.10 – (a) Domaine contrôlable exact (hachuré) (b) Approximation convexe donnée par l'algorithme 9 (en pointillés)

Comme le montre la figure 5.10, la trajectoire par renversement du temps selon le contrôle $u = (1, 0)$ quitte immédiatement la cellule D_q considérée au point 0. De ce fait, le seul point calculé par l'algorithme 9 est celui trouvé pour le contrôle $u = (0, 1)$. L'approximation convexe contrôlable est donc très éloignée du domaine contrôlable exact.

Il ressort de cet exemple que pour améliorer l'approximation, nous avons besoin de davantage de points contrôlables sur la frontière de la cellule D_q , ce qui implique de considérer d'autres contrôles que ceux donnés par les sommets du polytope \mathbb{U}_m .

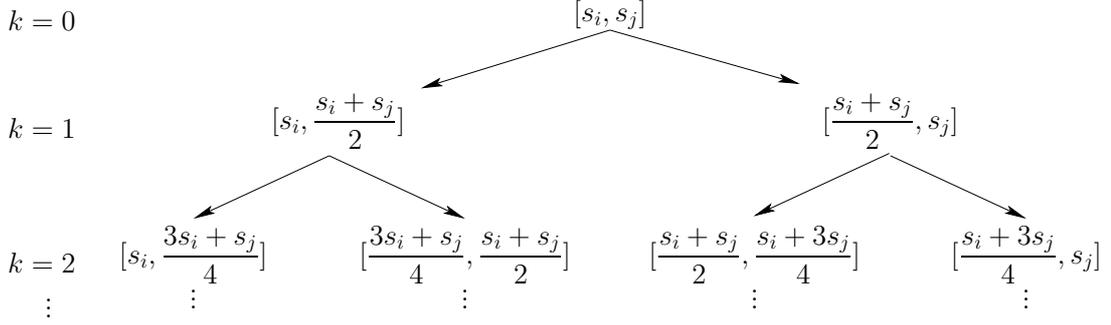
Choix des fonctions de contrôles

L'astuce consiste à subdiviser les arêtes du domaine de contrôle \mathbb{U}_m par dichotomie pour une précision $\delta > 0$ fixée.

Considérons une arête $[s_i, s_j]$ du domaine de contrôle \mathbb{U}_m . On introduit alors un schéma de subdivision du segment $[s_i, s_j]$ par dichotomie de la façon suivante :

À chaque étape k du schéma présenté sur la figure 5.11, on obtient une subdivision uniforme $(u_{k,p})_{p=0..k+1}$ de $[s_i, s_j]$ de taille $\frac{d(s_i, s_j)}{2^k}$, définie comme suit :

$$\begin{cases} u_{0,0} = s_i, & u_{0,1} = s_j \\ u_{k+1,p+1} = \frac{u_{k,p} + u_{k,p+1}}{2} \end{cases}, \quad (5.15)$$

FIG. 5.11 – Subdivision par dichotomie de l'arête $[s_i, s_j]$ du domaine de contrôle \mathbb{U}_m

soit :

$$\forall k \in \mathbb{N}, \forall p = 0, \dots, k+1, u_{k,p} = \frac{(2^k - p)s_i + ps_j}{2^k}.$$

L'algorithme se termine au pire dès que la taille de la subdivision est inférieure à la précision δ choisie :

$$\frac{d(s_i, s_j)}{2^k} \leq \delta. \quad (5.16)$$

Nous allons maintenant montrer qu'il n'est pas toujours nécessaire de subdiviser l'intervalle $[s_i, s_j]$ jusqu'à ce que la condition (5.16) soit satisfaite. L'idée est en fait d'éviter d'effectuer une étape de plus dans la subdivision lorsque l'approximation ne peut plus être améliorée.

Amélioration de l'approximation convexe

On se place dans un mode q de l'automate \mathcal{H}_{aff} . On rappelle que dans la cellule D_q associée, la dynamique du système \mathcal{H}_{aff} est régie par le système différentiel suivant :

$$\dot{X}(t) = A_q X(t) + B_q u(t) + c_q$$

L'algorithme 8 proposé ci-après, calcule une approximation convexe du domaine contrôlable dans la cellule D_q grâce au schéma de subdivision (5.15) défini précédemment. En particulier, la subdivision n'est effectuée que si l'approximation peut encore être améliorée. Cet algorithme repose sur le lemme auxiliaire suivant :

Lemme 5.3.1 (Lemme auxiliaire). *On considère le système affine suivant : $\dot{X}(t) = AX(t) + Bu(t) + c$. Soit u un contrôle constant tel que : $u \in]u_i, u_j[$. Alors, quel que soit le point initial X_0 dans l'espace d'état, on a :*

$$\forall t, X[X_0, u](t) \in]X[X_0, u_i](t), X[X_0, u_j](t)[,$$

De même, si u est un contrôle constant à valeurs dans $\text{Conv}(u_1, \dots, u_k)$, alors :

$$\forall t, X[X_0, u](t) \in \text{Conv}(X[X_0, u_i](t), i = 1, \dots, k).$$

Preuve. Soit $u \in \text{Conv}(u_1, \dots, u_k)$ un contrôle constant ; par définition de la convexité, il existe des scalaires $(\alpha_i)_{i=1\dots k} \in [0, 1]^k$, tels que : $u = \sum_{i=1}^k \alpha_i u_i$ et $\sum_{i=1}^k \alpha_i = 1$. La dynamique considérée étant affine, nous pouvons calculer chacune des trajectoires issues du point X_0 selon les contrôles $(u_i)_{i=1\dots k}$ de façon formelle (cf paragraphe 5.1.1) :

$$\forall i = 1, \dots, k, \quad X[X_0, u_j](t) = e^{A_q t} X_0 + e^{A_q t} \int_0^t e^{-A_q s} (B_q u_j + c_q) ds.$$

Nous pouvons alors calculer la trajectoire $X[X_0, u]$ en fonction des $X[X_0, u_j]$:

$$\begin{aligned} X[X_0, u](t) &= e^{A_q t} X_0 + e^{A_q t} \int_0^t e^{-A_q s} (B_q u + c_q) ds \\ &= e^{A_q t} X_0 + e^{A_q t} \int_0^t e^{-A_q s} (B_q \sum_{i=1}^k \alpha_i u_i + c_q) ds \\ &= \sum_{i=1}^k \alpha_i \left(e^{A_q t} X_0 + e^{A_q t} \int_0^t e^{-A_q s} (B_q u_i + c_q) ds \right) = \sum_{i=1}^k \alpha_i X[X_0, u_i](t) \end{aligned}$$

ce qui nous permet de conclure :

$$X[X_0, u](t) \in \text{Conv}(X[X_0, u_i](t), i = 1, \dots, k).$$

□

Soit $[u_i, u_j]$ un sous-intervalle d'une arête de \mathbb{U}_m obtenu après k dichotomies selon le schéma (5.15). On introduit maintenant pour $k \in \{i, j\}$, le premier instant $T_{q,k} = \sup\{t < 0 ; X[\tau, u_k](t) \in \partial D_q\}$ où la trajectoire $X[\tau, u_k]$ calculée par renversement du temps, quitte la cellule D_q . On note alors, s'il existe,

$$X_{q,k} = X[\tau, u_k](T_{q,k})$$

le point d'intersection de $X[\tau, u_k]$ avec le bord de la cellule D_q et $F_{q,k}$ la face de sortie :

1. Si les faces $F_{q,i}$ et $F_{q,j}$ existent et sont égales, alors il n'est pas utile de raffiner l'approximation en subdivisant à nouveau l'intervalle $[u_i, u_j]$. En effet, grâce au lemme 5.3.1, on sait que pour tout contrôle $u \in]u_i, u_j[$, la trajectoire associée est comprise entre celles associées à u_i et u_j . Par convexité, le point d'intersection de la trajectoire selon u avec le bord de D_q appartient donc déjà à l'approximation (cf figure 5.12).
2. Supposons que les faces $F_{q,i}$ et $F_{q,j}$ sont distinctes : $F_{q,i} \neq F_{q,j}$ (cf figures 5.13-(a) et 5.13-(b)) ou alors toutes les deux vides : $F_{q,i} = F_{q,j} = \emptyset$ (cf figure 5.13-(c)).

On subdivise alors à nouveau l'intervalle $[u_i, u_j]$ par dichotomie, puis on applique l'algorithme de calcul du point sortie de la cellule D_q pour le contrôle $\frac{u_i + u_j}{2}$: suivant la configuration géométrique des trajectoires (cf figure 5.13), on obtient ou non un nouveau point X et une face F (éventuellement vide) de D_q . On réitère alors l'algorithme sur les intervalles $[u_i, \frac{u_i + u_j}{2}]$ et $[\frac{u_i + u_j}{2}, u_j]$.

D'un point de vue expérimental, les résultats obtenus par l'algorithme 8 fournissent d'excellents résultats. Une étude de la convergence de l'approximation convexe du domaine contrôlable ainsi calculée est développée au paragraphe 4.3.3.

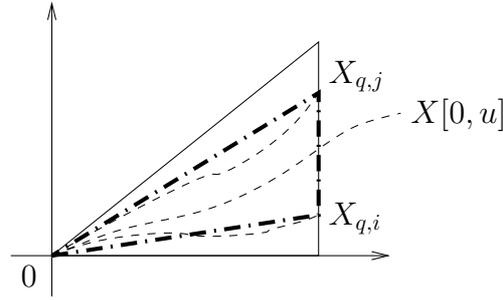


FIG. 5.12 – Quel que soit $u \in [u_i, u_j]$, le point de sortie de la trajectoire $X[0, u]$ de la cellule D_q est déjà dans l'approximation convexe contrôlable

Algorithme 8 DiscreteEdge Raffinement de l'approximation

Données : q, X_f (point cible), $u_i, u_j, h > 0$ pas de discrétisation.

$(X_i, F_i) := OutCell(q, X_f, u_i)$; $(X_j, F_j) := OutCell(q, X_f, u_j)$;

Sortie : Λ ensemble des points contrôlables.

- 1: $\Lambda := \emptyset$;
 - 2: **si** $distance(u_i, u_j) \geq h$ et $(F_i \neq F_j$ or $F_i = F_j = \emptyset)$ {cas 1.} **alors**
 - 3: $(X, F) := OutCell(q, X_f, \frac{u_i+u_j}{2})$;
 - 4: **si** $X \neq \emptyset$ **alors** $\Lambda := \{X\}$; **fin si** { X est un point de l'approximation cherchée}
 - 5: **si** $X_i \neq \emptyset$ et $X_j \neq \emptyset$ {cf figure 5.13} **alors**
 - 6: **si** $F \neq F_i$ **alors** $\Lambda := \Lambda \cup DiscreteEdge(q, X_f, u_i, \frac{u_i+u_j}{2}, h)$; **fin si**
 - 7: **si** $F \neq F_j$ **alors** $\Lambda := \Lambda \cup DiscreteEdge(q, X_f, \frac{u_i+u_j}{2}, u_j, h)$; **fin si**
 - 8: **sinon**
 - 9: {Cas (b) ou (c), figure 5.13}
 - 10: **si** $F_i = \emptyset$ ou $X \neq \emptyset$ **alors** $\Lambda := \Lambda \cup DiscreteEdge(q, X_f, u_i, \frac{u_i+u_j}{2}, h)$; **fin si**
 - 11: **si** $F_j = \emptyset$ ou $X \neq \emptyset$ **alors** $\Lambda := \Lambda \cup DiscreteEdge(q, X_f, \frac{u_i+u_j}{2}, u_j, h)$; **fin si**
 - 12: **fin si**
 - 13: **fin si**
 - 14: Retourne Λ ;
-

Exemple 5.3.2 (En dimension 2). On considère le système linéaire local suivant :

$$\dot{X}(t) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} X(t) + \begin{bmatrix} -1 & -2 \\ -3 & -1 \end{bmatrix} u(t),$$

défini dans la cellule $D_q = Conv\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right)$, sous la contrainte :

$$\forall t, u(t) \in \mathbb{U}_2 = Conv\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right).$$

La figure 5.14 montre les approximations convexes contrôlables calculées dans la cellule D_q pour différents pas de discrétisation. On constate ainsi que pour une précision donnée, l'algorithme 8 fournit des résultats d'excellente qualité, bien meilleurs que ceux obtenus par l'algorithme précédent.

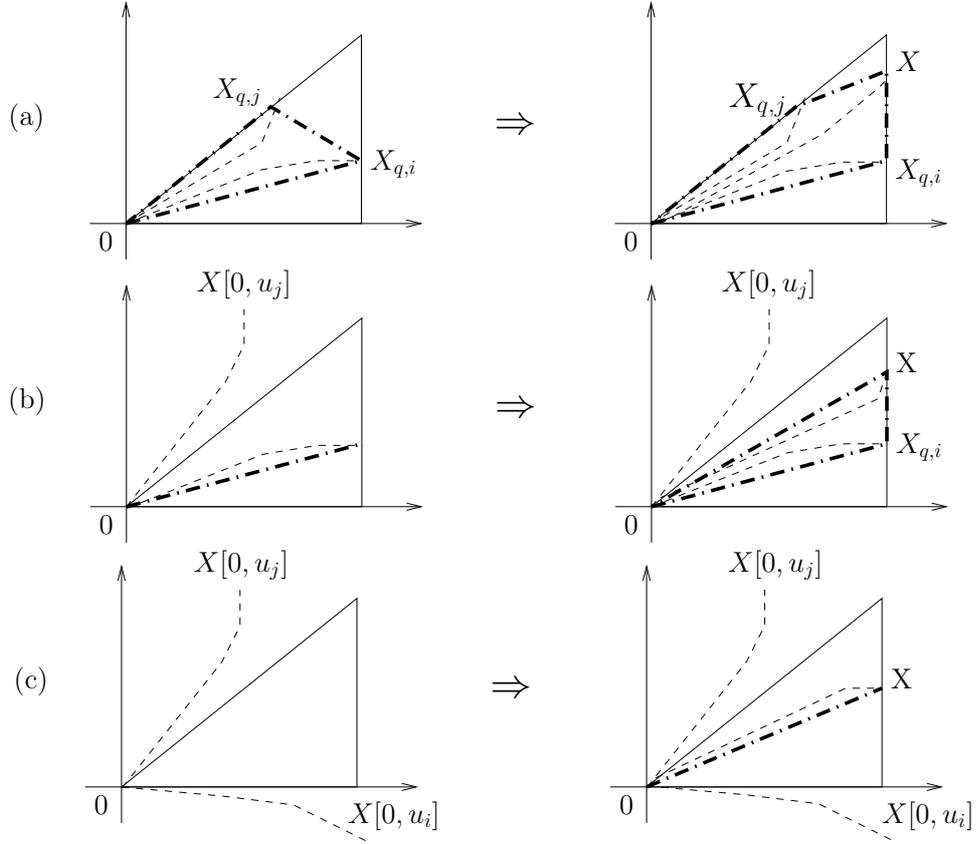


FIG. 5.13 – Amélioration de l'approximation convexe du domaine contrôlable hybride dans une cellule

5.3.3 Approximation convexe sur un chemin

Considérons maintenant une suite $\gamma = (q_i)_{i=0\dots r}$ de modes discrets de l'automate hybride \mathcal{H} telle que :

$$\forall i \in \{0, \dots, r-1\}, G_{(q_i, q_{i+1})} \neq \emptyset \text{ et } 0 \in D_{q_0},$$

ce qui revient à considérer une suite de cellules adjacentes de \mathbb{R}^n , contenant la cible 0. On introduit le domaine :

$$\Omega_\gamma = \bigcup_{i=0}^r D_{q_i}$$

défini comme la réunion des cellules du chemin γ . Grâce à l'algorithme 7 de calcul d'une approximation convexe dans un mode, nous voulons maintenant proposer un algorithme d'approximation du domaine contrôlable dans Ω_γ .

Reprenons pour cela la démarche décrite au chapitre 4 : d'après la proposition 4.3.2, le domaine contrôlable du modèle hybride \mathcal{H} dans Ω_γ est approché de façon récursive par l'union de domaines contrôlables définis successivement à l'intérieur des cellules D_{q_i} du chemin :

$$\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma) = \mathcal{C}_{\mathcal{H}}(\tau_0, D_{q_0}) \cup \mathcal{C}_{\mathcal{H}}(\tau_1, D_{q_1}) \cup \dots \cup \mathcal{C}_{\mathcal{H}}(\tau_r, D_{q_r}) \subset \mathcal{C}_{\mathcal{H}}(\Omega_\gamma),$$

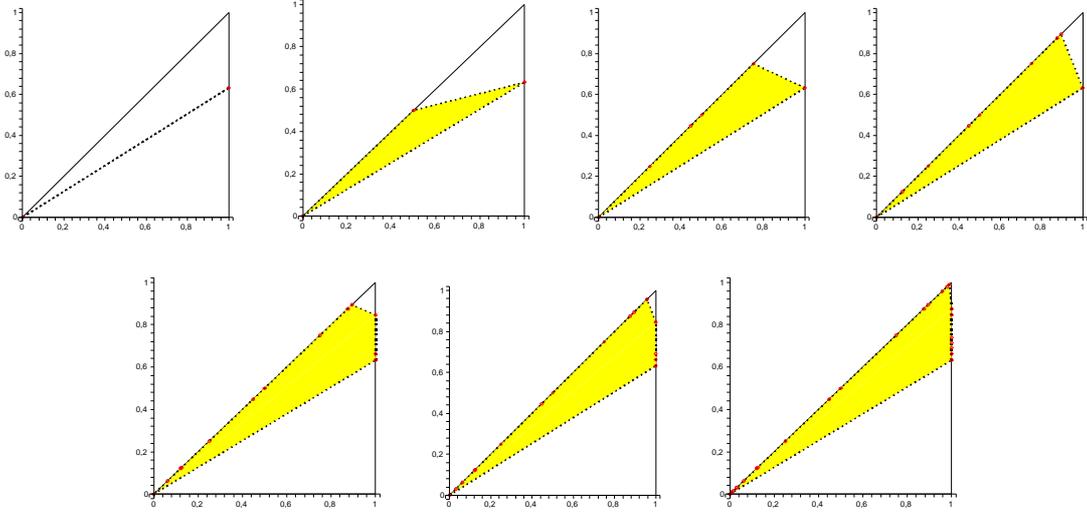


FIG. 5.14 – Approximation convexe contrôlable calculée par l'algorithme 8 pour différentes valeurs du pas de discrétisation : $h \in \{2, 1, 0.5, 0.25, 0.1, 0.05, 0.01\}$

sachant que τ_i ($i = 0 \dots, r$) désigne la cible à atteindre localement dans la cellule D_{q_i} . Selon le même schéma nous pouvons alors définir une approximation $\Lambda(\Omega_\gamma)$ du domaine contrôlable hybride $\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma)$ par approximations convexes locales successives :

$$\Lambda(\Omega_\gamma) = \Lambda(\tilde{\tau}_0, D_{q_0}) \cup \Lambda(\tilde{\tau}_1, D_{q_1}) \cup \dots \cup \Lambda(\tilde{\tau}_r, D_{q_r}),$$

où :

- $\Lambda(\tilde{\tau}_i, D_{q_i})$ désigne l'approximation contrôlable calculée par l'algorithme 7.
- $(\tilde{\tau}_i)_{i=0 \dots r}$ est la suite des cibles le long du chemin γ , définie par récurrence de la façon suivante :

$$\begin{cases} \tilde{\tau}_0 = \{0\} \\ \tilde{\tau}_{i+1} = \Lambda(\tilde{\tau}_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}, \quad i = 0, \dots, r \end{cases} \quad (5.17)$$

Par récurrence, on montre facilement que, quel que soit $i = 0, \dots, r$, la cible approchée $\tilde{\tau}_i$ est un polyèdre convexe, i.e. un polytope.

En d'autres termes, nous commençons par calculer une approximation convexe du domaine contrôlable dans l'état q_0 qui contient la cible (i.e. $0 \in D_{q_0}$) grâce à l'algorithme 7. Ensuite, l'intersection de cette première approximation avec la garde $G_{(q_0, q_1)}$ entre les états q_0 et q_1 définit une nouvelle cible à atteindre dans l'état q_1 , ce qui nous permet de poursuivre l'approximation de la même façon (voir figure 5.15).

L'algorithme se termine lorsque l'intersection de l'approximation courante avec la cellule suivante du chemin est vide, ou alors lorsque l'on a atteint le dernier état du chemin (ici q_r).

Nous pouvons alors vérifier que chaque cible intermédiaire $\tilde{\tau}_i$ est une approximation convexe de la cible τ_i , ce qui nous permet de démontrer pas à pas que $\Lambda(\tilde{\tau}_i, D_{q_i})$ est bien une approximation convexe du domaine contrôlable $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$:

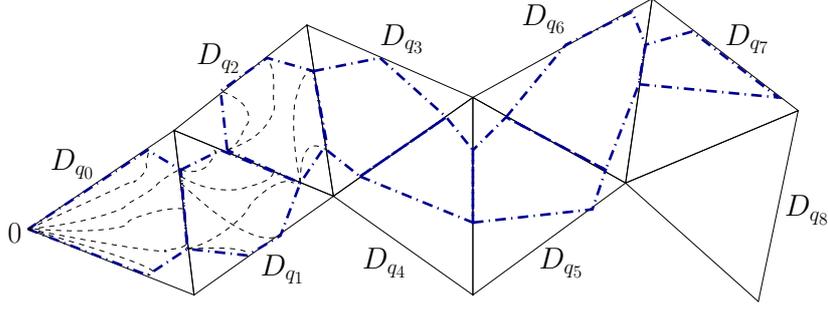


FIG. 5.15 – Construction d’une approximation convexe du domaine contrôlable sur une suite donnée $(q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8)$ de modes discrets de l’automate \mathcal{H} .

(Trajectoires par renversement du temps en - - - - - Approximation convexe contrôlable en - · - · -)

Proposition 5.3.2. *Les cibles $\tilde{\tau}_i$, $i = 0, \dots, r$, définies par la relation de récurrence (5.17), sont des approximations convexes respectives des cibles τ_i , $i = 0, \dots, r$. Quel que soit $i = 0, \dots, r$, $\Lambda(\tilde{\tau}_i, D_{q_i})$ est une approximation convexe du domaine contrôlable $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$ dans le mode q_i .*

Si, de plus, les domaines $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$, $i = 0, \dots, r$, sont convexes alors :

$$\forall i = 0, \dots, r, \begin{cases} \tilde{\tau}_i \subset \tau_i \\ \Lambda(\tilde{\tau}_i, D_{q_i}) \subset \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \end{cases}$$

Preuve. Supposons la relation suivante vérifiée (cf paragraphe 5.2) :

$$\forall q \in \mathcal{Q}, [\tau \text{ convexe} \Rightarrow \mathcal{C}_{\mathcal{H}}(\tau, D_q) \text{ convexe}].$$

Notons

$$\mathcal{P}(i) : \tilde{\tau}_i \subset \tau_i \text{ et } \Lambda(\tilde{\tau}_i, D_{q_i}) \subset \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$$

la propriété à démontrer pour $i = 0, \dots, r$. On procède par récurrence sur l’indice i :

Au rang $i = 0$: par définition, $\tilde{\tau}_0 = \tau_0 = \{0\}$. D’autre part, d’après la proposition 5.3.1, $\Lambda(\{0\}, D_{q_0})$ est une sous-approximation du domaine contrôlable $\mathcal{C}_{\mathcal{H}}(\{0\}, D_{q_0})$. Donc $\mathcal{P}(0)$ est vraie.

Supposons $\mathcal{P}(i)$ vraie, i.e. : $\tilde{\tau}_i \subset \tau_i$ et $\Lambda(\tilde{\tau}_i, D_{q_i}) \subset \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$.

D’après la construction des cibles $\tilde{\tau}_j$ et τ_j , nous avons les relations suivantes :

$$\tilde{\tau}_{i+1} = \Lambda(\tilde{\tau}_i, D_{q_i}) \cap G_{(q_i, q_{i+1})} \text{ et } \tau_{i+1} = \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}$$

Or, par hypothèse de récurrence, on a : $\Lambda(\tilde{\tau}_i, D_{q_i}) \subset \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$, d’où : $\tilde{\tau}_{i+1} \subset \tau_{i+1}$. D’après la proposition 5.3.1, la sous-approximation $\Lambda(\tilde{\tau}_{i+1}, D_{q_{i+1}})$ vérifie :

$$\Lambda(\tilde{\tau}_{i+1}, D_{q_{i+1}}) \subset \mathcal{C}_{\mathcal{H}}(\tilde{\tau}_{i+1}, D_{q_{i+1}}).$$

De plus, par définition du domaine contrôlable, on a :

$$\tilde{\tau}_{i+1} \subset \tau_{i+1} \Rightarrow \mathcal{C}_{\mathcal{H}}(\tilde{\tau}_{i+1}, D_{q_{i+1}}) \subset \mathcal{C}_{\mathcal{H}}(\tau_{i+1}, D_{q_{i+1}}),$$

ce qui implique le résultat cherché : $\Lambda(\tilde{\tau}_{i+1}, D_{q_{i+1}}) \subset \mathcal{C}_{\mathcal{H}}(\tau_{i+1}, D_{q_{i+1}})$. La propriété $\mathcal{P}(i+1)$ est donc bien vérifiée.

En conclusion, nous avons démontré que $\mathcal{P}(0)$ est vraie et que, pour tout $i \in \mathbb{N}$, si la propriété \mathcal{P} est vraie au rang i , elle l'est également au rang $i+1$. Elle est donc vraie quel que soit $i = 0, \dots, r$.

□

Algorithme 9 Approximation convexe contrôlable sur un chemin de cellules

Données : \mathcal{H} , $\gamma = (q_i)_{i=0\dots r}$ une suite d'états discrets de l'automate \mathcal{H} .

Sortie : Une approximation convexe du domaine contrôlable dans $\bigcup_{i=0}^r D_{q_i}$.

- 1: Approximation initiale : $\Lambda := \emptyset$;
 - 2: Cible : $\tau := \{0\}$;
 - 3: {Tant que la cible n'est pas vide, on parcourt la suite γ des états discrets,}
 - 4: $i := 0$;
 - 5: **tant que** $\tau \neq \emptyset$ et $i \leq r$ **faire**
 - 6: On calcule une approximation convexe du domaine contrôlable vers la cible τ dans l'état q_i grâce à l'algorithme 7 :
 $\Lambda := \Lambda \cup \text{ConvexApproximation}(\mathcal{H}, q_i, \tau)$;
 - 7: {On réinitialise la cible τ à l'ensemble des sommets de l'intersection de l'approximation courante avec la garde $G_{(q_i, q_{i+1})}$:}
 $\tau := \text{Sommets}(\Lambda \cap G_{(q_i, \text{succ}_{\gamma}(q_i))})$
 - 8: {On passe à la cellule suivante : } $i := i + 1$;
 - 9: **fin tant que**
 - 10: Retourner Λ .
-

Exemple 5.3.3 (Exemple du ressort non linéaire [81, paragraphe 7.3.1]). *Dans cet exemple, on s'intéresse au contrôle d'un ressort non linéaire astreint à se déplacer le long d'un axe horizontal, jusqu'à sa position de repos en temps minimum. Le mouvement du ressort est modélisé par le système différentiel suivant :*

$$\begin{cases} \dot{x}(t) = y(t) \\ \dot{y}(t) = -x(t) - 2x(t)^3 + u(t) \end{cases} \quad (5.18)$$

(se reporter au chapitre 1 de [81] pour une description complète du modèle général du ressort).

Le contrôle u représente la force extérieure horizontale appliquée au ressort et est donc borné, choisi de sorte que : $\forall t > 0, |u(t)| \leq 1$. Le problème qui nous intéresse est de calculer l'ensemble des positions initiales $(x_0, y_0 = \dot{x}_0)$ à partir desquelles le ressort peut atteindre sa position de repos supposée à l'origine $(0, 0)$.

Les figures 5.16 et 5.3.3 présentent les différents résultats obtenus grâce l'algorithme 9 d'approximation convexe pour deux pas $h > 0$ différents. Comme prévu, l'approximation contrôlable calculée pour un pas de maillage plus fin, est de bien meilleure qualité.

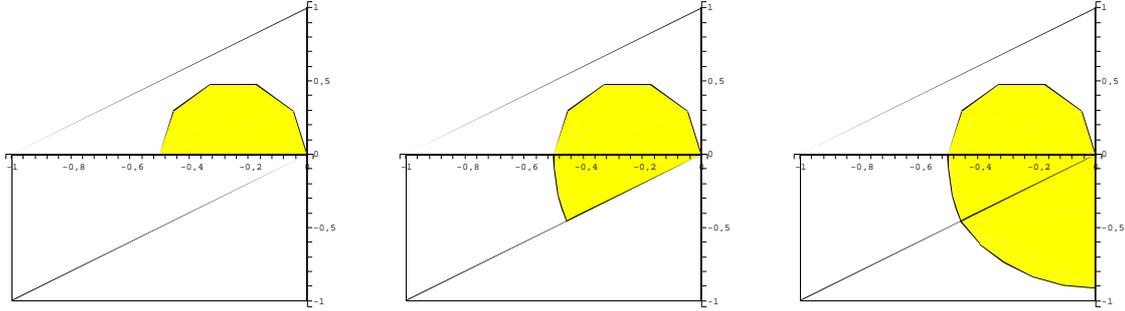


FIG. 5.16 – Approximations contrôlables successives sur trois cellules D_{q_1} , D_{q_2} et D_{q_3} pour un pas $h = 1$ de discrétisation pour le problème du ressort non linéaire.

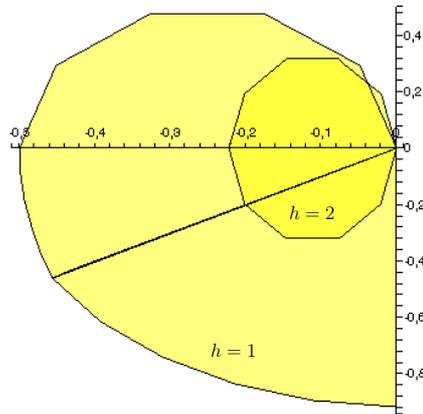


FIG. 5.17 – Approximations contrôlables aux échelles $h = 2$ et $h = 1$ pour le problème du ressort non linéaire.

Exemple 5.3.4 (Un exemple académique en dimension 3). On considère maintenant la famille de systèmes :

$$\dot{X}(t) = \begin{bmatrix} q & 0 & q \\ 0 & 3q & 0 \\ 0 & q & q \end{bmatrix} X(t) + \begin{bmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -1 \end{bmatrix} u(t), \quad q \in \mathbb{N},$$

sous la contrainte :

$$\forall t, u(t) \in \text{Conv} \left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right).$$

On applique alors l'algorithme 8 dans deux modes $q = 0$, puis $q = 6$, respectivement associés à deux cubes adjacents dans l'espace d'état \mathbb{R}^3 . Cet exemple illustre l'utilisation des algorithmes

présentés dans ce chapitre à un automate hybride affine par morceaux défini sur des cubes et non sur des simplexes.

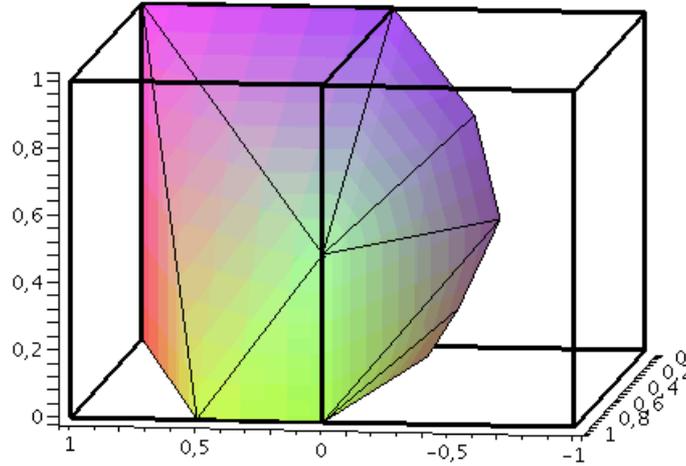


FIG. 5.18 – Approximation convexe contrôlable en dimension 3 pour l'exemple 5.3.4.

5.4 Validité de l'approximation convexe

Pour conclure ce chapitre, nous proposons tout d'abord une étude de l'erreur d'approximation commise par l'algorithme 9 puis quelques conclusions quant à la contrôlabilité d'un point donné.

5.4.1 Étude de l'erreur d'approximation

Soit $\gamma = (q_i)_{i=0\dots r}$ un chemin donné de cellules adjacentes dans l'espace d'état et $\Omega_\gamma = \bigcup_{i=0}^r D_{q_i}$ le domaine associé. Dans cette partie, nous cherchons à évaluer la distance du domaine contrôlable $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$ du système hybride à son approximation convexe $\Lambda(\Omega_\gamma)$.

D'après les propositions 4.3.2 et 4.2.1, il suffit d'évaluer l'approximation dans chaque cellule D_{q_i} du chemin considéré, pour obtenir une évaluation de l'approximation sur Ω_γ . En effet, d'après la proposition 4.3.2, les ensembles $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$ et $\Lambda(\Omega_\gamma)$ s'écrivent respectivement :

$$\mathcal{C}_{\mathcal{H}}(\Omega_\gamma) = \mathcal{C}_{\mathcal{H}}(\tau_0, D_{q_0}) \cup \mathcal{C}_{\mathcal{H}}(\tau_1, D_{q_1}) \cup \dots \cup \mathcal{C}_{\mathcal{H}}(\tau_r, D_{q_r})$$

$$\Lambda(\Omega_\gamma) = \Lambda(\tilde{\tau}_0, D_{q_0}) \cup \Lambda(\tilde{\tau}_1, D_{q_1}) \cup \dots \cup \Lambda(\tilde{\tau}_r, D_{q_r})$$

où les cibles successives $\tau = (\tau_i)_{i=0,\dots,r}$ et $\tilde{\tau} = (\tilde{\tau}_i)_{i=0,\dots,r}$ sont définies récursivement de la

façon suivante (cf proposition 4.3.2) :

$$\begin{cases} \tau_0 = \tilde{\tau}_0 = \{0\} \\ \tau_{i+1} = \mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}, & i = 0, \dots, r-1 \\ \tilde{\tau}_{i+1} = \Lambda(\tilde{\tau}_i, D_{q_i}) \cap G_{(q_i, q_{i+1})}, & i = 0, \dots, r-1 \end{cases}$$

En conséquence, si, pour tout $i \in \{0, \dots, r\}$, on connaît une majoration de la distance du domaine contrôlable dans la cellule D_{q_i} à son approximation :

$$d_H(\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i}), \Lambda(\tilde{\tau}_i, D_{q_i})) \leq \delta_i,$$

alors, grâce à la proposition 4.2.1, on peut en déduire une majoration de la distance d'Hausdorff entre le domaine contrôlable le long de γ et son approximation :

$$d_H(\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma), \Lambda(\Omega_\gamma)) \leq \max_{i=0, \dots, r} \delta_i.$$

De plus quelle que soit la cellule D_{q_i} considérée, la distance du domaine contrôlable $\mathcal{C}_{\mathcal{H}}(\tau_i, D_{q_i})$ à l'approximation convexe $\Lambda(\tilde{\tau}_i, D_{q_i})$ est majorée par la taille du maillage \mathcal{D} de l'espace d'état, d'où d'après les résultats du paragraphe 3.1.2 :

$$d_H(\underline{\mathcal{C}}_{\mathcal{H}}(\Omega_\gamma), \Lambda(\Omega_\gamma)) \leq \sqrt{nh}.$$

5.4.2 Contrôlabilité d'un point initial donné

Dans ce chapitre, nous avons proposé un algorithme permettant de calculer une approximation convexe du domaine contrôlable défini dans une suite finie de modes discrets de l'automate hybride considéré. Revenons maintenant à la question de la contrôlabilité d'un point donné par un système hybride.

Soit $X_0 \in \mathbb{R}^n$ un point donné de l'espace d'état. On considère un automate hybride \mathcal{H} affine par morceaux et un chemin γ de modes discrets adjacents, arbitrairement choisi de sorte que la cible 0 et le point initial X_0 appartiennent au domaine Ω_γ associé.

On rappelle qu'à tout chemin fini de modes discrets $\gamma = (q_i)_{i=0 \dots r}$ contenant la cible 0, on associe un domaine compact Ω_γ de l'espace d'état défini par :

$$\Omega_\gamma = \bigcup_{i=0}^r D_{q_i}$$

et un domaine contrôlable noté $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$ qui est l'ensemble des points de Ω_γ à partir desquels le système \mathcal{H} peut atteindre la cible sans sortir de Ω_γ .

Dans le cas où les domaines contrôlables locaux sont convexes, l'algorithme 9 présenté dans chapitre nous permet de tester directement la contrôlabilité du point X_0 dans Ω_γ . En effet, le domaine $\Lambda(\Omega_\gamma)$ calculé étant dans ces conditions une sous-approximation du domaine contrôlable hybride $\mathcal{C}_{\mathcal{H}}(\Omega_\gamma)$, il suffit de tester l'appartenance de X_0 à $\Lambda(\Omega_\gamma)$; par construction, X_0 est alors contrôlable jusqu'en 0. Dans le cas contraire, nous ne pouvons pas encore conclure.

Une solution serait alors de combiner sous- et sur-approximation du domaine contrôlable $\mathcal{C}_{\mathcal{H}}(\Omega_{\gamma})$ dans Ω_{γ} . Ainsi si le point X_0 est au delà de la sur-approximation, nous pouvons garantir qu'il n'est pas contrôlable dans le chemin de cellules considéré. L'erreur commise est alors majorée par la distance de Hausdorff entre la sous- et la sur-approximation contrôlables.

L'approche multi-échelle proposée dans le paragraphe 4.2.3 devrait permettre de localiser un chemin γ de modes de l'automate \mathcal{H} de sorte que si le point X_0 considéré est contrôlable, alors il est contrôlable dans le chemin γ .

Dans ce chapitre, nous avons développé des algorithmes efficaces de calcul d'une approximation convexe du domaine contrôlable du modèle hybride présenté au chapitre 3. Remarquons que d'après l'algorithme 7, la cible considérée peut être un point comme c'est le cas pour le problème de contrôle considéré ou n'importe quel polytope défini par la donnée de ses sommets. Les algorithmes présentés permettent ainsi le calcul approché du domaine contrôlable selon deux types de propagation (cf chapitre 4) mais aussi par renversement du temps, des ensembles atteignables du modèle hybride affine par morceaux

Troisième partie

Recherche de solutions optimales

Cette partie est consacrée à la résolution algorithmique des problèmes de contrôles non linéaires grâce au modèle hybride défini dans la partie I. On considère un système de contrôle non linéaire général de la forme :

$$\begin{cases} \dot{X}(t) &= f(X(t), u(t)) \\ X(0) &= X_0 \end{cases}$$

et une cible à l'origine 0 de l'espace d'état. On suppose que $(0, 0)$ est un point d'équilibre du système i.e. : $f(0, 0) = 0$. Le point initial X_0 considéré est supposé contrôlable : il existe donc au moins une trajectoire du système de contrôle non linéaire qui relie X_0 à la cible en temps fini.

On introduit maintenant un critère de coût le long d'une trajectoire. Ce coût peut être par exemple le temps mis par le système pour aller de X_0 à la cible ou encore son énergie. Le problème de contrôle optimal s'exprime alors de la façon suivante : parmi l'ensemble des trajectoires admissibles qui relient X_0 à la cible, on veut sélectionner celle(s) qui minimise(nt) le coût choisi.

Grâce à la modélisation hybride proposée dans la partie I, le problème de contrôle optimal non linéaire est approché par un problème de contrôle optimal hybride. Un point clé traité dans cette partie consiste à s'assurer que les solutions optimales du problème de contrôle hybride approchent effectivement celles du problème initial. Dans un second temps, il s'agit de calculer les solutions optimales du problème hybride.

Cette partie présente différentes approches de la résolution des problèmes de contrôle optimal. Le chapitre 6 est consacré à l'étude de l'approximation des solutions optimales du problème de contrôle non linéaire par les solutions optimales du problème hybride qui le modélise. Cette étude nous conduit en particulier à étudier une approche des problèmes de contrôle par les équations d'Hamilton-Jacobi-Bellman. Dans le chapitre 7, nous proposons une approche globale des problèmes de contrôle optimal hybrides basée sur un principe du maximum hybride.

Chapitre 6

Approche hybride pour le contrôle optimal des systèmes non linéaires

Nous nous intéressons dans ce chapitre aux méthodes de résolution des problèmes de contrôle optimal non linéaires présentés au chapitre 1. Nous considérons un système non linéaire sous la forme :

$$\dot{X}(t) = f(X(t), u(t)). \quad (6.1)$$

Soit $X_0 \in \mathbb{R}^n$ un point donné de l'espace d'état, supposé contrôlable par le système (6.1) jusqu'en 0. Parmi l'ensemble des trajectoires admissibles du système considéré reliant X_0 à 0, on veut sélectionner celle(s) qui amène(nt) le système (6.1) de la position initiale X_0 à la cible 0 en minimisant un coût $J(X_0, u)$ donné.

La théorie du contrôle optimal a connu un véritable essor depuis les années cinquante avec la découverte d'outils puissants tels que le principe de programmation dynamique de R. Bellman [9] ou le principe du maximum de Pontriaguine [70]. Ces résultats jouent un rôle essentiel en théorie du contrôle optimal des systèmes non linéaires et ont donné naissance à deux approches différentes des problèmes de contrôle optimal (cf chapitre 1 pour une synthèse des méthodes numériques usuellement employées dans chacune des approches présentées ci-après).

La première approche est basée sur l'utilisation du principe du maximum de Pontriaguine [70, 42] qui fournit une formulation hamiltonienne des problèmes de contrôle ainsi que des conditions nécessaires d'optimalité. Une famille importante de méthodes numériques issues de cette approche est constituée des méthodes de tir indirectes. Ces méthodes très efficaces en toute dimension peuvent être difficiles à mettre en oeuvre d'un point de vue théorique et demandent une connaissance a priori de la structure de la trajectoire optimale.

La seconde approche repose sur le principe de programmation dynamique de R. Bellman [9] et la caractérisation de la fonction coût $J(X_0, u)$ en termes de solution de viscosité d'une équation d'Hamilton-Jacobi-Bellman (HJB) non linéaire [26, 8, 7]. Les méthodes numériques de résolution de l'équation (HJB) appliquent des schémas de discrétisation en temps et/ou en espace [7, 10, 18] et sont donc généralement simples à mettre en oeuvre. Malheureusement, ces algorithmes efficaces en petite dimension demandent un coût trop élevé en mémoire pour être applicables en grande dimension.

La méthode que nous proposons dans ce chapitre consiste à remplacer la résolution du problème de contrôle optimal non linéaire qui nous intéresse par celle d'un problème hybride dont les solutions approchent celles du problème initial.

D'après l'étude menée dans la partie I de ce manuscrit, tout système de contrôle non linéaire peut être approché par un système hybride affine par morceaux. Cette modélisation nous amène naturellement à définir deux problèmes de contrôle optimal respectivement associés au système non linéaire et à son approximation hybride. L'idée que nous défendons est alors que la résolution du problème hybride permet une résolution approchée du problème non linéaire initial.

Ce chapitre est consacré à la justification de l'approche hybride pour la résolution des problèmes de contrôle optimal non linéaires. Après avoir succinctement présenté des résultats d'existence de solutions optimales, nous proposons deux approches pour la résolution des problèmes de contrôle non linéaire. La première s'appuie sur le principe du maximum nous amenant ainsi à énoncer un principe du maximum hybride adapté à notre modélisation. La seconde se place dans le contexte de la théorie des solutions de viscosité des équations d'Hamilton-Jacobi-Bellman. Nous terminons ce chapitre par une étude algorithmique des problèmes de contrôle linéaires grâce au principe du maximum de Pontriaguine.

6.1 Conditions d'existence de trajectoires optimales

Dans ce chapitre, nous considérons un système de contrôle très général de la forme :

$$\dot{X}(t) = f(X(t), u(t)), \quad (6.2)$$

où les contrôles admissibles u sont des fonctions mesurables bornées à valeurs dans un compact \mathbb{U}_m de \mathbb{R}^m . On rappelle que \mathbb{U}_m est un polytope de \mathbb{R}^m défini par la donnée de ses sommets :

$$\mathbb{U}_m = \text{Conv}(s_1, \dots, s_p)$$

et contenant l'origine 0 de \mathbb{R}^m . On suppose également que le système (6.2) admet un point d'équilibre en $(0, 0)$ i.e. que : $f(0, 0) = 0$. On introduit ensuite une fonction coût de la forme :

$$J(X_0, u) = \int_0^{t_f} l(X(t), u(t)) dt, \quad (6.3)$$

où la fonction l est appelée coût instantané et t_f est le temps mis par la trajectoire issue de X_0 selon le contrôle u pour atteindre une cible donnée.

Étant donné un point initial X_0 et une cible $X_f = 0$ dans l'espace d'état, le problème de contrôle associé au système (6.2) et au coût (6.3) consiste à déterminer des trajectoires du système (6.2) qui relient X_0 à 0 en un coût minimum. Le problème est dit en horizon infini si le temps final t_f mis pour atteindre la cible n'est pas fixé.

Avant de s'intéresser aux méthodes de résolution de ce type de problème, la première question que nous nous posons est celle de l'existence de trajectoires optimales du système (6.2) par rapport au coût considéré.

Existence de trajectoires optimales

Remarquons tout d'abord que si le point initial X_0 n'est pas contrôlable, alors il n'existe pas de contrôle u admissible ni donc de trajectoire du système considéré reliant X_0 à la cible. Une première condition nécessaire d'existence de trajectoires optimales du problème (6.2)-(6.3) est donc la contrôlabilité à la cible du point initial X_0 .

Sous cette condition, parmi l'ensemble des trajectoires admissibles du système (6.2), on veut sélectionner celles, si elles existent, qui minimisent le critère (6.3). L'existence de tels contrôles dépend fortement des régularités du champ non linéaire f et du coût instantané l .

Le théorème énoncé ci-dessous fournit un résultat d'existence de solutions optimales entre deux sous-ensembles de l'espace d'état et repose sur la compacité des ensembles accessibles en temps fini T , cf [1, chapitre 10], [81, chapitres 5 et 6].

Théorème 6.1.1 ([61, chapitre 4, th 4],[81, chapitre 6, th 6.2.1]). *On suppose f et l de classe C^1 sur $\mathbb{R}^n \times \mathbb{R}^m$. On peut éventuellement avoir des contraintes sur l'état de la forme :*

$$c_1(X) \geq 0, \dots, c_r(X) \geq 0,$$

que l'on suppose continues sur \mathbb{R}^n le cas échéant. Soient M_0 et M_1 deux compacts de \mathbb{R}^n tels que M_1 est accessible depuis M_0 . On note \mathfrak{U} l'ensemble des contrôles admissibles joignant M_0 à M_1 et $t_f(u)$ le temps mis par le système pour aller de M_0 à M_1 selon le contrôle $u \in \mathfrak{U}$.

On suppose alors que :

- i. *il existe un réel positif b tel que toute trajectoire associée à un contrôle $u \in \mathfrak{U}$ et notée X_u , est uniformément bornée par b sur $[0, t_f(u)]$, i.e. :*

$$\exists b > 0, \forall u \in \mathfrak{U}, \forall t \in [0, t_f(u)], \|X_u(t)\| \leq b$$

- ii. *Pour tout $X \in \mathbb{R}^n$, l'ensemble $\tilde{V}(X) = \{(l(X, u), f(X, u)) ; u \in \mathbb{U}_m\}$ est convexe*

Alors il existe un contrôle optimal u défini sur $[0, t_f(u)]$ tel que la trajectoire associée relie M_0 à M_1 en temps $t_f(u)$ et en coût minimum.

Des conditions d'existence de trajectoires optimales plus générales peuvent être obtenues [14, 55, 61]. Dans la suite de ce chapitre, nous nous plaçons sous les hypothèses du théorème 6.1.1 garantissant ainsi l'existence de solutions optimales pour le problème de contrôle non linéaire (6.2)-(6.3). Nous supposons également :

$$\left\{ \begin{array}{l} f \text{ de classe } C^1 \text{ sur } \mathbb{R}^n \times \mathbb{U}_m \\ f \text{ lipschitzienne par rapport à } X, \text{ uniformément par rapport à } u \text{ i.e. :} \\ \exists L_1 > 0, \forall (X_1, X_2) \in (\mathbb{R}^n)^2, \forall u \in \mathbb{U}_m, \|f(X_1, u) - f(X_2, u)\| \leq L_1 \|X_1 - X_2\| \end{array} \right. \quad (6.4)$$

$$\left\{ \begin{array}{l} l \text{ de classe } C^1 \text{ sur } \mathbb{R}^n \times \mathbb{U}_m \\ l \text{ lipschitzienne par rapport à } X, \text{ uniformément par rapport à } u \text{ i.e. :} \\ \exists L_2 > 0, \forall (X_1, X_2) \in (\mathbb{R}^n)^2, \forall u \in \mathbb{U}_m, |l(X_1, u) - l(X_2, u)| \leq L_2 \|X_1 - X_2\| \end{array} \right. \quad (6.5)$$

Une fois obtenue l'existence de solutions optimales, nous nous intéressons à la recherche de solutions du problème de contrôle optimal non linéaire grâce au modèle hybride introduit au chapitre 3.

Dans la suite, les notations (\mathcal{P}_{NL}) et $(\mathcal{P}_{\mathcal{H}})$ désignent respectivement les problèmes de contrôle optimal linéaire et hybride définis de la façon suivante :

(\mathcal{P}_{NL}) Minimiser la fonction coût $J(X_0, u) = \int_0^{t_f} l(X(t), u(t))dt$ par rapport au contrôle u sous la contrainte :

$$\begin{cases} \dot{X}(t) &= f(X(t), u(t)) \\ X(0) &= X_0 \end{cases}, \quad X(t_f) = 0$$

sachant que : $\forall t \geq 0, u(t) \in \mathbb{U}_m \subseteq \mathbb{R}^m$. Le temps final t_f n'est pas fixé.

$(\mathcal{P}_{\mathcal{H}})$ Minimiser la fonction coût $J_h(X_0, u) = \int_0^{t_f} l(X_h(t), u(t))dt$ par rapport au contrôle u sous la contrainte :

$$\begin{cases} \dot{X}_h(t) &= f_h(X_h(t), u(t)) \\ X_h(0) &= X_0 \end{cases}, \quad X_h(t_f) = 0$$

sachant que : $\forall t \geq 0, u(t) \in \mathbb{U}_m \subseteq \mathbb{R}^m$. Le temps final t_f n'est pas fixé.

Nous rappelons que f_h désigne l'approximation affine par morceaux du champ non linéaire f , construite sur un maillage de pas $h > 0$ de l'espace état-contrôle (cf partie I).

6.2 Première approche : principe du maximum de Pontriaguine

Nous nous intéressons maintenant à la résolution approchée du problème de contrôle optimal non linéaires (\mathcal{P}_{NL}) grâce au principe du maximum de Pontriaguine.

L'ensemble des contrôles admissibles manque en général de bonnes propriétés de compacité ce qui complique singulièrement la résolution pratique des problèmes de contrôle optimal. Un outil puissant pour rechercher les trajectoires optimales d'un système est le principe du maximum de Pontriaguine [70]. Ce théorème permet pour un système continu classique d'obtenir des conditions nécessaires d'ordre un d'optimalité ainsi qu'une formulation pseudo hamiltonienne du problème de contrôle optimal considéré.

Après avoir rappelé le principe du maximum de Pontriaguine, nous montrons dans cette partie que, grâce aux bonnes propriétés de régularité de notre modèle hybride, nous pouvons également énoncer un principe du maximum adapté au problème hybride $(\mathcal{P}_{\mathcal{H}})$. Ceci nous permet alors d'étudier la pertinence de l'approche hybride pour la résolution approchée du problème de contrôle optimal non linéaire (\mathcal{P}_{NL}) considéré.

6.2.1 Formulation hamiltonienne des problèmes de contrôle optimal

Commençons par rappeler le principe du maximum de Pontriaguine, ou plus exactement du minimum pour le problème qui nous intéresse, dans le cadre classique des systèmes lisses.

Le théorème 6.2.1 énoncé ci-après est une version forte du principe du maximum [70] prenant en compte les contraintes sur le contrôle. Les références concernant ce principe sont nombreuses ; citons en particulier [70, 61, 21, 1] pour une démonstration de ce théorème ou encore [12, paragraphe 1.5] et [81] pour une étude détaillée de son cadre d'application :

Théorème 6.2.1 (Principe du maximum de Pontriaguine). *On considère le système de contrôle :*

$$\dot{X}(t) = f(X(t), u(t)),$$

où le champ f est supposé de classe C^1 sur $\mathbb{R}^n \times \mathbb{R}^m$. Soit u un contrôle admissible associé à une trajectoire X qui transfère le système de la position X_0 au temps $t = 0$ à la cible $0 \in \mathbb{R}^n$ au temps final t_f non fixé. On introduit l'hamiltonien H du système :

$$H(X, u, \lambda, \lambda_0) = \lambda_0 l(X, u) + \lambda^T f(X, u).$$

Si le contrôle u est optimal sur l'intervalle de temps $[0, t_f]$, alors il existe une application non triviale $\lambda : [0, t_f] \rightarrow \mathbb{R}^n$ absolument continue, appelée vecteur adjoint, et un réel $\lambda_0 \geq 0$ tels que, pour presque tout $t \in [0, t_f]$:

- H atteint son minimum par rapport au contrôle au point $u(t)$, i.e. :

$$H(X(t), u(t), \lambda(t), \lambda_0) = \min_{v \in \mathbb{U}_m} H(X(t), v, \lambda(t), \lambda_0)$$

- $\dot{X}(t) = \frac{\partial H}{\partial \lambda}(X(t), u(t), \lambda(t), \lambda_0)$
- $\dot{\lambda}(t)^T = -\frac{\partial H}{\partial X}(X(t), u(t), \lambda(t), \lambda_0)$ (équation d'Euler-Lagrange)
- $H(X(t), u(t), \lambda(t), \lambda_0) = 0$ le long de la trajectoire optimale.

On appelle extrémale du problème de contrôle optimal tout quadruplet $(X, u, \lambda, \lambda_0)$ vérifiant les conditions du principe du maximum énoncé ci-dessus. Dans la suite de ce manuscrit, nous nous intéressons uniquement aux extrémales normales i.e. pour lesquelles λ_0 est non nul (ici supposé égal à 1). Nous écrirons donc en conséquence $H(X, u, \lambda)$ au lieu de $H(X, u, \lambda, \lambda_0)$.

Remarque 6.2.1. *La condition “ $H(X(t), u(t), \lambda(t)) = 0$ le long de la trajectoire optimale” provient d'une part du fait que le problème de contrôle considéré est autonome : en effet, les fonctions f et l ne dépendent pas explicitement du temps, H n'en dépend pas non plus et on a :*

$$\forall t \in [0, t_f], \min_{v \in \mathbb{U}_m} H(X(t), v, \lambda(t)) = Cte$$

D'autre part, le temps final est supposé libre, d'où la contrainte à $t = t_f$:

$$\min_{v \in \mathbb{U}_m} H(X(t_f), v, \lambda(t_f)) = 0$$

Remarque 6.2.2. *Dans le théorème 6.2.1, le champ f est supposé de classe C^1 sur $\mathbb{R}^n \times \mathbb{R}^m$. Cependant, soulignons le fait qu'il suffit en fait de supposer f continue et f continûment différentiable par rapport à la variable X pour démontrer le principe du maximum énoncé ci-dessus [70, 61, 71].*

On rappelle que si une variété M est définie par : $M = \{X \in \mathbb{R}^n; \varphi_1(X) = \dots = \varphi_r(X) = 0\}$ où les fonctions φ_i sont de classe C^1 sur \mathbb{R}^n , alors l'espace tangent à M au point $X_0 \in M$ est :

$$T_{X_0}M = \{v \in \mathbb{R}^n; \forall i \in \{1, \dots, r\}, \nabla \varphi_i(X_0).v = 0\}.$$

Le théorème 6.2.1 s'étend au cas où l'ensemble de départ et l'ensemble cible sont des variétés de \mathbb{R}^n . Il suffit de prendre en compte des conditions supplémentaires sur le vecteur adjoint, appelées conditions de transversalité :

Corollaire 6.2.1 (Conditions de transversalité [81]). *Dans le cas où l'ensemble de départ M_0 (resp. d'arrivée M_1) est une variété de \mathbb{R}^n ayant un espace tangent en $X(0)$ (resp. en $X(t_f)$), alors le théorème 6.2.1 s'applique et le vecteur adjoint est soumis à des conditions dites de transversalité :*

$$\lambda(0) \perp T_{X(0)}M_0 \quad (\text{resp. } \lambda(t_f) \perp T_{X(t_f)}M_1)$$

En optimisation classique, minimiser une fonction H par rapport à un paramètre u implique dans un premier temps de chercher les points stationnaires du système, i.e. les valeurs $u(X, \lambda)$ de u pour lesquelles :

$$\frac{\partial H}{\partial u}(X, u, \lambda) = 0,$$

puis d'étudier en ces points la positivité de la matrice hessienne :

$$\frac{\partial^2 H}{\partial u^2}(X, u, \lambda) = \begin{bmatrix} \frac{\partial^2 H}{\partial u_1^2}(X, u, \lambda) & \dots & \frac{\partial^2 H}{\partial u_m \partial u_1}(X, u, \lambda) \\ \vdots & & \vdots \\ \frac{\partial^2 H}{\partial u_1 \partial u_m}(X, u, \lambda) & \dots & \frac{\partial^2 H}{\partial u_m^2}(X, u, \lambda) \end{bmatrix}.$$

Notation. *Dans un soucis d'alléger les notations, on pose à partir de maintenant :*

$$H_u = \frac{\partial H}{\partial u} \quad \text{et} \quad H_{uu} = \frac{\partial^2 H}{\partial u^2}.$$

Nous introduisons alors la notion d'arc extrémal du système hamiltonien :

Définition 6.2.1 (Arc extrémal). *Une solution (X, u, λ) du système hamiltonien défini par le théorème de Pontriaguine est appelée extrémale ou arc extrémal du système.*

Cependant, dans certains problèmes d'optimisation, des extrémales peuvent exister même si la matrice H_{uu} est singulière. Il est alors nécessaire d'effectuer des tests supplémentaires afin de déterminer si ces extrémales dites *singulières* sont réellement optimales [60, 14].

Le cas le plus fréquent où des extrémales singulières apparaissent, est celui où l'hamiltonien du système de contrôle considéré dépend affinement du contrôle u , soit :

$$H(X, u, \lambda) = H_0(X, \lambda) + H_1(X, \lambda)u.$$

En effet, puisque H est affine par rapport au contrôle u , la relation $H_u = 0$ ne dépend pas de u et ne suffit donc plus à déterminer les contrôles optimaux éventuels. Nous entrons alors dans le cadre de la théorie des extrémales singulières étudiée dans la partie 6.4 de ce chapitre.

6.2.2 Énoncé d'un principe du maximum hybride

Dans le paragraphe précédent, nous avons énoncé le principe du maximum de Pontriaguine sous sa forme la plus régulière en supposant le champ de vecteur f de classe C^1 . Nous voulons maintenant adapter ce théorème aux systèmes hybrides affines par morceaux construits au chapitre 3 afin d'obtenir des conditions nécessaires d'optimalité pour le problème $(\mathcal{P}_{\mathcal{H}})$.

L'écriture d'un principe du maximum hybride est un problème délicat qui a suscité de nombreuses recherches. La principale difficulté est d'arriver à prendre en compte les discontinuités du système hybride considéré provoquées par les événements discrets et ensuite de les intégrer dans la formulation du problème d'optimisation. La théorie du contrôle optimal des systèmes hybrides manquant d'un cadre unifié, il existe à l'heure actuelle diverses formulations de ce principe du maximum hybride liées à la définition ainsi qu'à la régularité des systèmes hybrides considérés [77],[85, paragraphe 12.4.2]. Citons en particulier les travaux de P. Riedinger, C. Jung et F. Kratz qui énoncent et démontrent un principe du maximum hybride valable pour une classe très générale de systèmes hybrides [71, théorème 2.3].

Considérons un système hybride \mathcal{H} selon le modèle construit au chapitre 3 dont la dynamique est régie par le système différentiel :

$$\dot{X}(t) = f_h(X(t), u(t)).$$

Par construction, le champ f_h est affine par morceaux et continu sur $\mathbb{R}^n \times \mathbb{U}_m$ (cf proposition 2.1.1). Nous pouvons donc en déduire que f_h est non seulement continue sur $\mathbb{R}^n \times \mathbb{U}_m$ mais aussi différentiable par rapport à la variable X et sa différentielle est continue par morceaux. De plus par définition du système hybride \mathcal{H} , nous savons qu'il n'y a pas de ré-initialisation de la variable continue ce qui signifie qu'à tout instant de transition t_i entre deux modes q_i et q_{i+1} , nous avons :

$$X(t_i^+) = X(t_i^-).$$

Remarquons de plus qu'à cet instant de transition t_i , le système \mathcal{H} peut soit effectuer la transition discrète du mode q_i vers le mode q_{i+1} , soit rester dans le mode q_i . Ce choix dépend uniquement de la valeur du champ f_h au point $X(t_i)$. Les transitions entre les modes discrets dans notre modèle hybride sont dites non contraintes (ceci nous permet d'obtenir la continuité de l'état adjoint aux instants de transition dans le principe du maximum hybride).

En conséquence, d'après le théorème 2.3 donné dans [71], nous appliquons la version régulière du principe du maximum donnée par le théorème 6.2.1 dans chaque mode q de l'automate hybride \mathcal{H} en la complétant par des arguments de programmation dynamique permettant de prendre en compte les transitions discrètes d'un mode à un autre. Nous obtenons ainsi un principe du maximum adapté aux systèmes hybrides continus, affines par morceaux sous la forme suivante :

Théorème 6.2.2 (Principe du maximum hybride). *On introduit l'hamiltonien H_h associé au système hybride \mathcal{H} :*

$$H_h(X, u, \lambda) = l(X, u) + \lambda^T f_h(X, u).$$

Si le contrôle u est optimal sur l'intervalle de temps $[0, t_f]$, alors il existe une application non triviale $\lambda : [0, t_f] \rightarrow \mathbb{R}^n$ absolument continue par morceaux telle que :

- pour presque tout $t \in [0, t_f]$, H atteint son minimum par rapport au contrôle au point $u(t)$ i.e. :

$$H_h(X(t), u(t), \lambda(t)) = \min_{v \in \mathbb{U}_m} H_h(X(t), v, \lambda(t))$$

- $\dot{X}(t) = \frac{\partial H_h}{\partial \lambda}(X(t), u(t), \lambda(t))$
- $\dot{\lambda}(t)^T = -\frac{\partial H_h}{\partial X}(X(t), u(t), \lambda(t))$ (équation d'Euler-Lagrange)
- $H_h(X(t), u(t), \lambda(t)) = 0$ le long de la trajectoire optimale.
- A chaque instant de transition t_i entre deux modes q_i et q_{i+1} , les conditions de transversalité s'écrivent :

$$\begin{cases} \lambda(t_i^+) = \lambda(t_i^-) \\ H_h(X(t_i^+), u(t_i^+), \lambda(t_i^+)) = H_h(X(t_i^-), u(t_i^-), \lambda(t_i^-)) \end{cases}$$

Remarque 6.2.3. La continuité de l'hamiltonien H_h aux instants de transition est une conséquence du caractère autonome du modèle hybride considéré et de la non ré-initialisation de l'état continu X (cf conditions de transversalité (2.18) dans [71]).

Dans le cas où les ensembles de départ et d'arrivée sont des variétés de \mathbb{R}^n , les conditions de transversalité données par le corollaire 6.2.1 s'appliquent et s'ajoutent aux conditions d'optimalité données par ce principe du maximum hybride.

6.2.3 Étude de l'approximation hybride

Revenons maintenant aux problèmes de contrôle optimal non linéaire (\mathcal{P}_{NL}) et hybride ($\mathcal{P}_{\mathcal{H}}$), sachant que le système hybride \mathcal{H} est une approximation affine par morceaux et continue du système non linéaire initial considéré. Dans les paragraphes précédents, nous avons énoncé deux principes du maximum, le principe de Pontriaguine et un principe du maximum hybride, qui nous permettent d'obtenir des conditions nécessaires d'optimalité sur des couples de trajectoires état-contrôle (X, u) . Nous voulons maintenant montrer que la résolution du problème hybride par le principe du maximum associé permet une résolution approchée du problème non linéaire.

Nous définissons pour chacun des problèmes de contrôle considérés l'hamiltonien associé i.e. :

$$\begin{aligned} H(X, u, \lambda) &= l(X, u) + \lambda^T f(X, u) \\ H_h(X, u, \lambda) &= l(X, u) + \lambda^T f_h(X, u) \end{aligned}$$

Le premier résultat que nous obtenons est alors la convergence uniforme sur tout compact de l'hamiltonien H_h vers l'hamiltonien H :

Proposition 6.2.1. *Sous l'hypothèse (6.4), l'hamiltonien H_h converge uniformément vers H sur tout compact de $\mathbb{R}^n \times \mathbb{R}^n$, uniformément par rapport à u quand h tend vers 0^+ .*

Preuve. Par définition de H et H_h , nous avons :

$$|H(X, u, \lambda) - H_h(X, u, \lambda)| = |\lambda^T [f(X, u) - f_h(X, u)]| \leq \|\lambda^T\| \|f(X, u) - f_h(X, u)\|$$

Soit Ω_1 et Ω_2 des compacts de \mathbb{R}^n . Selon l'hypothèse (6.4), le champ f est de classe C^1 sur $\mathbb{R}^n \times \mathbb{R}^m$ ce qui nous permet d'appliquer la convergence uniforme de f_h vers f sur tout compact de \mathbb{R}^n , uniformément par rapport à u (cf proposition 2.3.1). Nous avons donc :

$$\sup_{(X,u) \in \Omega_1 \times \mathbb{U}_m} \|f(X, u) - f_h(X, u)\| \leq \varepsilon_{\Omega_1}(h)$$

où $\varepsilon_{\Omega_1}(h)$ désigne l'erreur d'interpolation sur $\Omega_1 \times \mathbb{U}_m$ et : $\lim_{h \rightarrow 0^+} \varepsilon_{\Omega_1}(h) = 0$. En posant :

$$M_2 = \sup_{\lambda \in \Omega_2} \|\lambda^T\| < \infty,$$

nous obtenons donc :

$$\sup_{(X,u,\lambda) \in \Omega_1 \times \mathbb{U}_m \times \Omega_2} |H(X, u, \lambda) - H_h(X, u, \lambda)| \leq M_2 \varepsilon_{\Omega_1}(h),$$

ce qui prouve la convergence uniforme de H_h vers H sur tous compacts de \mathbb{R}^n par rapport à X et λ , uniformément par rapport à u . □

Considérons maintenant un point contrôlable X_0 de l'espace d'état. Nous supposons également connue une trajectoire optimale (X, u) du système non linéaire (6.2) reliant X_0 à la cible en temps $T > 0$; d'après le principe du maximum de Pontriaguine, il existe donc un paramètre λ défini sur $[0, T]$ et absolument continu tel que pour presque tout $t \in [0, T]$:

$$H(X(t), u(t), \lambda(t)) = \min_{v \in \mathbb{U}_m} H(X(t), v, \lambda(t))$$

sachant que :

$$\dot{X}(t) = \frac{\partial H}{\partial \lambda}(X(t), u(t), \lambda(t)) = f(X(t), u(t))$$

$$\dot{\lambda}(t)^T = -\frac{\partial H}{\partial X}(X(t), u(t), \lambda(t))$$

$$H(X(t), u(t), \lambda(t)) = 0 \text{ le long de la trajectoire optimale.}$$

D'après la proposition 4.2.4, nous pouvons construire une famille $(Y_{h,0})_{h>0}$ de points de \mathbb{R}^n tels que pour tout $h > 0$, la trajectoire Y_h solution du système :

$$\dot{Y}_h(t) = f_h(Y_h(t), u(t))$$

relie $Y_{h,0}$ à la cible en temps T i.e. : $Y_h(0) = Y_{h,0}$ et $Y_h(T) = 0$ et qui converge vers X_0 :

$$\lim_{h \rightarrow 0^+} Y_{h,0} = X_0.$$

Il reste maintenant à vérifier que pour un pas $h > 0$ donné, la trajectoire (Y_h, u) vérifie les conditions nécessaires d'optimalité du principe du maximum hybride.

Nous introduisons donc pour cela le paramètre d'état adjoint p_h associé à la trajectoire Y_h et solution du système :

$$\dot{p}_h(t)^T = -\frac{\partial H_h}{\partial X}(Y_h(t), u(t), p_h(t)), \quad t \in [0, T].$$

Par construction, le champ f_h est continue, de classe C^1 par morceaux sur $\mathbb{R}^n \times \mathbb{U}_m$, ce qui implique que l'état adjoint p_h est continu et même absolument continu sur $[0, T]$.

Il reste alors vérifier qu'à tout instant $t \in [0, T]$,

– la famille $(H_h(Y_h(t), u(t), p_h(t)))_{h>0}$ tend vers 0 quand h tend vers 0^+ :

$$\lim_{h \rightarrow 0^+} H_h(Y_h(t), u(t), p_h(t)) = 0.$$

– il existe une famille de réels positifs $(\delta(h))_{h>0}$ qui converge vers 0 quand h tend vers 0 tels que :

$$\min_{v \in \mathbb{U}_m} H_h(Y_h(t), v, p_h(t)) \leq H_h(Y_h(t), u(t), p_h(t)) \leq \min_{v \in \mathbb{U}_m} H_h(Y_h(t), v, p_h(t)) + \delta(h)$$

L'optimalité approchée de l'hamiltonien hybride difficile à démontrer dans ce contexte s'obtient plus simplement dans l'approche par les équations d'Hamilton-Jacobi-Bellman présentée dans la suite.

6.3 Seconde approche : solutions de viscosité des équations d'Hamilton-Jacobi-Bellman

La plupart des problèmes de contrôle optimal en horizon infini peut être ramenée de façon équivalente à la résolution d'une équation d'Hamilton-Jacobi-Bellman (HJB) du premier ordre du type : $H(X, V, DV) = 0$. Cette équation n'admet généralement pas de solution au sens classique, ce qui nous amène à considérer un type plus faible de solution, les solutions de viscosité.

La théorie des solutions de viscosité des équations d'Hamilton-Jacobi-Bellman fait l'objet de nombreuses recherches et des méthodes de résolution numérique ont été proposées : discrétisation en temps et/ou en espace [18], différences finies, méthode des caractéristiques, méthodes Max-Plus entre autres. Dans cette partie, notre but n'est pas de faire une étude complète de ce type de solution mais de montrer que l'approche hybride développée tout au long de ce manuscrit, nous permet de construire ce type de solution.

6.3.1 Contrôle optimal et solutions de viscosité

Dans ce paragraphe, nous proposons un rappel théorique rapide sur les solutions de viscosité et leur rôle en théorie du contrôle optimal. Nous justifions en particulier l'équivalence entre la résolution d'un problème de contrôle optimal et celle d'une certaine équation d'Hamilton-Jacobi-Bellman.

Rappels sur les solutions de viscosité

Soit Ω ouvert non vide de \mathbb{R}^n et $H : \Omega \times \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}$ une application continue. On considère une équation d'Hamilton-Jacobi-Bellman du premier ordre :

$$H(X, V, \nabla V) = 0 \quad (6.6)$$

Sans hypothèse supplémentaire, cette équation n'admet en général pas de solution V au sens classique. On fait alors intervenir le concept plus faible de solution de viscosité introduit pour la première fois en 1981 par M.G. Crandall et P.L. Lions [26, 23, 24, 25] puis largement exploré par la suite ([8], [7, chapitre 2] entre autres) :

Définition 6.3.1 (Solution de viscosité). $V \in C(\Omega)$ est solution de viscosité de (6.6) si et seulement si :

i. $\forall \Phi \in C^1(\Omega)$, si $X_0 \in \Omega$ est un point de maximum local de $(V - \Phi)$, on a :

$$H(X_0, V(X_0), D\Phi(X_0)) \leq 0$$

ii. $\forall \Phi \in C^1(\Omega)$, si $X_0 \in \Omega$ est un point de minimum local de $(V - \Phi)$, on a :

$$H(X_0, V(X_0), D\Phi(X_0)) \geq 0$$

Remarque 6.3.1. Si on a uniquement l'assertion i. (respectivement ii.), on parle de sous-solution (respectivement de sur-solution) de viscosité de l'équation (6.6).

Nous proposons ici une rapide introduction à la théorie des solutions de viscosité continues des équations d'Hamilton-Jacobi-Bellman, ce qui nous permettra ensuite de les introduire dans le contexte des problèmes de contrôle optimal. Commençons ces rappels par quelques propriétés remarquables de ce type de solution :

Proposition 6.3.1.

1. Si V est une solution classique de (6.6) alors V est aussi une solution de viscosité.
2. Si V est une solution de viscosité de (6.6) et si V est différentiable en X_0 alors :

$$H(X_0, V(X_0), DV(X_0)) = 0$$

3. **Changement de variable** : soit $V \in C(\Omega)$ une solution de viscosité de (6.6) et ϕ une application de classe C^1 sur \mathbb{R} telle que $\dot{\phi} > 0$ sur \mathbb{R} ; alors $W = \phi(V)$ est une solution de viscosité de l'équation :

$$K(X, W, DW) = 0 \text{ dans } \Omega$$

en posant : $K(X, Z, p) = H(X, \phi^{-1}(Z), (\phi^{-1})'(Z) p)$

4. $V \in C(\Omega)$ est une sous-solution (resp. sur-solution) de (6.6) si et seulement si $W = -V$ est une sur-solution (resp. sous-solution) de : $-H(X, -W, -DW) = 0$.

En pratique, la définition 6.3.1 est assez peu utilisée pour démontrer qu'une application continue est une solution de viscosité. En fait, une formulation plus commode consiste à introduire la notion de sous- (ou sur-) différentiel d'une fonction continue :

Définition 6.3.2 (Sur- et sous-différentiel d'une fonction continue). Soit $V \in C(\Omega)$.

$$D^+V(X) = \{p \in \mathbb{R}^n / \limsup_{Y \rightarrow X} \frac{V(Y) - V(X) - \langle p, Y - X \rangle}{\|Y - X\|} \leq 0\}.$$

$$D^-V(X) = \{p \in \mathbb{R}^n / \liminf_{Y \rightarrow X} \frac{V(Y) - V(X) - \langle p, Y - X \rangle}{\|Y - X\|} \geq 0\}.$$

Remarquons en particulier que, si V est différentiable en X_0 , on a alors : $D^+V(X_0) = D^-V(X_0) = \{DV(X_0)\}$. Nous pouvons maintenant donner une condition nécessaire et suffisante pour qu'une fonction continue donnée soit une solution de viscosité :

Théorème 6.3.1.

1. $V \in C(\Omega)$ est sous-solution de (6.6) si et seulement si :

$$\forall X \in \Omega, \forall p \in D^+V(X), H(X, V(X), p) \leq 0$$

2. $V \in C(\Omega)$ est sur-solution de (6.6) si et seulement si :

$$\forall X \in \Omega, \forall p \in D^-V(X), H(X, V(X), p) \geq 0$$

Nous terminons ces rappels par un résultat de stabilité des solutions de viscosité dans l'espace $C(\Omega)$ des fonctions continues :

Proposition 6.3.2 ([7, proposition II.2.2]). Soit $V_n \in C(\Omega)$, $n \in \mathbb{N}$, une suite de solutions de viscosité de l'équation :

$$H_n(X, V_n(X), DV_n(X)) = 0 \quad \text{dans } \Omega. \quad (6.7)$$

Supposons que :

$$\begin{aligned} V_n &\xrightarrow{CU} V \text{ uniformément localement sur } \Omega \\ H_n &\xrightarrow{CU} H \text{ uniformément localement sur } \Omega \times \mathbb{R} \times \mathbb{R}^n. \end{aligned}$$

Alors V est une solution de viscosité de l'équation (6.7) dans Ω .

Application au contrôle optimal des systèmes non linéaires

Le problème abordé maintenant est de formuler le problème de contrôle optimal en horizon infini en termes de solution de viscosité d'une équation d'Hamilton-Jacobi-Bellman (HJB). On considère le système de contrôle non linéaire :

$$\begin{cases} \dot{X}(t) &= f(X(t), u(t)) \\ X(0) &= X_0 \end{cases}, \quad X(t_f) = 0, \quad (6.8)$$

sachant que le temps final t_f n'est pas fixé et dépend implicitement du choix du contrôle u . On rappelle que les contrôles admissibles sont des fonctions mesurables, bornées à valeurs dans le polytope \mathbb{U}_m . La fonction coût à minimiser est définie de la façon suivante :

$$J(X_0, u) = \int_0^{t_f} l(X(t), u(t)) dt.$$

On introduit ensuite la fonction valeur V du problème de contrôle optimal \mathcal{P} :

$$V(X_0) = \inf_{u \in L^\infty([0, +\infty[, \mathbb{U}_m)} J(X_0, u) = \inf_{u \in L^\infty([0, +\infty[, \mathbb{U}_m)} \int_0^{t_f} l(X(t), u(t)) dt, \quad (6.9)$$

qui évalue le coût minimum pour transférer le système de la position X_0 à la cible 0. On suppose dans ce paragraphe que pour tout $X \in \mathbb{R}^n$, il existe une trajectoire du système (6.8) qui relie X à 0 en temps fini.

Remarque 6.3.2. *Tous les résultats présentés dans la suite de ce paragraphe s'appliquent tels quels à tout problème de contrôle optimal sans la contrainte sur l'état final. Il n'est alors pas nécessaire de supposer l'existence de trajectoires optimales du système considéré.*

On dispose alors d'un résultat fondamental : le principe de programmation dynamique dû à R. Bellman [9], qui exprime sous forme d'équation l'idée intuitive suivante : pour atteindre le minimum de la fonction valeur, on laisse le système évoluer sur un intervalle de temps $[0, T]$ fixé selon un contrôle arbitrairement fixé. On paie le coût correspondant sur $[0, T]$, à savoir $\int_0^T l(X(t), u(t)) dt$. Il reste alors à choisir les meilleurs contrôles possibles pour minimiser le coût $V(X(T))$ qu'il reste à payer à partir du point $X(T)$. Enfin, on cherche le minimum du coût ainsi calculé sur l'ensemble des contrôles admissibles :

Proposition 6.3.3 (Principe de Programmation Dynamique).

$$\forall X_0 \in \mathbb{R}^n, \forall T > 0, V(X_0) = \inf_{u \in L^\infty([0, +\infty[, \mathbb{U}_m)} \left[\int_0^T l(X(t), u(t)) dt + V(X(T)) \right]$$

Ce résultat est une des bases du contrôle optimal : il permet en particulier de relier la valeur de la fonction V au point X_0 avec ses valeurs en des points voisins, ce qui nous conduit à l'équation satisfaite par V :

Théorème 6.3.2. *La fonction valeur V est une solution de viscosité de l'équation d'Hamilton-Jacobi-Bellman (HJB) :*

$$\sup_{u \in \mathbb{U}_m} \{-l(X, u) - f(X, u)DV(X)\} = 0.$$

On laisse le soin au lecteur de se référer à [7, chapitre 2, propositions 2.5 et 2.8] pour le détail des démonstrations du principe de programmation dynamique et du théorème 6.3.2.

La résolution de l'équation (HJB) comporte trois étapes :

1. Trouver une fonction $S : \mathbb{R}^n \rightarrow \mathbb{U}_m$ tel que : $S(Z) \in \operatorname{argmax}_{u \in \mathbb{U}_m} \{-l(Z, u) - f(Z, u)DV(Z)\}$.
2. Résoudre le système $\begin{cases} \dot{X}(t) = f(X(t), S(X(t))) \\ X(0) = X_0 \end{cases}$ pour $t > 0$.
3. La solution X^* du système précédent engendre un contrôle $u^*(t) = S(X^*(t))$ optimal pour la condition initiale X_0 .

Nous rencontrons ici à l'étape 1, le même problème que celui rencontré lors de la résolution des problèmes de contrôle optimal, à savoir le calcul de la solution directement en fonction de la position du système. Cet obstacle justifie les approches numériques telles que la discrétisation en temps puis en espace de l'équation (HJB), afin de calculer la fonction S en fonction de l'état. Ce qui nous intéresse maintenant est de montrer comment l'approche hybride nous permet d'approcher les solutions de viscosité des équations d'Hamilton-Jacobi-Bellman.

6.3.2 Approche hybride pour la résolution de l'équation (HJB)

D'après l'étude menée dans la partie I, tout système de contrôle non linéaire de la forme : $\dot{X} = f(X, u)$ peut être approché par un système hybride dont la dynamique est affine par morceaux : $\dot{X} = f_h(X, u)$. On définit les hamiltoniens suivants associés respectivement au problème de contrôle optimal non linéaire et hybride :

$$H_0(X, p) = \sup_{u \in \mathbb{U}_m} \{-l(X, u) - p^T f(X, u)\}$$

$$H_h(X, p) = \sup_{u \in \mathbb{U}_m} \{-l(X, u) - p^T f_h(X, u)\}.$$

Les équations (HJB) respectives s'écrivent donc :

$$H_0(X, DV(X)) = 0 \quad H_h(X, DV_h(X)) = 0,$$

où V (respectivement V_h) désigne la fonction valeur associée au problème non linéaire (respectivement hybride).

Le premier résultat que nous obtenons est la convergence uniforme de l'hamiltonien hybride vers l'hamiltonien non linéaire sur tout compact :

Proposition 6.3.4. *On suppose f est de classe C^1 sur $\mathbb{R}^n \times \mathbb{U}_m$. Alors H_h converge uniformément vers H_0 sur tout compact de $\mathbb{R}^n \times \mathbb{R}^n$ quand h tend vers 0^+ :*

$$H_h \xrightarrow{CU} H \text{ sur tout compact de } \mathbb{R}^n \times \mathbb{R}^n.$$

Preuve. Pour $(X, p) \in \mathbb{R}^n \times \mathbb{R}^n$ donnés, d'après la continuité des champs f et f_h , les applications : $\varphi : u \mapsto -l(X, u) - p^T f(X, u)$ et $\varphi_h : u \mapsto -l(X, u) - p^T f_h(X, u)$ sont continues sur \mathbb{U}_m compact. Par conséquent, φ et φ_h sont bornées et atteignent leurs bornes respectives sur \mathbb{U}_m :

$$\begin{aligned} \exists u_0 \in \mathbb{U}_m, \quad H_0(X, p) &= -l(X, u_0) - p^T f(X, u_0) \\ \exists u_h \in \mathbb{U}_m, \quad H_h(X, p) &= -l(X, u_h) - p^T f_h(X, u_h) \end{aligned}$$

On obtient alors :

$$\begin{aligned} H_h(X, p) - H_0(X, p) &= -l(X, u_h) - p^T f_h(X, u_h) - H_0(X, p) \\ &\leq -l(X, u_h) - p^T f_h(X, u_h) + l(X, u_h) + p^T f(X, u_h) \\ &\leq p^T [f(X, u_h) - f_h(X, u_h)] \leq \|p\| \|f(X, u_h) - f_h(X, u_h)\| \end{aligned}$$

Soient Ω_1 et Ω_2 deux compacts de \mathbb{R}^n . Selon le même procédé que pour la démonstration de la proposition 6.2.1, nous utilisons la convergence uniforme de f_h vers f sur tout compact de \mathbb{R}^n , uniformément par rapport à u , ce qui nous donne pour tout $(X, p) \in \Omega_1 \times \Omega_2$:

$$H_h(X, p) - H_0(X, p) \leq M_2 \varepsilon_{\Omega_1}(h) \text{ en posant : } M_2 = \sup_{p \in \Omega_2} \|p\|.$$

Nous rappelons que $\varepsilon_{\Omega_1}(h)$ désigne l'erreur d'interpolation commise sur le compact $\Omega_1 \times \mathbb{U}_m$ et : $\lim_{h \rightarrow 0^+} \varepsilon_{\Omega_1}(h) = 0$.

De même, pour tout $(X, p) \in \Omega_1 \times \Omega_2$, nous avons également :

$$H_0(X, p) - H_h(X, p) \leq \|p\| \|f_h(X, u_0) - f(X, u_0)\| \leq M_2 \varepsilon_{\Omega_1}(h).$$

D'où :

$$\sup_{(X,p) \in \Omega_1 \times \Omega_2} |H_h(X, p) - H_0(X, p)| \leq M_2 \varepsilon_{\Omega_1}(h),$$

ce qui prouve la convergence uniforme de H_h vers H_0 sur tout compact de $\mathbb{R}^n \times \mathbb{R}^n$. □

Considérons maintenant un point X_0 donné de l'espace d'état. Supposons qu'il existe une trajectoire optimale du système non linéaire qui relie X_0 à la cible 0 en temps fini $T > 0$; il existe donc un contrôle admissible $u : [0, T] \rightarrow \mathbb{U}_m$ et une trajectoire X dans \mathbb{R}^n tels que :

$$\begin{cases} \dot{X}(t) &= f(X(t), u(t)) \\ X(0) &= X_0, X(T) = 0 \end{cases} \quad \text{et } V(X_0) = J(X_0, u).$$

D'après la proposition 4.2.4 et selon la même idée que celle utilisée au paragraphe 6.2.3, nous introduisons la famille $(Y_{h,0})_{h>0}$ de points de \mathbb{R}^n tels que pour tout $h > 0$, la trajectoire Y_h solution du système :

$$\dot{Y}_h(t) = f_h(Y_h(t), u(t))$$

relie $Y_{h,0}$ à la cible en temps T i.e. : $Y_h(0) = Y_{h,0}$ et $Y_h(T) = 0$ et vérifiant :

$$\lim_{h \rightarrow 0^+} Y_{h,0} = X_0.$$

De plus le coût pour amener $Y_{h,0}$ à la cible selon le contrôle u est donné par $J_h(Y_{h,0}, u)$ et on a :

$$\begin{aligned} |V(X_0) - J_h(Y_{h,0}, u)| &= |J(X_0, u) - J_h(Y_{h,0}, u)| = \left| \int_0^T [l(X(t), u(t)) - l(Y_h(t), u(t))] dt \right| \\ &\leq \int_0^T |l(X(t), u(t)) - l(Y_h(t), u(t))| dt \\ &\leq L_2 \int_0^T \|X(t) - Y_h(t)\| dt \leq \varepsilon(h) L_2 \frac{e^{L_1 T} - 1 - L_1 T}{L_1^2} \end{aligned}$$

Nous en déduisons ainsi : $\lim_{h \rightarrow 0^+} J_h(Y_{h,0}, u) = V(X_0)$. Nous avons donc montré le résultat suivant :

Proposition 6.3.5. *Soit $X_0 \in \mathbb{R}^n$ un point donné de l'espace d'état. Supposons qu'il existe $T > 0$ et un contrôle admissible $u : [0, T] \rightarrow \mathbb{U}_m$ tels que la trajectoire $X[X_0, u]$ soit une solution optimale du problème (\mathcal{P}_{NL}) .*

Alors il existe une famille $(Y_{h,0})_{h>0}$ de points contrôlables en temps T du système hybride \mathcal{H} qui converge vers X_0 quand h tend vers 0^+ et telle que :

$$\lim_{h \rightarrow 0^+} J_h(Y_{h,0}, u) = V(X_0).$$

Pour améliorer encore ce résultat, il faudrait prouver la convergence uniforme sur tout compact de la fonction valeur hybride V_h vers la fonction valeur V .

6.4 Cas particulier des problèmes de contrôle optimal affines

Nous nous intéressons à la résolution de problèmes de contrôle optimal affines par rapport au contrôle : la dynamique f et la fonction coût l sont supposées affines par rapport à la variable de contrôle u .

Nous proposons dans cette partie une étude détaillée des points de commutation de la trajectoire optimale cherchée, i.e. des points de discontinuité du contrôle associé. Les algorithmes proposés ci-après et présentés dans [35] permettent le calcul des courbes de commutation du problème de contrôle optimal considéré.

On considère un hamiltonien H affinement dépendant du contrôle u , i.e. de la forme :

$$H(X, u, \lambda) = H_0(X, \lambda) + H_1(X, \lambda)u.$$

D'après le théorème 6.2.1 de Pontriaguine, nous cherchons à minimiser H par rapport au contrôle u sous certaines contraintes, sachant que u appartient à un polytope $\mathbb{U}_m = \text{Conv}(s_1, \dots, s_p)$ de \mathbb{R}^m . Le contrôle u est donc soumis à des contraintes constantes affines. Le problème posé par le théorème de Pontriaguine est donc ramené à un problème de programmation linéaire dont la solution est :

$$u^* = s_i \quad \text{si} : \forall j \neq i, H_1(X, \lambda)(s_i - s_j) < 0. \quad (6.10)$$

La relation (6.10) partitionne l'espace en (X, λ) en régions à l'intérieur desquelles le contrôle optimal est constant, égal à un sommet s_i du polytope de contrôle (cf figure 6.1). Ainsi, toute

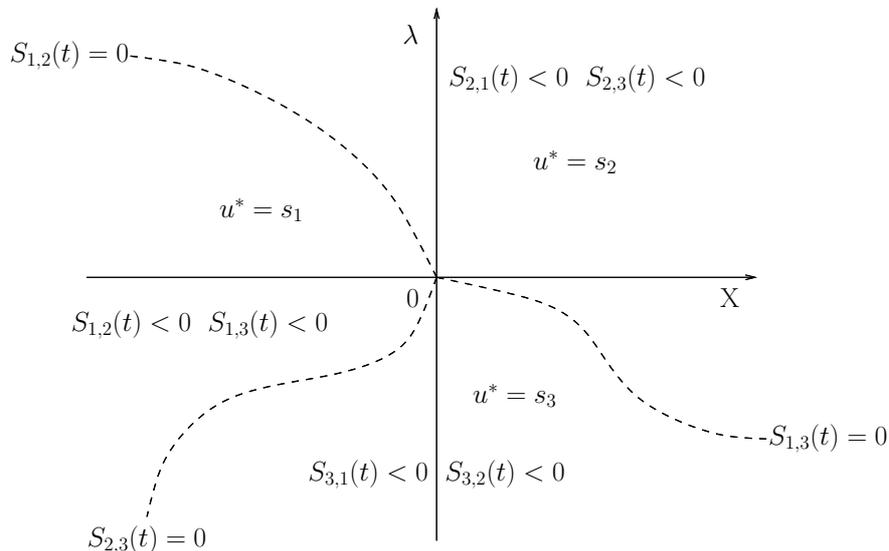


FIG. 6.1 – Partition de l'espace (X, λ) en régions où le contrôle optimal est constant

la difficulté de la résolution d'un problème affine par rapport au contrôle consiste à déterminer les instants et lieux de transition entre deux contrôles s_i et s_j donnés. On introduit pour cela les fonctions de commutation entre les contrôles s_1, \dots, s_p :

Définition 6.4.1 (Fonction de saut). On appelle fonction de saut ou de commutation entre deux contrôles distincts s_i et s_j , l'application $S_{i,j} : (X, \lambda) \mapsto H_1(X, \lambda)(s_i - s_j)$. On a alors la relation : $S_{i,j} \equiv -S_{j,i}$.

La frontière entre les domaines où les contrôles s_i et s_j sont optimaux, si elle existe, a donc pour équation :

$$S_{i,j}(X, \lambda) = H_1(X, \lambda)(s_i - s_j) = 0. \quad (6.11)$$

L'idéal serait, dans un premier temps, d'obtenir les équations de commutation (6.11) uniquement en fonction de la variable d'état X , ce qui nous permettrait de construire un contrôle optimal en boucle de rétroaction (*feedback* en anglais), i.e. dépendant de la position X du système. Malheureusement, il n'est généralement pas possible d'éliminer la variable λ des équations de commutation. Dans un second temps, nous devons déterminer la nature de ces frontières i.e. savoir si la trajectoire optimale traverse la frontière ou si elle "glisse" le long de cette frontière.

En pratique, nous sommes donc ramenés à chercher les points de commutation le long de la trajectoire optimale cherchée, i.e. à étudier les zéros des fonctions de commutation $t \mapsto S_{i,j}(X(t), \lambda(t))$ le long de cette trajectoire.

Considérons deux contrôles s_i et s_j distincts donnés pour lesquels la frontière $S_{i,j}(X, \lambda) = 0$ existe. Soit t_0 un instant de commutation de la trajectoire considérée i.e. vérifiant :

$$S_{i,j}(X(t_0), \lambda(t_0)) = 0.$$

Deux cas se présentent alors :

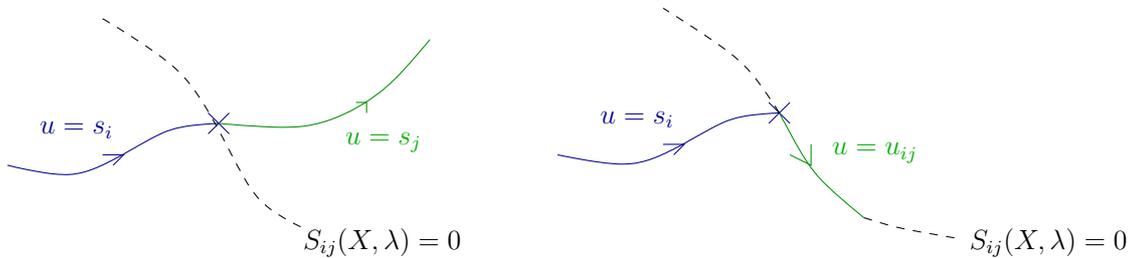


FIG. 6.2 – Arcs régulier (à gauche) et singulier (à droite) avec commutation

- t_0 est un zéro isolé de $t \mapsto S_{i,j}(X(t), \lambda(t))$: dans un voisinage de t_0 , le contrôle optimal est constant par morceaux à valeurs dans $\{s_i, s_j\}$. Les arcs extrémaux sont dits **réguliers** (cf figure 6.2, schéma de gauche).
- Il existe un intervalle de temps $[t_0, t_1]$ non réduit à $\{t_0\}$ tel que la fonction saut est identiquement nulle sur cet intervalle i.e. : $\forall t \in [t_0, t_1], S_{i,j}(X(t), \lambda(t)) = 0$. Les arcs extrémaux sont dits **singuliers** (cf figure 6.2, schéma de droite).

Après avoir rappelé des éléments de théorie des extrémales, nous proposons des algorithmes de calcul des frontières de commutation. L'exemple 6.4.1 présenté ci-après nous permettra d'illustrer les algorithmes proposés tout au long de cette section.

Exemple 6.4.1 (Exemple de Johnson et Gibson [53, exemple 1]). On considère le système de contrôle linéaire :

$$\dot{X}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} X(t) + \begin{bmatrix} 1 \\ -1 \end{bmatrix} u(t), \quad (6.12)$$

où la variable de contrôle est réelle, soumise à la contrainte : $\forall t \geq 0, |u(t)| \leq 1$. On introduit également la fonction coût suivante :

$$J(X_0, u) = \int_0^{+\infty} \frac{1}{2} x_1(t)^2 dt.$$

Le problème de contrôle optimal est alors d'amener le système à la position $X = 0$ en minimisant la fonctionnelle J , sachant que le temps final est libre.

On remarque pour commencer que le système (6.12) est déjà sous forme canonique (cf paragraphe 2.2), puisque la matrice $[B \mid AB]$ est de rang plein (égal à 2).

Appliquons le théorème de Pontriaguine au problème de contrôle considéré. On commence par introduire l'hamiltonien H défini comme suit :

$$H(X, u, \lambda) = \frac{1}{2} x_1^2 + \lambda_1 x_2 + (\lambda_1 - \lambda_2) u \quad (6.13)$$

Le problème de contrôle optimal consiste à minimiser H par rapport à la variable de contrôle u sous les contraintes :

- Système dynamique : $\dot{x}_1 = x_2 + u$ et $\dot{x}_2 = -u$.
- Équations d'Euler-Lagrange : $\dot{\lambda}_1 = -x_1$ et $\dot{\lambda}_2 = -\lambda_1$.
- $\frac{1}{2} x_1^2 + \lambda_1 x_2 + (\lambda_1 - \lambda_2) u \equiv 0$ le long de la trajectoire optimale.

6.4.1 Arcs extrémaux réguliers et règles de commutation

Dans un premier temps, on s'intéresse à la définition et au calcul des arcs extrémaux réguliers. Un arc extrémal régulier est défini de la façon suivante :

Définition 6.4.2 (Arc extrémal régulier). Un arc extrémal $(X(t), u(t), \lambda(t))$ est dit régulier sur un intervalle de temps $[t_0, t_1]$ si et seulement si, pour presque tout $t \in [t_0, t_1]$, il existe un indice $k \in \{1, \dots, p\}$, tel que :

$$H(X(t), \lambda(t))_{s_k} < \min_{i=1 \dots p} H(X(t), \lambda(t))_{s_i}.$$

En utilisant la relation (6.10), on connaît la valeur du contrôle optimal le long d'un arc régulier :

Proposition 6.4.1. Pour tout arc extrémal régulier $(X(t), u(t), \lambda(t))$, le contrôle est donné par la relation :

$$u(t) = s_i \text{ si } : \forall j \neq i, S_{i,j}(X(t), \lambda(t)) < 0.$$

Si, de plus, il existe deux indices $(i, j) \in \{1, \dots, p\}^2$, tels qu'à l'instant t_0 :

$$\forall k \in \{1, \dots, p\} - \{i, j\}, \quad H_1(X(t), \lambda(t))s_i = H_1(X(t), \lambda(t))s_j < H_1(X(t), \lambda(t))s_k,$$

alors t_0 est un instant de commutation du contrôle u de s_i vers s_j ou de s_j vers s_i .

Intéressons nous maintenant aux points de commutation entre deux contrôles distincts s_i et s_j , i.e. aux zéros isolés de la fonction de commutation $t \mapsto S_{i,j}(X(t), \lambda(t))$.

Les points de commutation d'une trajectoire optimale du système entre deux contrôles s_i et s_j , vérifient nécessairement les deux conditions suivantes : ils annulent la fonction de commutation et l'hamiltonien étant nul le long de la trajectoire optimales, il l'est également en ces points, ce qui s'écrit :

$$\begin{cases} S_{i,j}(X(t_0), \lambda(t_0)) = 0 \\ H(X(t_0), u(t_0), \lambda(t_0)) = 0. \end{cases}$$

Or, au point de commutation à l'instant t_0 , le contrôle est de la forme : $u = s_j + (s_i - s_j)v$ avec $v \in \{0, 1\}$. L'hamiltonien H s'écrit sous la forme : $H(X, u, \lambda) = H_0(X, \lambda) + H_1(X, \lambda)s_j + H_1(X, \lambda)(s_i - s_j)v$. Les conditions vérifiées au point de commutation entre les contrôles s_i et s_j s'écrivent donc également :

$$\begin{cases} H_1(X(t_0), \lambda(t_0))(s_i - s_j) = 0 \\ H_0(X(t_0), \lambda(t_0)) + H_1(X(t_0), \lambda(t_0))s_j = 0 \end{cases} \quad (6.14)$$

La question est maintenant de savoir déterminer si la commutation s'effectue de s_i vers s_j ou le contraire. Pour cela, il suffit d'étudier le signe de la dérivée par rapport au temps de la fonction de commutation $t \mapsto S_{i,j}(X(t), \lambda(t))$, i.e. de la quantité :

$$\left. \frac{d}{dt} [S_{i,j}(X(t), \lambda(t))] \right|_{t=t_0}$$

au point de commutation.

- Si $\frac{d}{dt} [S_{i,j}(X(t), \lambda(t))] > 0$ alors le contrôle optimal effectue un saut de s_i vers s_j .
- Si $\frac{d}{dt} [S_{i,j}(X(t), \lambda(t))] < 0$ alors le contrôle optimal effectue un saut de s_j vers s_i .

Il n'est malheureusement pas toujours possible d'exprimer les règles de commutation uniquement dans l'espace d'état. En effet, il aurait pour cela fallu pouvoir éliminer la variable d'état-adjoint λ de l'expression de $\frac{d}{dt} [S_{i,j}(X(t), \lambda(t))]$, ce qui n'est généralement pas faisable puisque (6.14) est un système de 2 équations à n inconnues $\lambda_1, \dots, \lambda_n$.

Il existe par contre bien entendu des exemples où l'on arrive à exprimer le vecteur adjoint λ en fonction de la position X aux points de commutation. C'est le cas de l'exemple 6.4.1 de Johnson et Gibson (cf exemple 6.4.2).

Exemple 6.4.2 (Retour à l'exemple de Johnson et Gibson). On cherche le graphe des commutations autorisées pour le système (6.12). Tout point de commutation $X = (x_1, x_2)$ vérifie les relations (6.14), à savoir ici :

$$\lambda_1 = \lambda_2 \quad \text{et} \quad \frac{1}{2}x_1^2 + \lambda_1 x_2 = 0,$$

soit, encore : $\lambda_1 = \lambda_2 = -\frac{x_1^2}{2x_2}$. Calculons maintenant la dérivée de la fonction de commutation S entre $u = 1$ et $u = -1$, définie comme suit :

$$S := (X, \lambda) \mapsto \lambda_1 - \lambda_2.$$

D'après les équations d'Euler-Lagrange, on sait que : $\dot{\lambda}_1 = -x_1$ et $\dot{\lambda}_2 = -\lambda_1$, d'où :

$$\frac{d}{dt}[\lambda_1 - \lambda_2] = -x_1 + \lambda_1 = -\frac{x_1(x_1 + 2x_2)}{2x_2}.$$

L'étude du signe de $\frac{d}{dt}[\lambda_1 - \lambda_2]$ partitionne ainsi l'espace des phases en régions de commutation autorisées, cf figure 6.4.2.

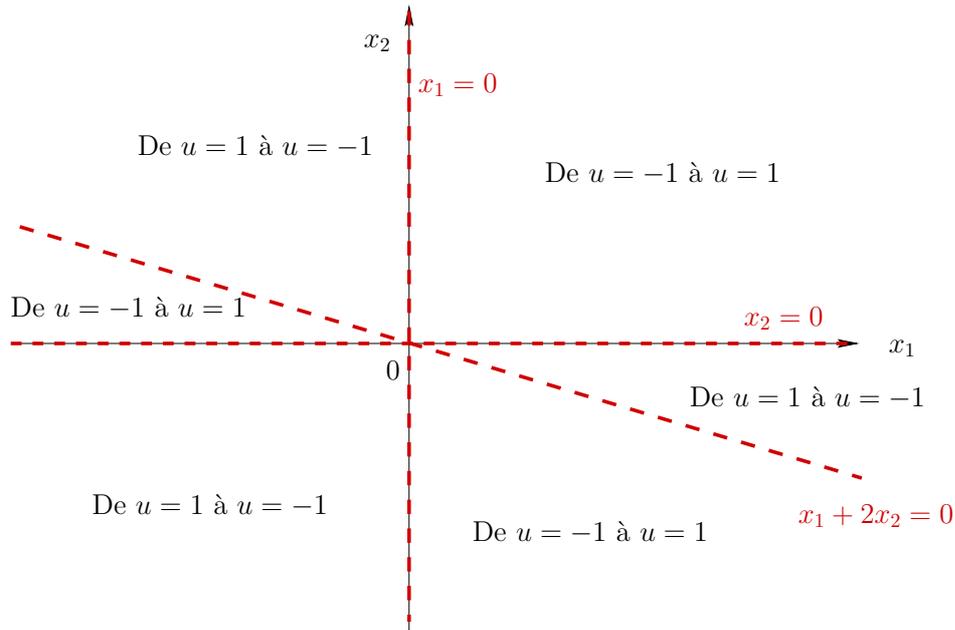


FIG. 6.3 – Graphe des commutations autorisées (exemple de Johnson et Gibson [53])

6.4.2 Arcs extrémaux singuliers

Considérons maintenant le cas où la fonction de commutation s'annule sur un intervalle de temps non trivial. On peut alors formaliser la notion d'arc extrémal singulier de la façon suivante :

Définition 6.4.3 (Arc extrémal singulier [86]). *Un arc extrémal $(X(t), u(t), \lambda(t))$ est dit ij -singulier sur un intervalle de temps $[t_0, t_1]$ (non réduit à un point), si et seulement si pour presque tout $t \in [t_0, t_1]$, on a :*

$$\forall k \in \{1, \dots, p\} - \{i, j\}, \quad H_1(X(t), \lambda(t))s_i = H_1(X(t), \lambda(t))s_j < H_1(X(t), \lambda(t))s_k.$$

Plus généralement, soit $I \subset \{1, \dots, p\}$; un arc $(X(t), u(t), \lambda(t))$ est dit I -singulier sur $[t_0, t_1]$ si et seulement si pour presque tout $t \in [t_0, t_1]$, on a :

$$\forall (i, j) \in I^2, \quad \forall k \in \{1, \dots, p\} - I, \quad H_1(X(t), \lambda(t))s_i = H_1(X(t), \lambda(t))s_j < H_1(X(t), \lambda(t))s_k.$$

Ainsi, géométriquement, sur l'intervalle de temps $[t_0, t_1]$, la trajectoire optimale "suit" la frontière entre les domaines où s_i est optimal, $i = 1, \dots, p$. Le contrôle dit singulier associé à cette trajectoire est à déterminer. On a le résultat suivant :

Proposition 6.4.2. *Soit $I \subset \{1, \dots, p\}$; pour tout arc extrémal $(X(t), u(t), \lambda(t))$ I -singulier sur un intervalle de temps $[t_0, t_1]$, le contrôle vérifie :*

$$\forall t \in [t_0, t_1], \quad u(t) \in \text{Conv}(s_i, i \in I).$$

Démonstration. Par définition, le domaine de contrôle est un polytope de \mathbb{R}^m de sommets s_1, \dots, s_p . Donc, quel que soit $t \in [t_0, t_1]$, on a la relation de convexité suivante :

$$\exists (\alpha_k(t))_{k=1\dots p} \in [0, 1]^p, \quad \sum_{k=1}^p \alpha_k(t) = 1 \quad \text{et} \quad u(t) = \sum_{k=1}^p \alpha_k(t)s_k.$$

En réinjectant maintenant le contrôle dans l'hamiltonien H , nous sommes donc ramenés de façon équivalente à minimiser H par rapport aux variables α_k , $k = 1, \dots, p$; redéfinissons donc H le long de l'arc singulier, en omettant d'écrire la dépendance en t de X , α et λ afin pour alléger les notations :

$$H(X, \alpha, \lambda) = H_0(X, \lambda) + H_1(X, \lambda) \sum_{k=1}^p \alpha_k s_k.$$

Soit $i \in I$; d'après la relation de convexité, on sait que : $\alpha_i = 1 - \sum_{k=1, k \neq i}^p \alpha_k$, d'où :

$$\begin{aligned} H(X, \alpha, \lambda) &= H_0(X, \lambda) + \alpha_i H_1(X, \lambda)s_i + \sum_{k=1, k \neq i}^p \alpha_k H_1(X, \lambda)s_k \\ &= H_0(X, \lambda) + H_1(X, \lambda)s_i + \sum_{k=1, k \neq i}^p \alpha_k H_1(X, \lambda)(s_k - s_i) \end{aligned}$$

Par ailleurs, d'après la définition 6.4.3 d'un arc I -singulier, on sait que :

$$\forall j \in I, \quad \forall k \in \{1, \dots, p\} - I, \quad H_1(X, \lambda)(s_k - s_j) > 0.$$

On obtient alors nécessairement : $\forall k \in \{1, \dots, p\} - I, \quad \alpha_k \equiv 0$, ce qui signifie donc que : $\forall t \in [t_0, t_1], \quad u(t) \in \text{Conv}(s_i; i \in I)$. \square

Il reste donc maintenant à calculer ces arcs singuliers et à déterminer s'ils sont optimaux ou non.

Remarque 6.4.1 (Contrôle optimal en temps minimum). *En appliquant les résultats précédents, nous pouvons vérifier que les problèmes de contrôle optimal linéaire en temps minimum ne comportent jamais d'arcs singuliers dans l'expression de leurs extrémales [16, 68].*

Calcul des contrôles singuliers

Le calcul du contrôle le long d'un arc singulier repose sur le fait que le long d'une extrémale singulière, les dérivées successives de la fonction $t \mapsto H_u(X(t), u(t), \lambda(t))$ sont nécessairement nulles, soit :

$$\forall k \in \mathbb{N}^*, \quad \frac{d^k}{dt^k} H_u(X(t), u, \lambda(t)) = 0.$$

De plus, dans [60], les auteurs démontrent que la variable de contrôle u n'apparaît que dans les dérivées d'ordre pair. On dérive donc $t \mapsto H_u(X(t), u(t), \lambda(t))$ jusqu'à la première dérivée d'ordre $2K$ faisant intervenir la variable u .

Par ailleurs, [60, 72] nous donne alors une condition nécessaire d'optimalité des arcs singuliers :

Proposition 6.4.3 (Condition de Legendre-Clebsch). *Une condition nécessaire d'optimalité d'un arc extrémal singulier est que :*

$$(-1)^k \frac{\partial}{\partial u} \left(\frac{d^{2k}}{dt^{2k}} \frac{\partial H}{\partial u} \right) \geq 0$$

le long de l'arc singulier.

Sous réserve que la condition de Legendre-Clebsch soit bien vérifiée, on résout donc le système :

$$\begin{cases} H(X, u, \lambda) = 0 \\ \frac{d^k}{dt^k} \frac{\partial H}{\partial u}(X, u, \lambda) = 0, \quad k = 0, 1, \dots, 2K, \end{cases}$$

où K est le premier indice tel que l'on ait : $\frac{\partial}{\partial u} \left(\frac{d^{2K}}{dt^{2K}} \frac{\partial H}{\partial u} \right) \neq 0$. La singularité, si elle existe, est dite d'ordre K .

Algorithme de calcul des frontières singulières

On se donne s_i et s_j deux sommets distincts du polytope de contrôle \mathbb{U}_m . L'algorithme 10 calcule symboliquement, si elle existe, l'équation de la frontière singulière entre les domaines où les contrôles s_i et s_j sont optimaux.

Exemple 6.4.3 (Cas des systèmes linéaires avec un coût quadratique). *On considère un système de contrôle de la forme :*

$$\dot{X}(t) = AX(t) + Bu(t)$$

Algorithme 10 Frontière ij -singulière

Données : s_i et s_j les contrôles sommets du polytope de contrôle \mathbb{U}_m considérés.

Données : $(X, u, \lambda) \mapsto H(X, u, \lambda)$ l'hamiltonien du système.

Sortie : φ , où $\varphi(X) = 0$ est l'équation de la frontière ij -singulière

Sortie : u^* contrôle ij -singulier, λ^* vecteur état adjoint le long de la frontière ij -singulière.

1: {D'après la proposition 6.4.2, le contrôle u^* cherché est de la forme : $u^* = s_i + (s_j - s_i)v$ avec $v \in [0, 1]$. On définit l'hamiltonien H en fonction de v :}

$$\tilde{H} : (X, v, \lambda) \mapsto H(X, s_i + (s_j - s_i)v, \lambda);$$

2: Calcul du plus petit entier $K \in \mathbb{N}$ vérifiant : $\frac{\partial}{\partial v} \left(\frac{d^{2K}}{dt^{2K}} \tilde{H}_v \right) \neq 0$.

3: **si** $(-1)^K \frac{\partial}{\partial v} \left(\frac{d^{2K}}{dt^{2K}} \tilde{H}_v \right) \geq 0$ (condition de Legendre-Clebsch) **alors**

4: "Pas de solution singulière".

5: **sinon**

6: Résoudre le système : $(\mathcal{S}) \left\{ \tilde{H} = 0, \tilde{H}_v = 0, \left(\frac{d^i}{dt^i} \tilde{H}_v = 0 \right)_{i=1..2K} \right\}$.

{ (\mathcal{S}) est linéaire par rapport à v et λ , ce qui nous permet de le résoudre de façon exacte.}

7: Retourner $(\varphi, s_j + (s_i - s_j)v, \lambda)$.

8: **fin si**

soumis à des contraintes affines constantes sur le contrôle : $u \in \mathbb{U}_m = \text{Conv}(s_1, \dots, s_p)$ et une fonction coût quadratique indépendante du contrôle u : $l : (X, u) \mapsto \frac{1}{2} X^T X$.

On introduit maintenant l'hamiltonien associé :

$$H(X, u, \lambda) = \frac{1}{2} X^T X + \lambda^T A X + \lambda^T B u.$$

On se donne deux contrôles s_i et s_j distincts. On veut calculer la frontière ij -singulière. D'après la proposition 6.4.2, on sait que le contrôle u cherché est de la forme :

$$u = s_i + (s_j - s_i)v \quad \text{où } 0 \leq v \leq 1.$$

On note $H(X, v, \lambda)$ l'hamiltonien $H(X, s_i + (s_j - s_i)v, \lambda)$ et on calcule :

$$\frac{\partial H}{\partial v}(X, v, \lambda) = \lambda^T B(s_j - s_i).$$

Calculons les dérivées par rapport au temps de H_v :

$$\begin{aligned} \frac{d}{dt} H_v &= \dot{\lambda}^T B(s_j - s_i) = -\frac{\partial H}{\partial X} B(s_j - s_i) \quad (\text{équations d'Euler-Lagrange}) \\ &= -X^T B(s_j - s_i) - \lambda^T A B(s_j - s_i) \end{aligned}$$

$$\begin{aligned} \frac{d^2}{dt^2} H_v &= -\dot{X}^T B(s_j - s_i) - \dot{\lambda}^T A B(s_j - s_i) \\ &= -(AX + B s_i + B(s_j - s_i)v)^T B(s_j - s_i) + X^T A B(s_j - s_i) + \lambda^T A^2 B(s_j - s_i) \\ &= -X^T A^T B(s_j - s_i) + X^T A B(s_j - s_i) - s_i^T B^T B(s_j - s_i) \end{aligned}$$

$$-(s_j - s_i)^T B^T B(s_j - s_i)v + \lambda^T A^2 B(s_j - s_i)$$

La condition de Legendre-Clebsch donnée par la proposition 6.4.3 est bien vérifiée ; en effet,

$$-\frac{\partial}{\partial v} \frac{d^2}{dt^2} H_v = (s_j - s_i)^T B^T B (s_j - s_i) = \|B(s_j - s_i)\|^2 \geq 0,$$

et le contrôle optimal v s'exprime en fonction de X et λ :

$$v = \frac{-X^T A^T B(s_j - s_i) + X^T A B(s_j - s_i) - s_i^T B^T B(s_j - s_i) + \lambda^T A^2 B(s_j - s_i)}{\|B(s_j - s_i)\|^2}$$

Pour calculer le paramètre λ en fonction de la position X du système, il nous reste donc à résoudre :

$$\begin{cases} \frac{1}{2} X^T X + \lambda^T A X + \lambda^T B s_i = 0 \\ \lambda^T B(s_j - s_i) = 0 \\ -X^T B(s_j - s_i) - \lambda^T A B(s_j - s_i) = 0 \end{cases}$$

C'est un système linéaire par rapport à la variable λ , de 3 équations à n inconnues. En conséquence, l'ensemble des solutions est un sous-espace vectoriel de dimension $n-3$ si $n > 3$. Si $n = 2$ ou $n = 3$, alors on peut calculer le paramètre de Pontriaguine λ sous forme de boucle de rétroaction (i.e. directement en fonction de la position X du système linéaire initial). Si $n > 3$, le contrôle singulier dépend du paramètre λ , ce qui nous incite à recourir à une méthode de tir (cf chapitre 1).

Exemple 6.4.4 (Retour à l'exemple de Johnson et Gibson). On rappelle que l'hamiltonien du système (6.12) est donné par l'expression (6.13) :

$$H(X, u, \lambda) = \frac{1}{2} x_1^2 + \lambda_1 x_2 + (\lambda_1 - \lambda_2) u.$$

Comme prévu, la dérivée de H par rapport à u ne dépend plus de u : $H_u(X, u, \lambda) = \lambda_1 - \lambda_2$. Calculons donc maintenant les dérivées successives de l'application : $t \mapsto H_u(X(t), u, \lambda(t))$ jusqu'à obtenir une dérivée qui dépende du contrôle u :

$$\begin{aligned} \frac{d}{dt} H_u &= \dot{\lambda}_1 - \dot{\lambda}_2 = -x_1 + \lambda_1. \\ \frac{d^2}{dt^2} H_u &= -\dot{x}_1 + \dot{\lambda}_1 = -x_2 - u - x_1. \end{aligned}$$

On en déduit alors la valeur du contrôle et du vecteur d'état-adjoint singuliers en fonction de la position :

$$u = -(x_1 + x_2) \quad \text{et} \quad \lambda_1 = \lambda_2 = x_1,$$

sachant que le contrôle est soumis à la contrainte : $|u| \leq 1$. En réinjectant ces résultats dans l'équation $H \equiv 0$ le long de la trajectoire optimale, on obtient alors :

$$\frac{1}{2} x_1^2 + x_1 x_2 = 0.$$

On obtient ainsi deux équations de trajectoires singulières possibles :

$$x_1 = 0 \quad \text{ou} \quad x_1 + 2x_2 = 0 \quad \text{sous la condition} \quad |x_1 + x_2| \leq 1,$$

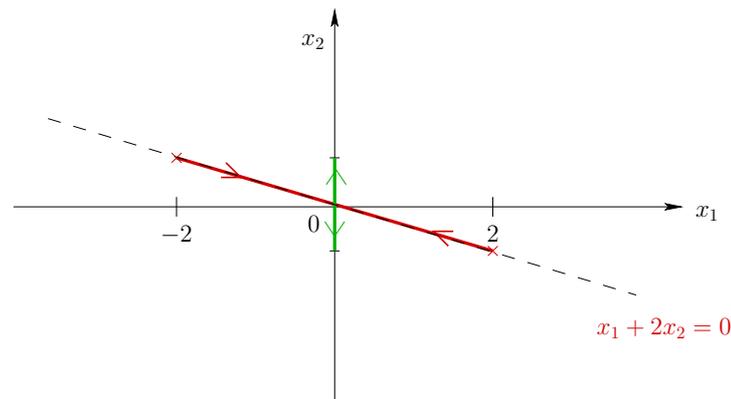


FIG. 6.4 – Frontières singulières potentielles du problème de Johnson et Gibson

représentées sur la figure 6.4. On rappelle que la cible donnée initialement est le point 0 de l'espace d'état. L'étude de la direction du champ de vecteurs le long de la trajectoire optimale (donnée sur la figure 6.4), montre que la trajectoire singulière selon $x_1 = 0$ s'éloigne de la cible 0 et ne peut donc pas être optimale. On en déduit alors que la seule frontière singulière admissible du problème de contrôle considéré a pour équation :

$$x_1 + 2x_2 = 0 \quad \text{avec} \quad |x_1| \leq 2,$$

sachant que le contrôle singulier associé est égal à : $u = x_2$.

Dans l'exemple 6.4.4, nous avons trouvé une trajectoire singulière admissible du problème de contrôle considéré. Toutefois, cette frontière singulière ne nous permet de construire une partition de l'espace d'état en sous-domaines où le contrôle optimal est constant (égal à l'un des sommets de contrôle s_i , $i = 1, \dots, p$).

Dans le prochain paragraphe, nous allons voir qu'il existe de nombreux cas où les trajectoires optimales sont constituées à la fois d'arcs extrémaux réguliers et d'arcs singuliers. Nous proposons en particulier un algorithme de calcul des frontières sous-jacentes (dites mixtes).

6.4.3 Arcs extrémaux mixtes et conditions de jonction

Un arc extrémal mixte correspond à la jonction d'un arc régulier avec un arc singulier. Considérons deux contrôles s_i et s_j distincts ; par extension, la frontière entre les domaines où les contrôles s_i et s_j sont respectivement optimaux, est susceptible de comporter des portions de natures différentes.

Dans ce paragraphe, après avoir rappelé très succinctement les conditions de jonction d'un arc régulier avec un arc singulier, nous proposons un algorithme permettant le calcul de la frontière mixte sous-jacente.

Conditions de jonction d'arcs régulier et singulier [60, §3.17]

Les auteurs de [60] résument les conditions de jonction d'un arc régulier avec un arc singulier de la façon suivante : soit K l'ordre de la singularité considérée,

- si K est impair, un saut (non continu) du contrôle à partir d'un arc singulier est possible sous réserve que la condition :

$$H_u \Delta u \leq 0.$$

Dans ce cas, si le contrôle singulier est maintenu jusqu'à saturation ($u = s_i$ par exemple), alors le contrôle doit effectuer un saut de s_i vers s_j .

- si K est pair, tout saut du contrôle à partir d'un arc singulier pour lequel on a la condition forte de Legendre-Clebsch, à savoir :

$$(-1)^K \frac{\partial}{\partial u} \left[\frac{d^{2K}}{dt^{2K}} H_u \right] > 0,$$

est exclu.

Dans le cas d'une jonction entre un arc régulier et un arc ij -singulier, nous obtenons le résultat suivant qui nous donne la valeur du contrôle régulier avant une commutation vers un arc singulier :

Proposition 6.4.4. *Soit $X(\cdot)$ une trajectoire optimale issue d'un point X_0 et u le contrôle optimal associé. Supposons qu'il existe un temps $T > 0$ et $\epsilon > 0$ tel que la trajectoire X soit régulière sur $[T - \epsilon, T]$ et singulière sur $[T, T + \epsilon[$.*

Alors :

$$\forall t \in [T - \epsilon, T], u(t) \in \{s_i, s_j\}.$$

Algorithme de calcul des arcs mixtes

On considère deux contrôles s_i et s_j distincts, sommets du polytope de contrôle \mathbb{U}_m . On suppose alors qu'il existe une frontière ij -singulière bornée qui sépare (partiellement) les domaines où les contrôles s_i et s_j sont respectivement optimaux. L'équation de cette frontière est supposée de la forme :

$$\varphi(X) = 0 \quad \text{sous la contrainte} \quad 0 \leq v_{i,j}(X) \leq 1,$$

où $u_{i,j} = s_j + (s_i - s_j)v_{i,j}$ et $\lambda_{i,j}$ désignent respectivement le contrôle et le vecteur d'état-adjoint le long de la trajectoire singulière.

On s'intéresse à la fonction de commutation entre les contrôles s_i et s_j :

$$S_{i,j}(X, \lambda) = H_1(X, \lambda)(s_i - s_j).$$

La frontière régulière que l'on cherche est, par définition, le lieu des points (X, λ) de $\mathbb{R}^n \times \mathbb{R}^n$ qui annulent la fonction $S_{i,j}$.

L'idée pour calculer la partie régulière de la frontière entre les domaines où s_i et s_j sont optimaux, est la suivante : on se donne un point X_0 de la frontière singulière et on calcule la trajectoire selon le contrôle $u = s_i$ (respectivement $u = s_j$) qui arrive en X_0 . On cherche alors le dernier point de commutation de la trajectoire.

Plus précisément, on procède en pratique de la façon suivante : étant donné un point X_0 de la frontière singulière,

1. on calcule la trajectoire $X[X_0, u]$ issue de X_0 selon le contrôle s_i (resp. s_j),
2. ainsi que le vecteur d'état-adjoint associé, noté $\lambda[X_0, u]$, en résolvant les équations d'Euler-Lagrange :

$$\begin{cases} \dot{\lambda}(t)^T &= -\frac{\partial H}{\partial \lambda}(X(t), s_i, \lambda(t)) \\ \lambda(0) &= \lambda_{i,j}(X_0) \end{cases}$$

3. On définit alors la fonction de commutation le long de la trajectoire $X[X_0, u]$ comme suit :

$$\tilde{S}_{i,j} := t \mapsto S_{i,j}(X[X_0, u](t), \lambda[X_0, u](t)).$$

4. On cherche le premier temps $t(X_0) < 0$, auquel la fonction de commutation $\tilde{S}_{i,j}$ s'annule. Par cet algorithme, on trouve ainsi un point $X[X_0, u](t(X_0))$, qui est un point de commutation régulier du contrôle s_j vers le contrôle s_i . En faisant varier X_0 le long de la trajectoire singulière, on reconstitue donc la partie régulière de la frontière cherchée (cf exemple 6.4.5).

Algorithme 11 FrontièreMixte

Données : s_i et s_j les contrôles considérés, sommets du polytope de contrôle \mathbb{U}_m , $H = H_0 + H_1 u$ l'hamiltonien du système.

Données : φ équation de la frontière ij -singulière.

Données : λ^* vecteur d'état-adjoint le long de la frontière ij -singulière.

Sortie : une paramétrisation de la courbe de commutation régulière entre s_i et s_j .

- 1: Calcul d'une paramétrisation de la frontière singulière (via le théorème des fonctions implicites) : $\psi(\xi)$ (i.e. telle que $\varphi(\psi(\xi)) = 0$).
 - 2: **pour** $s \in \{s_i, s_j\}$ **faire**
 - 3: Calcul de la trajectoire $X[\psi(\xi), s]$ issue de $\psi(\xi)$ selon le contrôle $u = s$.
 - 4: Résolution des équations d'Euler-Lagrange avec la condition initiale $\lambda[\psi(\xi), s](0) = \lambda^*(\psi(\xi))$.
 - 5: {Fonction de commutation :} $S_{i,j} := t \mapsto H_1(X[\psi(\xi), s](t), \lambda[\psi(\xi), s](t))(s_i - s_j)$.
 - 6: Calcul du premier temps $t(\xi) < 0$ auquel la fonction de commutation entre s_i et s_j s'annule i.e. : $S_{i,j}(t(\xi)) = 0$.
 - 7: **fin pour**
 - 8: Retourner la courbe de commutation, si elle existe, sous forme paramétrée : $X[\psi(\xi), s](t(\xi)) = 0$, $s \in \{s_i, s_j\}$
-

Exemple 6.4.5 (Retour à l'exemple de Johnson et Gibson). Dans l'exemple 6.4.4, nous avons trouvé une trajectoire singulière admissible du système (6.12), d'équation :

$$x_1 + 2x_2 = 0 \quad \text{avec} \quad |x_1| \leq 2.$$

Le long de cette trajectoire, on connaît les valeurs du contrôle optimal, ainsi que du vecteur d'état-adjoint :

$$u = x_2 \quad \text{et} \quad \lambda_1 = \lambda_2 = x_1 = -2x_2.$$

Malheureusement, cette trajectoire ou frontière singulière ne suffit pas à identifier les domaines où le contrôle optimal est respectivement égal à 1 et -1 . Cependant, l'algorithme 11 va nous

permettre de compléter l'étude menée dans [53], en calculant la partie régulière de la frontière de commutation entre $u = 1$ et $u = -1$. On définit la fonction de commutation entre les contrôles 1 et -1 :

$$S := t \mapsto \lambda_1(t) - \lambda_2(t).$$

L'algorithme 11 procède de la façon suivante :

1. On calcule une paramétrisation de la frontière singulière ; ici : $\psi(\xi) = (-2\xi, \xi)$.
2. Pour $u = 1$, on calcule la trajectoire issue de $\psi(\xi)$ selon le contrôle u :

$$X[\psi(\xi), u](t) = \left[-\frac{1}{2}t^2 + (\xi + 1)t - 2\xi, -t + \xi \right].$$

3. On résout les équations d'Euler-Lagrange, sachant qu'à $t = 0$, le vecteur d'état-adjoint λ appartient à la frontière singulière, i.e. : $\lambda_1(0) = \lambda_2(0) = -2\xi$:

$$\begin{cases} \dot{\lambda}_1(t) = \frac{1}{2}t^2 - (\xi + 1)t + 2\xi \\ \dot{\lambda}_2(t) = -\lambda_1(t) \end{cases} \Rightarrow \begin{cases} \lambda_1(t) = \frac{1}{6}t^3 - \frac{\xi + 1}{2}t^2 + 2\xi t - 2\xi \\ \lambda_2(t) = -\frac{1}{24}t^4 + \frac{\xi + 1}{6}t^3 - \xi t^2 + 2\xi t - 2\xi \end{cases}$$

afin d'en déduire la valeur de la fonction de commutation S le long de la trajectoire $X[\psi(\xi), u]$:

$$S(t) = \frac{1}{24}t^4 - \frac{1}{6}t^3 + \frac{\xi - 1}{2}t^2.$$

4. On calcule le premier temps $t(\xi) < 0$ tel que $S(t(\xi)) = 0$. On trouve : $t(\xi) = 2\xi - 2\sqrt{\xi^2 - 3\xi + 3}$.
5. On répète les étapes 2, 3 et 4 pour le contrôle $u = -1$.
6. On obtient une paramétrisation de la courbe de commutation en fonction de $\xi \in [-1, 1]$:

$$\begin{cases} (2\xi\sqrt{\xi^2 - 3\xi + 3} - 2\xi^2 + 6\xi - 6 - 2\sqrt{\xi^2 - 3\xi + 3}, -\xi + 2\sqrt{\xi^2 - 3\xi + 3}) \\ (2\xi\sqrt{\xi^2 + 3\xi + 3} + 2\xi^2 + 6\xi + 6 + 2\sqrt{\xi^2 + 3\xi + 3}, -\xi - 2\sqrt{\xi^2 + 3\xi + 3}) \end{cases}$$

La frontière entre les domaines où les contrôles $u = 1$ et $u = -1$ sont optimaux, est donc composée de trois portions de natures différentes.

On se donne la condition initiale $X_0 = (-7, 0)$. La trajectoire optimale du système (6.12) issue de ce point est représentée sur la figure 6.5 ; elle comporte deux points de commutation A et B de nature différentes :

- le point A appartient à la portion régulière de la frontière, le contrôle optimal effectuée en ce point une transition instantanée du contrôle $u = -1$ au contrôle $u = 1$.
- Le point B appartient à la portion singulière de la courbe. Le trajet optimal pour rejoindre la cible 0 consiste alors à glisser le long de la frontière jusqu'à atteindre la cible.

Pour terminer, remarquons que le long de la trajectoire singulière ($x_1 + 2x_2 = 0$, $|x_1| \leq 2$), le système (6.12) s'écrit :

$$\begin{cases} \dot{x}_1 = 2x_2 \\ \dot{x}_2 = -x_2 \end{cases} \quad \text{ce qui nous donne : } \begin{cases} x_1(t) = -2Ce^{-t} \\ x_2(t) = Ce^{-t} \end{cases}$$

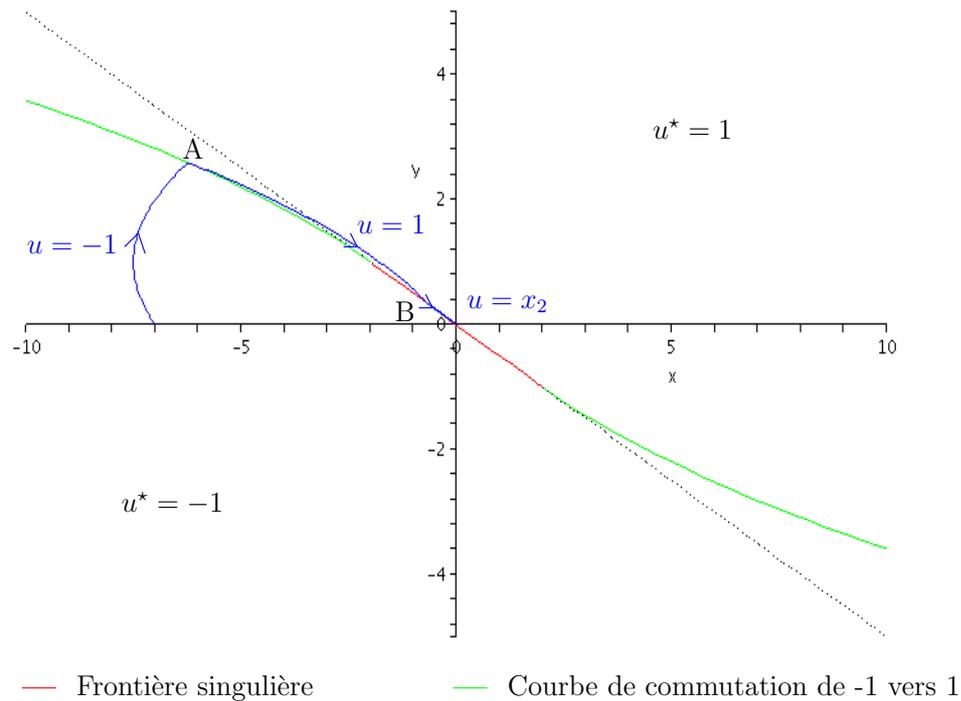


FIG. 6.5 – Frontière mixte dans le problème de Johnson et Gibson et trajectoire optimale du système à partir du point initial $(-7, 0)$

où $C \in \mathbb{R}$ est une constante dépendant de la condition initiale. Par conséquent, la trajectoire atteint la cible 0 en temps infini, le coût minimal $V(-7, 0)$ pour aller de $(-7, 0)$ à 0 étant fini (égal à environ 80.3).

En pratique, nous ne sommes pas toujours en mesure de résoudre de façon exacte l'équation $\tilde{S}_{i,j}(t) = 0$ de l'étape 6. et donc d'obtenir le temps $t(\xi)$ nécessaire pour reconstruire la courbe de commutation cherchée. Pour pallier à cette difficulté, la solution que nous proposons consiste à discrétiser la frontière singulière, puis à appliquer l'algorithme de reconstruction de la partie régulière de la frontière à partir de ces points. Nous obtenons ainsi une discrétisation de la partie régulière de la frontière que nous pouvons alors reconstituer par interpolation (par exemple).

Application au calcul des frontières régulières

Nous nous intéressons maintenant au calcul de la courbe de commutation entre deux contrôles s_i et s_j donnés, supposée totalement non singulière. Nous supposons également qu'il existe un sommet s_k du domaine de contrôle \mathbb{U}_m , autre que s_i ou s_j , tel qu'il existe une frontière de commutation singulière entre s_j et s_k .

L'idée est alors d'appliquer la même technique que celle utilisée pour l'algorithme de calcul des frontières mixtes. De plus, d'après la proposition 6.4.4, toute trajectoire optimale atteint

la frontière jk -singulière avec un contrôle optimal constant égal à s_j ou s_k . L'idée est donc la suivante : par renversement du temps selon le contrôle s_j on cherche le premier point de commutation avec le contrôle s_i . L'algorithme est donc le suivant :

1. On calcule une paramétrisation $\psi(\xi)$ de la frontière jk -singulière.
2. On introduit la trajectoire $X[\psi(\xi), s_j]$ issue du point $\psi(\xi)$ selon le contrôle s_j .
3. On calcule le premier instant de commutation t_i de la trajectoire $X[\psi(\xi), s]$, i.e. :

$$t_i = \sup\{t < 0; S_{i,j}(t) = 0\}.$$

4. On en déduit une paramétrisation de la frontière de commutation régulière :

$$X[\psi(\xi), s](t(\xi)) = 0, s \in \{s_j, s_k\}.$$

Supposons maintenant que toutes les frontières de commutation sont régulières. Dans ce cas, le contrôle optimal est dit "bang-bang" i.e. constant par morceaux à valeurs dans l'ensemble $\{s_1, \dots, s_p\}$ des sommets du domaine de contrôle \mathbb{U}_m . L'idée est alors de calculer toutes les fonctions de commutation $S_{i,j}$ et d'étudier leurs zéros et leurs changements de signes pour en déduire la trajectoire optimale. Par renversement du temps et en partant de la cible, on reconstruit ainsi les frontières de commutation régulières. Cette méthode est décrite dans [68].

En conclusion, nous avons développé dans le paragraphe 6.4 des algorithmes permettant de calculer les courbes de commutation - régulières, singulières et mixtes - pour des problèmes de contrôle optimal linéaires dans le cas particulier où l'on arrive à exprimer l'état adjoint λ en fonction de la position X . Ces algorithmes devraient constituer une première piste vers un algorithme de résolution plus global.

Pour terminer, citons le phénomène de Fuller [41], exemple caractéristique dans l'étude des contrôles singuliers non pris en compte dans l'étude précédente. Ce phénomène correspond en pratique à l'existence d'un nombre infini de commutations sur un intervalle de temps fini et constitue une branche à part en théorie du contrôle géométrique. Le lecteur pourra se référer à [86] pour une étude approfondie de ce type de phénomène.

Dans ce chapitre, nous avons montré que quelle que soit l'approche considérée, la résolution du problème de contrôle hybride permet une résolution approchée du problème non linéaire qu'il modélise. Dans la dernière partie, nous avons proposé des algorithmes de calcul des frontières de commutation dans le cas où celles-ci peuvent être exprimées directement en fonction de la position X dans l'espace d'état. Nous avons insisté sur la difficulté d'obtenir des algorithmes génériques de résolution de ce type de problème.

Le chapitre suivant est consacré à la recherche de trajectoires optimales des problèmes de contrôle associés aux systèmes hybrides affines par morceaux définis au chapitre 3 de ce manuscrit.

Chapitre 7

Stratégie de résolution du problème de contrôle optimal hybride

Ce chapitre est consacré à la résolution algorithmique des problèmes de contrôle optimal associés à la classe des systèmes hybrides affines par morceaux introduits au chapitre 3. L'objectif est de mettre au point des algorithmes efficaces de résolution permettant ainsi d'approcher les solutions optimales des problèmes de contrôle non linéaires que l'on modélise.

Le contrôle optimal des systèmes hybrides est un domaine de recherche relativement récent et en plein essor. Cependant, contrairement au contrôle des systèmes continus pour lesquels les problèmes sont bien identifiés, celui des systèmes hybrides manque à l'heure actuelle d'un cadre théorique unifié. Outre le manque de régularité des problèmes de contrôle hybrides en général, une difficulté majeure est d'arriver à concilier le caractère à la fois continu et discret de ces systèmes : en effet, même si la dynamique continue dans un mode donné de l'automate hybride est bien connue, les transitions discrètes entre les modes peuvent engendrer des phénomènes inattendus tels que les phénomènes de *murs noirs* par exemple, sur lesquels les solutions au sens classique ne sont pas définies [69, 39].

Face à ces difficultés, une approche possible est la discrétisation du problème de contrôle optimal hybride. Par exemple, S. Hedlund et A. Rantzer proposent une technique de résolution basée sur une discrétisation de l'équation de Bellman [9] se ramenant ainsi à un problème d'optimisation convexe avec des arguments de programmation linéaire [49].

D'autres approches tendent à respecter au mieux la nature même des systèmes hybrides en tenant compte des deux composantes - continue et discrète - de ces systèmes. Une première approche est la synthèse d'un contrôle discret séquentiel privilégiant ainsi la composante discrète du système hybride considéré : le principe est de fixer le nombre de transitions discrètes effectuées par le système ainsi que les instants de ces transitions. Ainsi la fonction coût ne dépendant plus que des instants de transition, l'objectif est alors de déterminer des instants de transition optimaux ce qui s'exprime sous la forme d'un problème d'optimisation non linéaire classique [84]. Une telle approche se trouve donc limitée par le choix de la suite de modes discrets traversés, question difficile et à notre connaissance non résolue à ce jour.

Une seconde approche repose sur la synthèse d'un contrôle mixte alliant le choix d'une trajectoire optimale discrète dans les modes de l'automate au choix d'une commande conti-

nue dans chacun de ces états. Cette approche basée en général sur l'utilisation du principe du maximum de Pontriaguine a le mérite de poser le problème de façon générale ; cependant les difficultés sont nombreuses, la toute première étant de parvenir à écrire un principe du maximum hybride adapté aux systèmes considérés [78, 71]. De surcroît, la solution proposée n'est en général pas facile à exploiter et des algorithmes numériques restent à développer [85, chapitre 12].

L'approche que nous proposons appartient à la seconde classe de méthodes et repose sur l'utilisation du principe du maximum hybride énoncé au chapitre 6, théorème 6.2.2. Grâce à la structure affine par morceaux du champ de vecteurs hybride considéré, l'idée que nous présentons est d'exploiter les conditions nécessaires d'optimalité données par le principe du maximum dans chaque mode de l'automate hybride considéré, nous permettant ainsi de trouver la forme générale de la commande sous forme de boucle de rétroaction.

Dans ce chapitre, le coût instantané l est supposé indépendant du contrôle u . Nous proposons dans ce chapitre une étude de la structure des trajectoires optimales basée sur l'exploitation du problème d'optimisation posé par le principe du maximum hybride. Nous nous intéressons pour cela dans un premier temps à la résolution locale du problème de contrôle optimal hybride dans un état discret. Cette approche nous permet ensuite de déterminer la forme générale du contrôle optimal cherché en boucle de rétroaction et de proposer un algorithme de simulation des trajectoires extrémales i.e. satisfaisant les conditions du principe du maximum. Pour terminer, nous proposons une piste pour la recherche des trajectoires optimales dans un chemin de cellules adjacentes de l'espace d'état garantissant ainsi leur convergence vers la cible.

7.1 Résolution locale du problème de contrôle hybride

Soit \mathcal{H} un système hybride de la classe des systèmes introduits au chapitre 3. La dynamique de l'automate \mathcal{H} est régie par un système différentiel :

$$\dot{X}(t) = f_h(X(t), u(t)),$$

où le champ f_h est affine par morceaux et continu sur $\mathbb{R}^n \times \mathbb{U}_m$. Nous supposons dans toute cette partie que le coût instantané l ne dépend pas explicitement de la variable de contrôle ; nous écrirons donc $l(X)$ au lieu de $l(X, u)$.

Nous nous intéressons dans cette partie à la structure des solutions optimales du problème de contrôle optimal hybride suivant :

($\mathcal{P}_{\mathcal{H}}$) *Minimiser la fonction coût $J(X_0, u) = \int_0^{t_f} l(X_h(t))dt$ par rapport au contrôle u sous la contrainte :*

$$\begin{cases} \dot{X}_h(t) &= f_h(X_h(t), u(t)) \\ X_h(0) &= X_0 \end{cases}, \quad X(t_f) = 0,$$

sachant que : $\forall t \geq 0, u(t) \in \mathbb{U}_m \subseteq \mathbb{R}^m$. Le temps final t_f n'est pas fixé.

grâce au principe du maximum hybride énoncé dans le paragraphe 6.2.2 du chapitre 6.

Nous définissons donc l'hamiltonien H associé au problème de contrôle $(\mathcal{P}_{\mathcal{H}})$:

$$H(X, u, \lambda) = l(X) + \lambda^T f_h(X, u).$$

D'après le principe du maximum hybride (cf théorème 6.2.2), si un couple (X, u) est une solution optimale du problème $(\mathcal{P}_{\mathcal{H}})$, alors il existe une application non triviale $\lambda : [0, t_f] \rightarrow \mathbb{U}_m$ absolument continue par morceaux telle que pour presque tout $t \in [0, t_f]$:

$$H(X(t), u(t), \lambda(t)) = \min_{v \in \mathbb{U}_m} H(X(t), v, \lambda(t)) \quad (7.1)$$

sous les contraintes :

$$\begin{cases} \dot{X}(t) = \frac{\partial H}{\partial \lambda}(X(t), u(t), \lambda(t)) = f_h(X(t), u(t)), \\ \dot{\lambda}(t) = -\frac{\partial H}{\partial X}(X(t), u(t), \lambda(t)), \\ H(X(t), u(t), \lambda(t)) = 0 \text{ le long de la trajectoire optimale.} \end{cases} \quad (7.2)$$

De plus, aux instants de transition t_i entre deux modes q_i et q_{i+1} , nous devons avoir :

$$\lambda(t_i^+) = \lambda(t_i^-) \quad (7.3)$$

Le problème est donc maintenant de trouver un contrôle admissible qui réalise le minimum de l'hamiltonien H sous les contraintes (7.2) et (7.3). Étant donnée la structure affine par morceaux du champ de vecteurs f_h , l'idée est de chercher à se ramener à un problème d'optimisation dans une cellule $\Delta_{q'}$ de l'espace état-contrôle où f_h est affine. Pour cela, nous traduisons dans un premier temps le problème d'optimisation dans un mode q donné de l'espace d'état. Cette étape de résolution locale nous permet ensuite d'en déduire la forme générale de tout contrôle optimal pour le problème hybride $(\mathcal{P}_{\mathcal{H}})$.

7.1.1 Minimisation de l'hamiltonien dans une cellule D_q

Soit $X_0 \in \mathbb{R}^n$ un point donné, supposé contrôlable jusqu'en 0 par l'automate hybride \mathcal{H} considéré. On note q le mode dans lequel se trouve le système hybride à l'instant initial ce qui signifie donc que :

$$X_0 \in D_q,$$

Nous supposons de plus que l'instant initial t_0 n'est pas un instant de transition du système hybride i.e. que le point X_0 ne se trouve pas sur une garde du système $\mathcal{H} : X_0 \in \overset{\circ}{D}_q$. Cette hypothèse garantit que la trajectoire reste dans la cellule D_q sur un intervalle de temps non trivial quel que soit le contrôle choisi. Dans un premier temps, nous voulons résoudre localement (i.e. dans la cellule D_q) le problème d'optimisation dicté par le principe du maximum hybride.

L'idée est de se ramener à des problèmes d'optimisation affines par rapport au contrôle et donc d'arriver à exprimer le problème de minimisation de l'hamiltonien dans les cellules $\Delta_{q'}$, $q' \in \mathcal{K}(q)$, où la dynamique f_h est affine.

D'après le lemme 3.2.1, le polytope de contrôle \mathbb{U}_m se décompose en sous-domaines de contrôle dépendant de la position X du système :

$$\forall q' \in \mathcal{K}(q), \forall X \in D_q, \mathbb{U}_m = \bigcup_{q' \in \mathcal{K}(q)} U_{q'}(X). \quad (7.4)$$

Remarque 7.1.1. Rappelons que pour un maillage $(\Delta_{q'})_{q' \in I}$ donné, les domaines $U_{q'}(X)$ sont définis de la façon suivante :

$$U_{q'}(X) = \{u \in \mathbb{U}_m; (X, u) \in \Delta_{q'}\}, \quad q' \in \mathcal{K}(q)$$

D'après la proposition 3.2.3 et son corollaire, $U_{q'}(X)$ est soit un m -simplexe soit un point de \mathbb{R}^m . Dans le cas où $X \in \overset{\circ}{D}_q$, $U_{q'}(X)$ est un m -simplexe.

Ainsi, d'après la relation (7.4) et tant que la trajectoire $X(t)$ reste dans la cellule D_q , nous pouvons écrire :

$$\inf_{v \in \mathbb{U}_m} H(X(t), v, \lambda(t)) = \min_{q' \in \mathcal{K}(q)} \left(\min_{v \in U_{q'}(X(t))} H(X(t), v, \lambda(t)) \right).$$

Or tant que : $X(t) \in D_q$ et $v \in U_{q'}(X(t))$, le point $(X(t), v)$ appartient à la cellule $\Delta_{q'}$ de l'espace état-contrôle et le champ f_h au point $(X(t), v)$ est affine. Dans ce cas, l'hamiltonien H est affine par rapport au contrôle :

$$\forall X \in D_q, \forall v \in U_{q'}(X), H(X, v, \lambda) = [l(X) + \lambda^T A_{q'} X + \lambda^T c_{q'}] + \lambda^T B_{q'} v,$$

et le problème de minimisation s'écrit :

$$H(X(t), u(t), \lambda(t)) = \min_{q' \in \mathcal{K}(q)} \left(\min_{v \in U_{q'}(X(t))} [l(X(t)) + \lambda(t)^T A_{q'} X(t) + \lambda(t)^T c_{q'}] + \lambda(t)^T B_{q'} v \right).$$

Localement, dans chaque cellule D_q , la résolution du problème de contrôle optimal hybride se ramène donc à la résolution simultanée de $\text{card } \mathcal{K}(q)$ problèmes d'optimisation affines tout en respectant les contraintes globales (7.2), la solution obtenue dépendant a priori de l'état X et de l'état adjoint λ .

Première minimisation Soit $q' \in \mathcal{K}(q)$ fixé. À un instant t fixé et à la position $X \in \overset{\circ}{D}_q$, le premier problème d'optimisation que nous avons donc à résoudre est le suivant :

$$\text{minimiser } [l(X) + \lambda^T A_{q'} X + \lambda^T c_{q'}] + \lambda^T B_{q'} v \text{ par rapport à } v \in U_{q'}(X).$$

Or le domaine $U_{q'}(X)$ est un m -simplexe de \mathbb{R}^m défini par la donnée de ses $m + 1$ sommets $\sigma_1(X), \dots, \sigma_{m+1}(X)$ affinement dépendants de la position X i.e. de la forme :

$$\forall i = 1, \dots, m + 1, \sigma_i(X) = F_i X + g_i.$$

La fonction à minimiser et les contraintes sur le contrôle étant affines, le problème considéré est un problème de programmation linéaire dont on connaît la solution :

$$v^* = \sigma_i(X) = F_i X + g_i \text{ si } : \forall j \neq i, \lambda^T B_{q'} [(F_i - F_j)X + g_i - g_j] < 0. \quad (7.5)$$

Notons que le choix du contrôle $\sigma_i(X)$ dépend de la valeur de l'état adjoint.

Nous pouvons alors introduire les fonctions de commutation entre les sommets $\sigma_i(X)$ du simplexe $U_{q'}(X)$:

$$S_{i,j} : t \mapsto \lambda(t)^T B_{q'} [(F_i - F_j)X(t) + g_i - g_j]. \quad (7.6)$$

Ainsi, si les zéros des fonctions de commutation $S_{i,j}$ sont des zéros isolés i.e. si pour presque tout $t \in [t_0, t_1]$, il existe un indice $i \in \{1, \dots, m+1\}$ tel que :

$$\forall j \in \{1, \dots, m+1\} - \{i\}, S_{i,j}(t) < 0,$$

alors le contrôle dans la cellule D_q considérée et à la position X est affine par morceaux par rapport à X à valeurs dans l'ensemble des sommets de $U_{q'}(X)$.

Pour des raisons similaires à celles exposées dans le paragraphe 6.4 du chapitre 6, il est de façon générale très difficile et pas toujours possible d'exprimer explicitement les fonctions de commutation $S_{i,j}$ uniquement en fonction de la position $X(t)$ du système.

Dans la suite de ce chapitre, nous nous intéressons donc uniquement à l'étude de la structure générale des extrémales du problème hybride i.e. des couples trajectoires (X, u) qui satisfont le principe du maximum hybride.

Seconde minimisation Une fois résolus les problèmes de minimisation par rapport au contrôle $v \in U_{q'}(X)$ pour tout $q' \in \mathcal{K}(q)$ et en fonction de (X, λ) , il ne reste plus qu'à calculer :

$$\min_{q' \in \mathcal{K}(q)} \{l(X) + \lambda^T [(A + BF_{q'})X + Bg_{q'} + c'_q]\},$$

sachant que l'ensemble $\mathcal{K}(q)$ est fini : $\text{card } \mathcal{K}(q) < +\infty$. Nous en déduisons ainsi le contrôle optimal dans la cellule D_q à la position X .

Ainsi, la minimisation de l'hamiltonien dans un mode q à une position X donnée nous donne (sauf cas singulier) la forme des contrôles optimaux en boucle de rétroaction affines par rapport la position du système i.e. de la forme : $u = FX + g$.

La prochaine étape est donc de vérifier que, dans chaque mode q , ces contrôles satisfont également les conditions nécessaires d'optimalité (7.2) et (7.3) données par le principe du maximum hybride.

7.1.2 Conditions nécessaires locales d'optimalité des contrôles $u = FX + g$

Soit q un mode de l'automate \mathcal{H} . Supposons que nous ayons une trajectoire (X, u) admissible du système hybride \mathcal{H} qui traverse le mode q sur un intervalle de temps $[t_0, t_1]$ non réduit à $\{t_0\}$ et qui vérifie les conditions suivantes :

$$\exists q' \in \mathcal{K}(q), \forall t \in [t_0, t_1], \begin{cases} u|_{[t_0, t_1]}(t) = FX(t) + g \\ (X(t), FX(t) + g) \in \Delta_{q'} \end{cases}. \quad (7.7)$$

Pour alléger les notations, on note : $X = X|_{[t_0, t_1]}$ et $u = u|_{[t_0, t_1]} = FX + g$.

Nous voulons maintenant démontrer que la trajectoire (X, u) dans la cellule $\Delta_{q'}$ considérée vérifie les conditions nécessaires d'optimalité (7.2) et (7.3) fournies par le principe du maximum hybride. Il s'agit donc de montrer qu'il existe une fonction λ absolument continue par morceaux telle que les conditions (7.2) et (7.3) soient satisfaites.

Définition de l'état adjoint

Par hypothèses, sur l'intervalle de temps $[t_0, t_1]$, le contrôle u considéré s'écrit :

$$u = FX + g$$

et la trajectoire état-contrôle (X, u) associée évolue dans la cellule $\Delta_{q'}$ du maillage état-contrôle. Or par construction, le champ de vecteur f_h est affine dans la cellule $\Delta_{q'}$ i.e. de la forme : $f_h(X, u) = AX + Bu + c$. La trajectoire X est donc une solution du système différentiel :

$$\dot{X}(t) = (A + BF)X(t) + Bg + c, \quad t \in [t_0, t_1]. \quad (7.8)$$

Nous définissons alors le paramètre λ comme solution du système :

$$\dot{\lambda}(t)^T = -\frac{\partial l}{\partial X}(X(t)) - \lambda(t)^T(A + BF) \quad (7.9)$$

sur l'intervalle de temps $[t_0, t_1]$. Pour l'instant, aucune condition sur la valeur initiale $\lambda(t_0)$ n'est imposée.

Par hypothèses, la fonction coût l est de classe C^1 sur $\mathbb{R}^n \times \mathbb{U}_m$ et la trajectoire X est continue ; l'application $t \mapsto \frac{\partial l}{\partial X}(X(t))$ est donc également continue, ce qui assure que l'état adjoint λ est absolument continu sur $[t_0, t_1]$.

Remarque 7.1.2. *Sur l'intervalle de temps $[t_0, t_1]$, nous pouvons également calculer explicitement la trajectoire du système :*

$$X(t) = e^{(A+BF)(t-t_0)}X(t_0) + e^{(A+BF)t} \left(\int_{t_0}^t e^{-(A+BF)s} ds \right) (Bg + c),$$

ainsi que celle du vecteur d'état adjoint :

$$\lambda(t)^T = \lambda(t_0)^T e^{-(A+BF)(t-t_0)} - \left(\int_{t_0}^t \frac{\partial l}{\partial X}(X(s)) e^{(A+BF)s} ds \right) e^{-(A+BF)t}.$$

En réutilisant l'astuce donnée par la remarque 5.1.1 du chapitre 5, on peut obtenir une évaluation numérique plus simple des trajectoires $X(t)$ et $\lambda(t)$ sur $[t_0, t_1]$.

Calcul de l'hamiltonien le long de la trajectoire

Vérifions maintenant que l'hamiltonien est nul le long de la trajectoire (X, u, λ) considérée. Afin de simplifier les calculs, le long de cette trajectoire, nous écrivons H sous la forme :

$$H(X(t), u(t), \lambda(t)) = l(X(t)) + \lambda(t)^T [(A + BF)X(t) + Bg + c] = l(X(t)) + \lambda(t)^T \dot{X}(t).$$

D'après les relations (7.8) et (7.9), nous pouvons vérifier que :

$$\ddot{X}(t) = (A + BF)\dot{X}(t),$$

ce qui nous permet maintenant de calculer à tout instant $t \in [t_0, t_1]$:

$$\begin{aligned} \frac{d}{dt}H(X(t), u(t), \lambda(t)) &= \frac{\partial l}{\partial X}(X(t))\dot{X}(t) + \dot{\lambda}(t)^T \dot{X}(t) + \lambda(t)^T \ddot{X}(t) \\ &= \frac{\partial l}{\partial X}(X(t))\dot{X}(t) - \left(\frac{\partial l}{\partial X}(X(t)) - \lambda(t)^T (A + BF) \right) \dot{X}(t) \\ &\quad + \lambda(t)^T (A + BF)\dot{X}(t) \\ &= 0 \end{aligned}$$

Donc : $\forall t \in [t_0, t_1]$, $H(X(t), FX(t) + g, \lambda(t)) = Cte$. Une condition locale nécessaire d'optimalité de la trajectoire $(X, FX + g)$ sur $[t_0, t_1]$ est donc que cette constante soit nulle. Il suffit donc de vérifier que l'hamiltonien s'annule à l'instant initial t_0 , soit :

$$l(X(t_0)) + \lambda(t_0)^T [(A + BF)X(t_0) + Bg + c] = 0. \quad (7.10)$$

Nous obtenons ainsi une contrainte sur l'état adjoint au temps d'entrée dans la cellule Δ_{q^r} .

Remarque 7.1.3. *En utilisant la remarque 7.1.2, sur l'intervalle de temps $[t_0, t_1]$ l'hamiltonien H devient :*

$$\begin{aligned} H(X(t), FX(t) + g, \lambda(t)) &= l(X(t)) + \lambda^T(t_0) [(A + BF)X(t_0) + Bg + c] \\ &\quad - \left(\int_{t_0}^t \frac{\partial l}{\partial X}(X(s)) e^{(A+BF)s} ds \right) e^{-(A+BF)t_0} [(A + BF)X(t_0) + Bg + c], \end{aligned}$$

ce qui nous permet de retrouver la condition nécessaire d'optimalité précédente.

Dans ce paragraphe, nous avons montré que les contrôles de la forme $u = FX + g$ sont de bons candidats locaux pour la recherche de solutions optimales du problème hybride $(\mathcal{P}_{\mathcal{H}})$. Le problème est donc maintenant de reconstruire globalement les trajectoires optimales du système hybride \mathcal{H} en tenant compte de la condition de minimisation (7.1) et des conditions de transversalité aux instants de transition.

7.2 Reconstruction des solutions extrémales candidates à l'optimalité

Dans ce paragraphe, nous voulons déterminer la forme générale des solutions optimales du problème de contrôle hybride $(\mathcal{P}_{\mathcal{H}})$.

Soit (X, u) une solution optimale du problème de contrôle $(\mathcal{P}_{\mathcal{H}})$ issue d'un point initial X_0 donné. Par définition 4.1.2 des solutions d'un système hybride, il existe alors une suite finie ou infinie d'intervalles de temps $([t_i, t_{i+1}])_{i=0\dots r}$ et une suite $\gamma = (q_i)_{i=0\dots r+1}$ de modes discrets associés telles que :

$$\forall t \in]t_i, t_{i+1}[, X(t) \in D_{q_i}.$$

D'après le principe du maximum hybride (cf théorème 6.2.2), nous savons de plus qu'il existe un vecteur d'état-adjoint λ absolument continu par morceaux tels que la trajectoire (X, u, λ) vérifie les conditions (7.1), (7.2) et (7.3).

Les conditions (7.3) sont des conditions de transversalité qui garantissent la continuité de l'état adjoint et de l'hamiltonien $t \mapsto H(X(t), u(t), \lambda(t))$ aux instants de transition t_i . Il est donc suffisant de déterminer la forme du contrôle optimal dans chaque mode q_i traversé par la trajectoire continue X .

Dans la suite de ce paragraphe, nous nous intéressons donc dans un premier temps au calcul du contrôle optimal dans un mode q_i donné. Nous proposons ensuite un algorithme de calcul des trajectoires candidates à l'optimalité du système hybride.

7.2.1 Structure du contrôle optimal dans un mode discret

Soit q_i un mode discret de l'automate hybride \mathcal{H} traversé par la trajectoire optimale (X, u) sur l'intervalle de temps $[t_i, t_{i+1}]$:

$$\forall t \in [t_i, t_{i+1}], X(t) \in D_{q_i}.$$

Dans ce paragraphe, nous nous intéressons à l'expression du contrôle optimal dans un mode q_i traversé par la trajectoire continue X .

Notation. Pour chaque mode q de l'automate \mathcal{H} , nous introduisons l'ensemble $\Sigma_q(X)$ des sommets des domaines de contrôle $U_{q'}(X)$ existant à la position X dans le mode q :

$$\Sigma_q(X) = \{ FX + g ; \exists q' \in \mathcal{K}(q), (FX + g) \text{ est un sommet de } U_{q'}(X) \} \subset \mathbb{U}_m.$$

L'appartenance d'un contrôle u à l'ensemble $\Sigma_q(X)$ se caractérise donc simplement de la façon suivante :

$$u \in \Sigma_q(X) \Leftrightarrow \exists q' \in \mathcal{K}(q), (X, u) \in \Delta_{q'}. \quad (7.11)$$

D'après l'étude menée au paragraphe 7.1.1, le contrôle optimal u dans le mode q_i est la solution d'un problème de programmation linéaire et est déterminé par la relation (7.5) : à tout instant $t \in]t_i, t_{i+1}[$:

$$u(t) = FX(t) + g \in \Sigma_{q_i}(X(t)) \text{ si : } \forall F'X + g' \in \Sigma_{q_i}(X), S_{FX+g, F'X+g'}(t) < 0,$$

en introduisant les fonctions de commutation entre deux contrôles $FX + g$ et $F'X + g'$ de $\Sigma_{q_i}(X)$ définies de la façon suivante :

$$S_{FX+g, F'X+g'}(t) = \lambda(t)^T [f_h(X(t), FX(t) + g) - f_h(X(t), F'X(t) + g')].$$

Remarque 7.2.1. Par extension des résultats présentés dans la partie 6.4, nous appelons arc singulier une solution (X, u, λ) pour laquelle il existe un intervalle de temps non trivial $[T, T + \epsilon[$ sur lequel il existe deux contrôles $FX + g$ et $F'X + g'$ dans l'ensemble $\Sigma_q(X)$ tels que :

$$\forall t \in [T, T + \epsilon[, \begin{cases} \Phi_{FX+g, F'X+g'}(t) = 0 \\ \forall F''X + g'' \in \Sigma_q(X), \Phi_{FX+g, F''X+g''}(t) < 0 \end{cases}$$

Dans le contexte des arcs singuliers, la relation (7.5) ne nous permet donc plus de déterminer la valeur du contrôle optimal cherché.

À partir de maintenant, nous faisons l'hypothèse que ces arcs singuliers n'entrent pas dans la composition des contrôles optimaux. Nous cherchons donc ces contrôles sous la forme de contrôles affines par morceaux par rapport à la position $X(t)$ du système hybride considéré :

Stratégie 1. Dans chaque mode q_i traversé, on cherche un contrôle optimal affine par morceaux par rapport à la position $X(t)$ du système ce qui signifie qu'il existe une subdivision $(\tilde{t}_j)_{j=0\dots N}$ de l'intervalle $]t_i, t_{i+1}[$ telle que :

$$\exists FX + g \in \Sigma_{q_i}(X), \forall t \in [\tilde{t}_j, \tilde{t}_{j+1}[, u(t) = FX(t) + g$$

et le contrôle u ainsi défini minimise l'hamiltonien i.e. pour presque tout $t \in [\tilde{t}_j, \tilde{t}_{j+1}[$, le contrôle $u = FX + g$ vérifie :

$$\forall F'X + g' \in \Sigma_{q_i}(X), S_{FX+g, F'X+g'}(t) < 0,$$

où $X(t)$ désigne la trajectoire associée au contrôle $u = FX + g$ à l'instant t .

Remarque 7.2.2. D'après la relation (7.11), la stratégie 1 revient à chercher la trajectoire (X, u) sur les "arêtes" du maillage état-contrôle (cf figure 7.2.2).

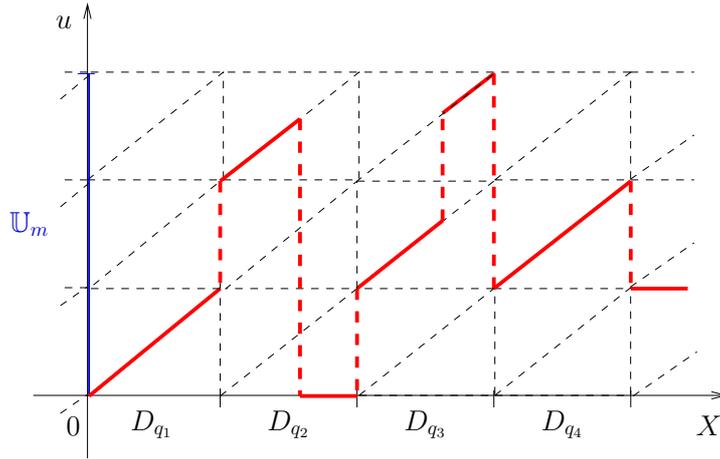


FIG. 7.1 – La trajectoire état-contrôle optimale se déplace selon les arêtes du maillage Δ de $\mathbb{R}^n \times \mathbb{U}_m$.

En effet le choix d'un contrôle $FX + g \in \Sigma_{q_i}(X)$ signifie qu'il existe un indice $q' \in \mathcal{K}(q_i)$ tel que $FX + g$ soit un sommet du simplexe $U_{q_i}(X)$ et donc par définition des sous-domaines $U_{q_i}(X)$ (cf chapitre 3, relation (3.10)), on a nécessairement :

$$\forall t \in [\tilde{t}_j, \tilde{t}_{j+1}[, (X(t), FX(t) + g) \in \Delta_{q'}.$$

Concrètement, selon la stratégie 1, la recherche du contrôle optimal dans le mode q_i s'effectue à tout instant \tilde{t}_j en deux temps :

1. on cherche le contrôle $u = FX + g \in \Sigma_{q_i}(X)$ optimal à partir de l'instant \tilde{t}_j ,
2. on calcule le premier instant $\tilde{t}_{j+1} > \tilde{t}_j$ auquel $u = FX + g$ n'est plus optimal.

Une difficulté de la mise en place d'une telle stratégie réside dans le calcul de l'état adjoint $\lambda(t)$ à un instant t donné, indispensable pour résoudre le problème de minimisation de l'hamiltonien. En effet, les équations d'Euler-Lagrange satisfaites par le paramètre λ dépendent du choix du contrôle et le contrôle optimal dépend à son tour de λ . Il est donc nécessaire d'obtenir une condition permettant de déterminer localement le contrôle optimal sans avoir à calculer la trajectoire.

Nous utilisons pour cela la condition (7.2) du principe du maximum hybride : d'après la stratégie 1, nous cherchons sur l'intervalle de temps $[\tilde{t}_j, \tilde{t}_{j+1}[$, un contrôle u de la forme :

$$u(t) = FX(t) + g \text{ avec } FX + g \in \Sigma_{q_i}(X)$$

et qui réalise le minimum de l'hamiltonien $v \mapsto H(X(t), v, \lambda(t))$ sur le domaine de contrôle \mathbb{U}_m .

Par définition de l'ensemble $\Sigma_{q_i}(X)$, il existe un indice $q' \in \mathcal{K}(q_i)$ tel que :

$$(X(t), FX(t) + g) \in \Delta_{q'}.$$

La dynamique du système hybride \mathcal{H} étant affine dans la cellule $\Delta_{q'}$ de la forme $\dot{X} = A_{q'}X + B_{q'}u + c_{q'}$, nous en déduisons que la trajectoire X considérée est solution du système :

$$\dot{X}(t) = (A_{q'} + B_{q'}F)X(t) + B_{q'}g + c_{q'}$$

sur l'intervalle de temps $[\tilde{t}_j, \tilde{t}_{j+1}[$. D'après les résultats obtenus au paragraphe 7.1.2, une condition nécessaire d'optimalité du contrôle $u = FX + g$ sur l'intervalle de temps $[\tilde{t}_j, \tilde{t}_{j+1}[$ est la suivante :

$$l(X(\tilde{t}_j)) + \lambda(\tilde{t}_j)^T [(A_{q'} + B_{q'}F)X(\tilde{t}_j) + B_{q'}g + c_{q'}] = 0 \quad (7.12)$$

En conclusion, dans le mode q_i , à l'instant initial \tilde{t}_j et pour une condition initiale $\lambda(\tilde{t}_j)$ donnée, le contrôle optimal est de la forme :

$$u(\tilde{t}_j) = FX(\tilde{t}_j) + g \in \Sigma_{q_i}(X(\tilde{t}_j))$$

sachant qu'il existe un indice $q' \in \mathcal{K}(q_i)$ tel que :

$$\begin{cases} (X(\tilde{t}_j), u(\tilde{t}_j)) \in \Delta_{q'} \\ \forall F'X + g' \in \Sigma_{q_i}(X(\tilde{t}_j)), S_{FX+g, F'X+g'}(\tilde{t}_j) \leq 0 \\ l(X(\tilde{t}_j)) + \lambda(\tilde{t}_j)^T [(A_{q'} + B_{q'}F)X(\tilde{t}_j) + B_{q'}g + c_{q'}] = 0 \end{cases}$$

Ainsi $t \mapsto FX(t) + g$ est le contrôle optimal cherché et il ne reste donc plus qu'à déterminer le premier instant \tilde{t}_{j+1} où $FX + g$ n'est plus optimal ce qui signifie qu'il existe un contrôle $F'X + g' \in \Sigma_{q_i}(X)$ différent de $FX + g$ qui devient optimal à l'instant \tilde{t}_{j+1} . Nous pouvons alors caractériser \tilde{t}_{j+1} de la façon suivante :

$$\tilde{t}_{j+1} = \inf \{ t > \tilde{t}_j ; s \mapsto S_{FX+g, F'X+g'}(s) \text{ s'annule et change signe au point } t \}.$$

7.2.2 Algorithme de simulation et problème d'initialisation de l'état adjoint

Jusqu'à présent, l'étude des conditions nécessaires d'optimalité du principe du maximum nous a permis de caractériser les contrôles optimaux du modèle hybride construit au chapitre 3 sous la forme de contrôles en boucle de rétroaction fermée affines par morceaux par rapport à la position du système. Nous nous intéressons maintenant à la simulation des trajectoires vérifiant les conditions nécessaires d'optimalité du principe du maximum hybride pour une condition initiale λ_0 donnée de l'état adjoint.

Simulation des trajectoires vérifiant le principe du maximum

Pour une condition initiale λ_0 donnée de l'état adjoint, l'algorithme de simulation 13 calcule une trajectoire qui vérifie les conditions nécessaires d'optimalité du principe du maximum hybride. Cet algorithme est défini récursivement à partir de l'algorithme 12 qui calcule la trajectoire cherchée dans chaque cellule D_{q_i} traversée.

On note t_i le temps d'entrée dans la cellule D_{q_i} au point X_i avec la condition initiale : $\lambda(t_i) = \lambda_i$. Le principe de l'algorithme de simulation dans le mode q_i est le suivant :

1. On calcule le contrôle optimal $u^*(t_i) = FX_i + g \in \Sigma_{q_i}(X_i)$ à l'instant t_i relativement à la condition λ_i i.e. tel que :

$$\begin{cases} \forall F'X + g' \in \Sigma_{q_i}(X(\tilde{t}_j)), S_{FX+g, F'X+g'}(\tilde{t}_j) \leq 0 \\ l(X_i) + \lambda_i^T f_h(X_i, FX_i + g) = 0 \end{cases}$$

2. On en déduit les trajectoires $X[X_i, FX + g]$ et $\lambda[X_i, FX + g]$ issues respectivement des points X_i et λ_i selon le contrôle $u^* = FX + g$:

$$\dot{X}(t) = f_h(X(t), FX(t) + g)$$

$$\dot{\lambda}(t)^T = -\frac{\partial H}{\partial X}(X[X_i, FX + g](t), FX[X_i, FX + g](t) + g, \lambda(t))$$

en posant : $H(X, u, \lambda) = l(X) + \lambda^T f_h(X, u)$.

3. On calcule alors le temps de sortie t_{i+1} de la trajectoire $X[X_i, FX + g]$ de la cellule D_{q_i} ainsi que la cellule $D_{q_{i+1}}$ vers laquelle se dirige $X[X_i, FX + g]$ à l'instant t_{i+1} .
4. On vérifie que la trajectoire $X[X_i, FX + g]$ est optimale sur l'intervalle de temps $[t_i, t_{i+1}]$: pour tout contrôle $v = F'X + g' \in \Sigma_{q_i}(X)$,

- (a) on définit la fonction de commutation du contrôle $u^* = FX + g$ vers le contrôle v :

$$S_v := t \mapsto H(X[X_i, u^*](t), u^*(t), \lambda[X_i, u^*](t)) - H(X[X_i, u^*](t), v(t), \lambda[X_i, u^*](t)),$$

où $v(t) = F'X[X_i, FX + g](t) + g'$ tant que $u^* = FX + g$ est optimal.

(b) Soit $t_{temp} > t_i$ le premier temps tel que :

$$S_v(t_{temp}) = 0 \text{ et } S_v(t) \text{ change de signe en } t = t_{temp}.$$

Si t_{temp} est bien défini et $t_{temp} < t_{i+1}$ alors une commutation se produit à l'instant t_{temp} du contrôle $FX + g$ vers $v = F'X + g'$ (ligne 21 : *commut := true* de l'algorithme 12) et :

$$t_{i+1} := t_{temp} ; u^* := v ;$$

5. On itère alors cet algorithme à partir de l'étape 2. avec comme conditions initiales dans le mode q_i :

$$t_{temp}, X[X_i, u^*](t_{temp}), \lambda[X_i, u^*](t_{temp}),$$

jusqu'à ce que la trajectoire X calculée étape par étape sorte de la cellule D_{q_i} (soit dans l'algorithme 12 : *commut := false*).

Remarque 7.2.3. Afin d'alléger l'écriture de l'algorithme 12, nous avons omis d'écrire les étapes de calcul de l'approximation affine f_h . En fait, à chaque contrôle $FX + g \in \Sigma_{q_i}(X)$ est associée une cellule $\Delta_{q'}$ de l'espace état-contrôle (gardée en mémoire lors de la construction de l'ensemble $\Sigma_{q_i}(X)$) sur laquelle nous calculons l'approximation affine grâce à l'algorithme donné au chapitre 2 :

$$f_h := (X, u) \mapsto HybridDynamic(n, f, \Delta_{q'}).$$

L'algorithme général de simulation des trajectoires candidates à l'optimalité est alors basé sur le calcul dans les états discrets q_i traversés de la trajectoire vérifiant les conditions d'optimalité et sur les conditions de transversalité qui garantissent la continuité de l'état adjoint λ .

Le principe est le suivant : partant d'une condition initiale (t_0, X_0, λ_0) donnée dans un mode q_0 , on calcule la trajectoire dans q_0 vérifiant les conditions d'optimalité grâce à l'algorithme 13 :

$$(t_1, q_1, X_1, \lambda_1, u^*) := OptimalSimulationMode(n, t_0, q_0, \Sigma_{q_0}(X), X_0, \lambda_0, f, l)$$

La continuité de l'état et de l'état adjoint nous donne alors une nouvelle condition initiale (t_1, X_1, λ_1) dans le mode q_1 et ainsi de suite.

Le principe du maximum ne nous donnant que des conditions nécessaires d'optimalité, nous ne sommes pour l'instant pas en mesure de garantir la convergence des trajectoires ainsi obtenues vers la cible 0. Par conséquent, nous fixons le nombre d'itérations de l'algorithme 12 dans l'algorithme 13 de calcul des trajectoires candidates à l'optimalité.

Problème de l'initialisation de l'état adjoint

L'approche globale de la résolution des problèmes de contrôle hybride par le principe du maximum hybride nous a permis d'obtenir la forme générale du contrôle optimal cherché. Nous avons vu qu'à toute condition initiale λ_0 de l'état adjoint est associée une trajectoire (X, u) vérifiant les conditions nécessaires d'optimalité du principe du maximum hybride. Le point clé est donc de trouver une approximation du point λ_0 qui donne une trajectoire optimale du problème de contrôle hybride.

Algorithme 12 OptimalSimulationMode : Simulation de trajectoires vérifiant les conditions nécessaires d'optimalité dans un mode discret

Données : n dimension de l'espace d'état, $\tilde{t}_j, q_i, \Sigma_{q_i}(X), X_j, \lambda_j$ conditions initiales, f, l .

Sortie : $t_{i+1}, q_{i+1}, X_{i+1}, \lambda_{i+1}, u^*$ les conditions de sortie de la trajectoire du mode q_i vers le mode q_{i+1} au temps t_{i+1} et à la position X_{i+1} .

```

1: commut := true ;
2:  $\tilde{t}_0 := t_i$  ;
3:  $H := (X, u, \lambda) \mapsto l(X, u) + \lambda^T f_h(X, u)$  ;
4: tant que commut := true faire
5:   {Initialisation :}  $u^* := \Sigma_{q_i}(X)[1]$  ;
6:   {Calcul du contrôle optimal à l'instant  $\tilde{t}_j$  :}
7:   pour  $FX + g \in \Sigma_{q_i}(X)$  faire
8:     si  $H(X_j, u^*, \lambda_j) \geq H(X_j, FX_j + g, \lambda_j)$  alors
9:        $u^* := FX + g$  ;
10:    fin si
11:  fin pour
12:   $X[X_j, u^*] :=$  solution du système :

```

$$\dot{X}(t) = f_h(X(t), FX(t) + g), \quad X(\tilde{t}_j) = X_j.$$

```

13:   $\lambda[X_j, u^*] :=$  solution du système :

```

$$\dot{\lambda}(t)^T = -\frac{\partial H}{\partial X}(X[X_j, u^*](t), u^*(t), \lambda(t)), \quad \lambda(\tilde{t}_j) = \lambda_j.$$

```

14:  {Calcul du temps de sortie de  $X[X_j, u^*]$  de la cellule  $D_{q_i}$  vers la cellule  $D_{q_{i+1}}$  :}
    ( $\tilde{t}_{j+1}, F$ ) :=  $OutCell(\mathcal{H}, q_i, X_j, u^*)$  ; (cf algorithme 6 du chapitre 5)
     $D_{q_{i+1}}$  := cellule de l'espace d'état telle que :  $G_{(q_i, q_{i+1})} = F$  ;
15:  {Calcul du temps  $\tilde{t}_{j+1}$  jusqu'au quel  $u^*$  est optimal :}
    commut := false ;
16:  pour  $v \in \Sigma_{q_i}(X)$  tel que  $v \neq u^*$  faire
17:    {Fonction de commutation :}
     $S_v := t \mapsto H(X[X_j, u^*](t), u^*(t), \lambda[X_j, u^*](t)) - H(X[X_j, u^*](t), v(t), \lambda[X_j, u^*](t))$  ;
18:     $t_{temp} := \inf\{t > \tilde{t}_j ; S_v(t) = 0 \text{ et } \dot{S}_v(t) \text{ ne change pas de signe}\}$  ;
19:    si  $t_{temp} < \tilde{t}_{j+1}$  alors
20:       $\tilde{t}_{j+1} := t_{temp}$  ;
21:      commut := true ;
22:    fin si
23:  fin pour
24:   $X_{j+1} := X[X_j, u^*](\tilde{t}_{j+1})$  ;  $\lambda_{j+1} := \lambda[X_j, u^*](\tilde{t}_{j+1})$  ;
25:   $j := j + 1$  ;
26: fin tant que
27: Retourner  $(\tilde{t}_j, q_{i+1}, X_j, \lambda_j, u^*)$ .

```

Algorithme 13 OptimalSimulation : Simulation de trajectoires vérifiant les conditions nécessaires d'optimalité

Données : n dimension de l'espace d'état, X_0, λ_0 conditions initiales sur l'état et l'état adjoint, f, l, N .

{On fixe pour le moment le nombre N de modes discrets traversés}

Sortie : (τ, γ, X) une exécution du système hybride qui vérifie les conditions nécessaires d'optimalité du principe du maximum hybride.

1: $t_0 := 0$; $D_{q_0} := CurrentMode(h, X_0)$;

2: **pour** i variant de 0 à N **faire**

3: $\Sigma_{q_i}(X) :=$ ensemble des sommets des domaines de contrôles $U_{q'}(X)$, $q' \in \mathcal{K}(q_i)$ (cf paragraphe 3.2.3).

4: $(t_{i+1}, q_{i+1}, X_{i+1}, \lambda_{i+1}, u_{i+1}) := OptimalSimulationMode(t_i, q_i, \Sigma_{q_i}(X), X_i, \lambda_i, f, l)$;

5: **fin pour**

6: $\tau := ([t_i, t_{i+1}])_{i=0\dots N}$; $\gamma := (q_i)_{i=0\dots N+1}$; $X := (X_i)_{i=0\dots N+1}$;

7: Retourner (τ, γ, X) ;

Une première chose consiste déjà à sélectionner les valeurs λ_0 pour lesquels il existe une trajectoire vérifiant les caractéristiques établies depuis le début de ce chapitre. Typiquement, pour une valeur initiale λ_0 de l'état adjoint arbitraire, il n'existe pas forcément de contrôle $FX + g$ qui vérifie les conditions nécessaires d'optimalité.

Une idée serait donc dans un premier temps d'arriver à restreindre l'ensemble des conditions initiales possibles de l'état adjoint (cf exemple 7.2.1).

Exemple 7.2.1 (Contrôle optimal d'un ressort non linéaire en temps minimum).

Reprenons l'exemple d'un ressort non linéaire (cf exemple 5.3.3) :

$$\begin{cases} \dot{x}(t) = y(t) \\ \dot{y}(t) = -x(t) - 2x(t)^3 + u(t) \end{cases} \quad (7.13)$$

Nous nous intéressons au contrôle optimal du système considéré en temps minimum.

D'après les éléments de résolution du problème de contrôle non linéaire démontrés dans [81], nous savons que l'état adjoint est nécessairement de la forme :

$$\lambda = [\cos(\alpha), \sin(\alpha)].$$

L'idée que nous proposons consiste à utiliser l'algorithme 12 pour calculer pour un λ donné le contrôle u_0 qui réalise le minimum de l'hamiltonien $H(X_0, v, \lambda_0)$ sur \mathbb{U}_m (cf étape 6 à 11 de l'algorithme 12). Si la valeur initiale de λ est bien choisie, alors l'hamiltonien $H(X_0, u_0, \lambda_0)$ est nul.

Les valeurs initiales de λ susceptibles de nous donner une trajectoire optimale sont donc celles qui annulent l'hamiltonien. Nous traçons donc la fonction :

$$\alpha \mapsto H(X_0, u_0, [\cos(\alpha), \sin(\alpha)]), \quad \alpha \in [0, 2\pi]$$

sur une période ce qui nous donne alors les valeurs possibles du paramètre α (cf figure 7.2).

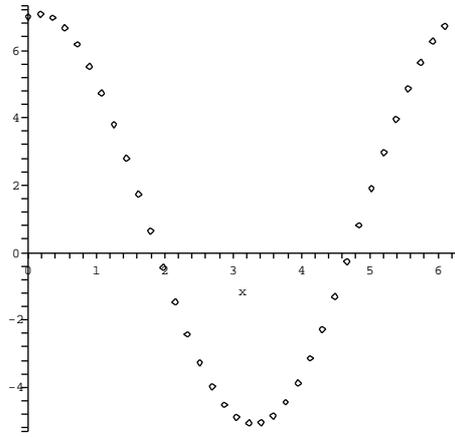


FIG. 7.2 – Tracé de l’hamiltonien H en fonction de $\alpha \in [0, 2\pi]$ où $X_0 = [0, 6]$ et $h = 1$. Les valeurs de α qui annulent H sont : $\alpha = \frac{3\pi}{2} \approx 2,094395103$ ou $\alpha \approx 1,901093561$ sachant que la valeur optimale de α pour le problème non linéaire est : $\alpha \approx 2,136718750$ (cf [81]).

7.3 Une autre approche : recherche de trajectoires optimales dans un chemin de cellules

D’après l’étude menée jusqu’à présent, l’approche globale des problèmes de contrôle optimal hybride ne nous permet donc pas pour l’instant d’assurer la convergence et l’optimalité des trajectoires calculées par l’algorithme 13. Dans cette partie, nous proposons une autre piste pour la recherche de solutions optimales des problèmes de contrôle hybride garantissant la convergence des trajectoires calculées vers la cible.

L’idée est de déterminer une suite $\gamma = (q_i)_{i=0\dots r}$ de modes discrets de sorte que le domaine Ω_γ associé contienne au moins une trajectoire optimale du problème de contrôle hybride puis d’appliquer l’algorithme de simulation 13.

Soit $\gamma = (q_i)_{i=0\dots r}$ un chemin de modes discrets de l’automate hybride considéré. On introduit le domaine :

$$\Omega_\gamma = \bigcup_{i=0}^r D_{q_i}.$$

Nous supposons également : $X_0 \in \Omega_\gamma$ et $0 \in \Omega_\gamma$. Nous obtenons par cette approche le problème de contrôle optimal avec contraintes sur l’état suivant :

$\mathcal{P}_H(\Omega_\gamma)$ Minimiser la fonction coût $J(X_0, u) = \int_0^{t_f} l(X_h(t))dt$ par rapport au contrôle u sous

les contraintes :

$$\begin{cases} \dot{X}_h(t) &= f_h(X_h(t), u(t)) \\ X_h(0) &= X_0 \end{cases}, \quad X(t_f) = 0,$$

$$\forall t \geq 0, X(t) \in \Omega_\gamma.$$

sachant que : $\forall t \geq 0, u(t) \in \mathbb{U}_m \subseteq \mathbb{R}^m$. Le temps final t_f n'est pas fixé.

La résolution du problème $\mathcal{P}_{\mathcal{H}}(\Omega_\gamma)$ se décompose alors en deux temps :

1. Écrire un principe du maximum hybride adapté aux systèmes hybrides avec contraintes d'état.
2. Calculer, si elle existe, une trajectoire optimale du problème $\mathcal{P}_{\mathcal{H}}(\Omega_\gamma)$.

Il s'agit ensuite de vérifier que si la trajectoire optimale globale est dans le domaine Ω_γ considéré, alors la trajectoire solution du problème $\mathcal{P}_{\mathcal{H}}(\Omega_\gamma)$ calculée est effectivement globalement optimale.

Remarque 7.3.1. *L'écriture d'un principe du maximum prenant en compte des contraintes sur l'état est un problème délicat et difficile. Entre autres, la présence de contraintes d'état peut introduire des discontinuités dans l'état adjoint. Dans [48], R. Hartl, S.P. Sethi et R.G. Vickson présentent des travaux très complets sur les différentes formes du principe du maximum avec contraintes d'état.*

Dans ce chapitre, nous avons abordé le problème du calcul des trajectoires optimales des problèmes de contrôle associés au modèle hybride du chapitre 3. L'exploitation des conditions nécessaires d'optimalité du principe du maximum hybride nous a permis d'obtenir la forme générale du contrôle optimal en boucle de rétroaction affine par morceaux par rapport à l'état continu du système considéré. Ce résultat capital est une première avancée vers la résolution complète des problèmes de contrôle optimal hybrides et la mise en place d'une méthode de tir hybride.

Chapitre 8

Réalisation logicielle et application

Ce chapitre est consacré à la présentation des outils algorithmiques développés en Maple pour la modélisation et l'étude des problèmes de contrôle non linéaires par des méthodes de calcul hybride. Dans la partie 8.1, nous décrivons les principales fonctionnalités des bibliothèques développées. Dans la partie 8.2, nous appliquons ces outils au problème du transfert orbital d'un satellite.

8.1 Réalisation logicielle en Maple

Nous avons développé sous Maple un ensemble de bibliothèques¹ permettant la mise en oeuvre des méthodes et algorithmes présentés tout au long de ce manuscrit. Cette première partie est consacrée à la description de leurs principales fonctionnalités.

8.1.1 Bibliothèque

Tout notre travail d'implémentation repose sur la représentation et la manipulation efficaces de polytopes en toute dimension. Nous avons ensuite développé différents niveaux d'outils permettant tout d'abord de construire et simuler le modèle hybride défini au chapitre 3 de ce manuscrit et enfin d'aborder la résolution des problèmes de contrôle (cf parties II et III).

La figure 8.1 donne une vue d'ensemble de l'organisation de ces bibliothèques entre elles. Dans la suite de cette première partie, nous décrivons les principales fonctionnalités de ces bibliothèques et donnons les entrées qui permettront à l'utilisateur de les manipuler.

Remarque 8.1.1. *Nous présentons dans ce chapitre la version générique des procédures implémentées. En effet, certaines comportent des options permettant de faire appel à des routines numériques. Cette possibilité se révèle particulièrement utile pour les exemples où le calcul des temps de sortie des modes de l'automate hybride considéré ne peut être effectué de façon satisfaisante.*

¹Ces bibliothèques ont été développées en Maple version 9.5 mais fonctionnent également sous la version 9. Elles sont disponibles à l'adresse : <http://www-lmc.imag.fr/lmc-mosaic/Jean-Guillaume.Dumas/SHOC>

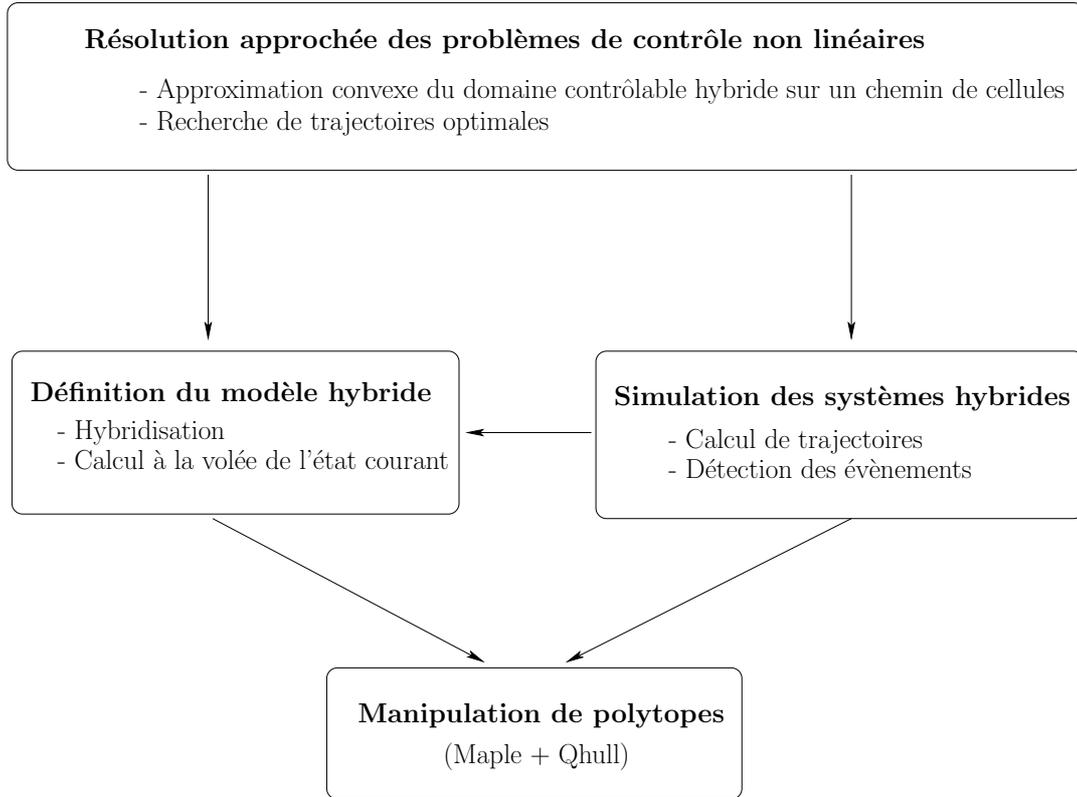


FIG. 8.1 – Organisation des bibliothèques Maple pour la résolution hybride des problèmes de contrôle non linéaires (les flèches indiquent les composantes sur lesquelles s'appuient ces bibliothèques).

Manipulation de polytopes en toute dimension

Combinant les fonctionnalités de Maple 9 ou 9.5 et celles de *Qhull* [6], la bibliothèque *GeomTools.lib* propose un ensemble de fonctions élémentaires permettant la représentation et la manipulation efficaces de polytopes en toute dimension.

Un polytope P en dimension n est représenté par la liste de ses sommets s_1, \dots, s_p , eux-mêmes représentés par la liste de leurs coordonnées :

$$P = [[s_{11}, \dots, s_{1n}], [s_{21}, \dots, s_{2n}], \dots, [s_{p1}, \dots, s_{pn}]].$$

Parmi les différentes fonctionnalités proposées, citons en particulier le calcul d'enveloppes convexes et le maillage (avec ajout de sommets) ou la triangulation (sans ajout de sommets) d'un polytope donné :

$L := QhullConvexHull(pointlist)$ **ou** $L := DegenerateConvexHull(pointlist, F)$

Procédures qui calculent la liste des sommets de l'enveloppe convexe d'une liste de points donnée, obtenue en interfaçant le logiciel *Qhull* avec Maple. La première permet le calcul d'une enveloppe convexe de dimension n d'un nuage de points dans \mathbb{R}^n ; la

seconde permet le calcul de l'enveloppe convexe de points contenus dans un hyperplan support d'un polytope F .

$L := \text{Triangulation}(\text{pointlist})$

Procédure qui interface *Qhull* et Maple et calcule une triangulation d'un polytope donné sans ajout de sommets. Elle renvoie la liste des simplexes obtenus.

$L := \text{Mesh}(\text{pointlist}, h)$

Procédure qui calcule un maillage de taille h d'un polytope défini par la liste de ses sommets *pointlist* et renvoie la liste des simplexes constituant le maillage.

$v := \text{ExitNormalVector}(F, D_q)$

Procédure qui calcule un vecteur normal à la face F du polytope D_q et dirigé vers l'extérieur de D_q .

Définition du modèle hybride

Une deuxième librairie *HybridSystem.lib* implémente les algorithmes des chapitres 2 et 3 nécessaires pour la modélisation des systèmes de contrôle non linéaires par les systèmes hybrides affine par morceaux définis au chapitre 3. Cette librairie permet entre autres le calcul à tout instant de l'état du modèle hybride sous-jacent en déterminant la cellule courante et les cellules de contrôle possibles.

Les procédures essentielles sont les suivantes :

$f_h := \text{Hdyn}(n, f, \text{cell})$

Procédure qui calcule une approximation affine du champ f par interpolation aux sommets du simplexe *cell* et qui renvoie :

$$f_h : (X, u) \mapsto AX + Bu + c \quad \text{où } [A, B, c] := \text{LinearApproximation}(n, m, f, \text{vtx}).$$

Si $m = 0$, la procédure *LinearApproximation* permet d'approcher un champ de vecteurs non contrôlé $X \mapsto f(X)$ par un champ affine : $X \mapsto AX + c$ et renvoie le couple $[A, c]$.

$D_q := \text{Hstate}(x_0, h)$

Procédure qui calcule la cellule D_q du maillage de pas h de l'espace d'état qui contient le point x_0 .

$\text{CellList} := \text{ControlCell}(n, D_q, \mathbb{U}_m, h)$

Procédure qui dans un mode q donné calcule la liste des cellules de l'espace état contrôle qui se projette exactement sur D_q .

$\text{ControlDomain} := \text{ControlSet}(n, \Delta_{q'})$

Procédure qui calcule la liste des sommets du domaine de contrôle $U_{q'}(X)$ en fonction de la position X du système dans la cellule $\Delta_{q'}$: ce sont les sommets de l'intersection de $\Delta_{q'}$ avec le sous-espace affine $\{(X, u) ; u \in \mathbb{U}_m\}$ (cf chapitre 3, paragraphe 3.2.3).

Simulation des systèmes hybrides

La librairie *Simulation.lib* implémente les algorithmes de simulation des systèmes hybrides décrits au chapitre 5, paragraphe 5.1. Les principales fonctions à disposition sont le test des conditions d'entrée dans un mode discret q du modèle hybride, le calcul de trajectoires et celui du temps et de l'état du système hybride à la sortie du mode considéré.

Nous rappelons que f_h désigne l'approximation affine par morceaux du champ non linéaire f construite sur un maillage de pas $h > 0$ et \mathcal{H} le système hybride associé approchant le système non linéaire $\dot{X} = f(X, u)$. Les procédures essentielles sont les suivantes :

$[t_s, F] := ExitTime(n, Y, D_q)$

Procédure qui calcule le temps de sortie t_s et la face de sortie F de la trajectoire Y du polytope $D_q \subset \mathbb{R}^n$.

$b := InsideDq(n, f, u, D_q, \mathbb{U}_m, X_0, h)$

Procédure qui teste si la trajectoire du système hybride construit sur un maillage de pas $h > 0$ issue de X_0 selon le contrôle u entre dans la cellule $D_q \subset \mathbb{R}^n$. \mathbb{U}_m désigne le polytope de contrôle.

$Simul(f, X_0, u, \mathbb{U}_m, h, N)$

Procédure qui calcule la trajectoire du modèle hybride issue du point X_0 selon le contrôle u sur N itérations i.e. dans N modes discrets adjacents.

Résolution approchée des problèmes de contrôle non linéaires

Grâce aux bibliothèques décrites précédemment, nous sommes maintenant en mesure de définir des bibliothèques permettant d'une part le calcul du domaine contrôlable hybride sur des chemins de cellules (cf chapitre 5), d'autre part le calcul d'extrémales des problèmes de contrôle optimal hybrides. Les principales fonctions implémentées sont les suivantes :

1. Librairie *ConvexApprox.lib* d'approximation du domaine contrôlable hybride

$approx := ConvexApprox(n, f, \mathbb{U}_m, \gamma, LFace, InitPoint, h)$

Procédure qui calcule une approximation convexe du domaine contrôlable hybride à la cible *InitPoint* dans le chemin de cellules $\gamma = (q_i)_{i=0\dots r}$ et renvoie la liste des approximations convexes contrôlables le long du chemin :

$$approx = [[X_{q_0,1}, \dots, X_{q_0,k_0}], [X_{q_1,1}, \dots, X_{q_1,k_1}], \dots, [X_{q_r,1}, \dots, X_{q_r,k_r}]].$$

Le paramètre *LFace* représente la liste des faces des cellules du chemin. S'il s'agit de cellules simpliciales, il suffit de prendre : $LFace := []$. La cible *InitPoint* peut être un point (0 comme dans tout ce manuscrit) ou bien un polytope quelconque.

$ConvexApproxplot(n, f, U, q, LFace, InitPoint, h)$

Procédure qui permet la représentation graphique de l'approximation calculée par l'algorithme précédent en dimensions 2 et 3.

2. Librairie *HybridSolving.lib* de recherche de solutions optimales des problèmes de contrôle hybrides

EulerLagrange($n, l, f, X, u, lambda$)

Procédure qui renvoie le second membre des équations d'Euler-Lagrange vérifiées par le vecteur d'état adjoint :

$$\dot{\lambda}(t) = EulerLagrange(n, l, f, X(t), u(t), lambda(t)).$$

ExtremalTrajectory($X_0, D_{q_0}, \lambda_0, f, l, \mathbb{U}_m, h, N, eps$)

Procédure qui calcule une extrémale (X, u, λ) du problème de contrôle optimal hybride pour la condition initiale (X_0, λ_0) . Le paramètre N fixe le nombre de modes discrets traversés de l'automate hybride et eps est un paramètre de précision réel pour le calcul des temps de sortie.

8.1.2 Quelques détails techniques

Une fois les bibliothèques *GeomTools.lib* et *Simulation.lib* implémentées, les difficultés se situent davantage au niveau des méthodes de résolution en théorie du contrôle optimal qu'au niveau de l'implémentation. Le plus difficile a donc été de mettre au point tout un ensemble de procédures permettant de construire et de manipuler de façon efficace le modèle hybride d'un système de contrôle non linéaire.

En effet que ce soit pour l'approximation du domaine contrôlable hybride ou pour le calcul d'extrémales du problème de contrôle hybride, les algorithmes présentés comportent dans chaque mode discret traversés,

1. une étape d'hybridisation : on calcule l'état courant, la ou les cellules de contrôle à considérer et l'approximation affine du champ non linéaire
2. une étape de calcul de trajectoires (conséquence dans le contexte des algorithmes d'approximation du domaine contrôlable)
3. une étape de calcul des conditions de sortie qui va nous permettre d'initialiser la prochaine itération.

dont le coût est récapitulé dans le tableau suivant :

Modélisation hybride	- Calcul de l'état courant	$\mathcal{O}(n \log n)$
	- Calcul des cellules de contrôle possibles	$\mathcal{O}(\frac{1}{h}m^3)$
	- Approximation affine dans une cellule	$\mathcal{O}(n^2(n + m))$
Simulation du modèle hybride	- Calcul de trajectoires	$\mathcal{O}(n^3)$
	- Calcul des temps de sortie	

Le coût des algorithmes d'approximation du domaine contrôlable et de simulation de trajectoires extrémales étant linéaire dans le nombre de modes discrets traversés, il est donc

crucial dans la perspective de simulations en grande échelle d'avoir à notre disposition des outils efficaces et rapides.

Afin de tester au plus tôt nos algorithmes de contrôlabilité et de calcul d'extrémales, nous avons été amenés à faire quelques compromis dans l'implémentation d'outils secondaires mais indispensables. Certains points sont donc à améliorer tels que :

L'algorithme de maillage du polytope de contrôle

Jusqu'à présent, nous nous sommes contentés d'un algorithme rudimentaire de maillage du polytope de contrôle, cf paragraphe 3.1.2. L'idée de cet algorithme est de placer des points régulièrement répartis dans le polytope que l'on souhaite mailler puis d'appliquer une triangulation de *Qhull* sur ces points. Cet algorithme se révèle extrêmement coûteux mais n'est heureusement effectué qu'une seule fois par pas h considéré au début de l'algorithme général de résolution.

Calcul des faces d'un polytope quelconque

L'implémentation des algorithmes de simulation des systèmes hybrides fait indirectement intervenir les faces des cellules D_q et $\Delta_{q'}$. Dans le cas de simplexes comme c'est le cas pour le modèle hybride construit au chapitre 3, la liste des faces d'une cellule s'obtient facilement de la façon suivante : soit $P = [s_1, \dots, s_{n+1}]$ un simplexe de \mathbb{R}^n . Les faces de P sont définies de la façon suivante :

$$F_i = [s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_{n+1}], \quad i = 1 \dots, n + 1.$$

Pour l'instant, l'utilisateur doit entrer manuellement les polytopes ainsi que la liste de leurs faces. Il serait donc intéressant d'implémenter un algorithme qui effectue de façon automatique le calcul des faces d'un polytope donné en dimension quelconque.

Calcul des temps de sortie

Il s'agit d'un problème inhérent à la définition des systèmes hybrides en général. Le problème est d'arriver à calculer de façon satisfaisante le temps de sortie $t_s > 0$ d'une trajectoire donnée d'un mode q considéré i.e. :

$$t_s = \inf\{t > 0 ; \phi(t) = 0\} \quad \text{où } \phi : \mathbb{R} \rightarrow \mathbb{R}.$$

Nous avons vu au chapitre 5 que le problème est d'arriver à déterminer la plus petite racine positive de la fonction ϕ . La meilleure solution nous semble actuellement être de reprendre l'algorithme de détection des événements proposés par A. Girard dans [44] et de l'implémenter sous Maple.

8.2 Problème du transfert orbital d'un satellite en temps minimal

Dans cette seconde partie, nous nous proposons d'appliquer les outils décrits tout au long de ce manuscrit au problème du transfert d'orbite d'un satellite en temps minimum.

8.2.1 Modélisation du problème de contrôle

On considère un satellite en orbite autour de la Terre que l'on souhaite amener sur une autre orbite toujours dans le champ d'attraction terrestre. Le satellite est assimilé à un

point matériel de masse initiale m_0 . Le contrôle du mouvement du satellite s'exerce par l'intermédiaire de la force de propulsion (ou poussée), notée u . De façon générale, d'après les lois fondamentales de la dynamique spatiale, le mouvement du satellite est régi par l'équation :

$$\ddot{r} = -\mu \frac{r}{\|r\|^3} + \frac{u}{m}, \quad (8.1)$$

où μ désigne la constante de gravitation (ici $\mu = gm_T$ en notant g la constante de gravitation universelle et m_T la masse de la Terre). Le contrôle u est borné :

$$\|u\| \leq F_{max} \quad (8.2)$$

et la variation de masse (due à la consommation de carburant) est décrite par l'équation :

$$\dot{m} = -\frac{\|u\|}{v_e} \quad \text{avec : } v_e \in \mathbb{R}^+. \quad (8.3)$$

Les valeurs numériques des constantes physiques sont données par le tableau de la figure 8.2.

Paramètre	Valeur
μ	5165,8620912 Mm ³ .h ⁻²
m_0	1500 kg
F_{max}	3 N

FIG. 8.2 – Constantes physiques pour le problème de transfert orbital

Le modèle du satellite ainsi formulé est dit à trois dimensions et à masse variable. Des modèles simplifiés ont été envisagés tels que le transfert coplanaire [17] (réduisant la dimension de la variable d'état à 2) et/ou les modèles à masse constante (i.e. $m \equiv m_0$). Pour chacun de ces modèles, des études géométriques ont été menées afin d'analyser leur contrôlabilité puis la structure de leurs extrémales [17, 13].

Dans la suite de ce chapitre, nous nous plaçons dans le contexte d'un transfert coplanaire et d'un modèle à masse constante ($m \equiv m_0$). Malgré la simplicité des équations du système (8.1) exprimées en coordonnées cartésiennes, celles-ci sont généralement exprimées dans un repère *radial-orthoradial* attaché au satellite. Ainsi en coordonnées orbitales (P, e_x, e_y, L), le mouvement du satellite est décrit par le système suivant :

$$\begin{bmatrix} \dot{P} \\ \dot{e}_x \\ \dot{e}_y \\ \dot{L} \end{bmatrix} = \sqrt{\frac{\mu}{P}} \begin{bmatrix} 0 \\ 0 \\ 0 \\ W^2/P \end{bmatrix} + \sqrt{\frac{P}{\mu}} \begin{bmatrix} 0 \\ \sin L \\ -\cos L \\ 0 \end{bmatrix} u_1 + \sqrt{\frac{P}{\mu}} \begin{bmatrix} 2P/W \\ \cos L + (e_x + \cos L)/W \\ \sin L + (e_y + \sin L)/W \\ 0 \end{bmatrix} u_2 \quad (8.4)$$

en posant :

$$W = 1 + e_x \cos L + e_y \sin L,$$

Les variables (P, e) décrivent les paramètres de l'ellipse osculatrice (c'est la trajectoire que décrirait le satellite s'il n'était plus soumis qu'à la gravitation terrestre) :

Conditions initiales	Conditions finales
$P^0 = 11.625 \text{ Mm}$	$P^f = 42.165 \text{ Mm}$
$e_x^0 = .75$	$e_x^f = 0$
$e_y^0 = 0$	$e_y^f = 0$
$L^0 = \pi \text{ rad}$	$L^f \text{ libre}$

FIG. 8.3 – Conditions initiales et finales pour le problème de transfert orbital

- P désigne le paramètre de l'ellipse,
- $e = (e_x, e_y)$ est le vecteur d'excentricité.

La variable L est la longitude vraie i.e. l'angle qui renseigne sur la position du satellite sur l'orbite. Les formules de changement de repère sont :

$$\begin{cases} r_1 = \frac{P}{W} \cos L \\ r_2 = \frac{P}{W} \sin L \end{cases} \quad \begin{cases} v_1 = -\sqrt{\frac{\mu}{P}}(e_y + \sin L) \\ v_2 = \sqrt{\frac{\mu}{P}}(e_x + \cos L) \end{cases} \quad (8.5)$$

où $v = \dot{r}$ désigne la vitesse du satellite.

On suppose à partir de maintenant que les contraintes sur le contrôle (8.2) s'écrivent par rapport à la norme 1 c'est-à-dire :

$$|u_1| + |u_2| < F_{max}.$$

et on note \mathbb{U}_m le domaine de contrôle défini par ces relations :

$$\mathbb{U}_m = \text{Conv}([-F_{max}, 0], [0, -F_{max}], [F_{max}, 0], [0, F_{max}]).$$

Le problème de contrôle lié au satellite est de transférer en temps minimum le satellite d'une orbite initiale vers une orbite finale toutes deux fixées, la position du satellite sur l'orbite finale étant libre. Les conditions initiales et finales du transfert orbital sont récapitulées sur le tableau de la figure 8.3. Dans la suite, nous nous intéressons dans un premier temps au problème de la contrôlabilité puis dans un second temps à la recherche de solutions optimales.

8.2.2 Étude du domaine contrôlable

La longitude finale L^f n'étant pas fixée, nous nous intéressons dans ce paragraphe au calcul approché du domaine atteignable à partir d'un point initial donné grâce à l'algorithme 9 d'approximation convexe (appliqué en renversant le temps).

Soit $X_0 = [11.625, .75, 0, \pi]$ un point initial exprimé dans le repère radial-orthoradial.

A l'échelle $h = 6$ (échelle la plus grossière), l'algorithme 9 calcule 451 points en 4 minutes, pour une approximation comportant au final 104 sommets. Par projection, nous obtenons par exemple :

- l'ensemble atteignable vis-à-vis de l'excentricité $e = (e_x, e_y)$ à partir du point $e_0 = (.75, 0)$ (cf figure 8.4-(a)).
- l'ensemble atteignable vis-à-vis des paramètres (P, e_y, L) (cf figure 8.4-(b)); signalons que malgré le défaut d'affichage le domaine obtenu est convexe).

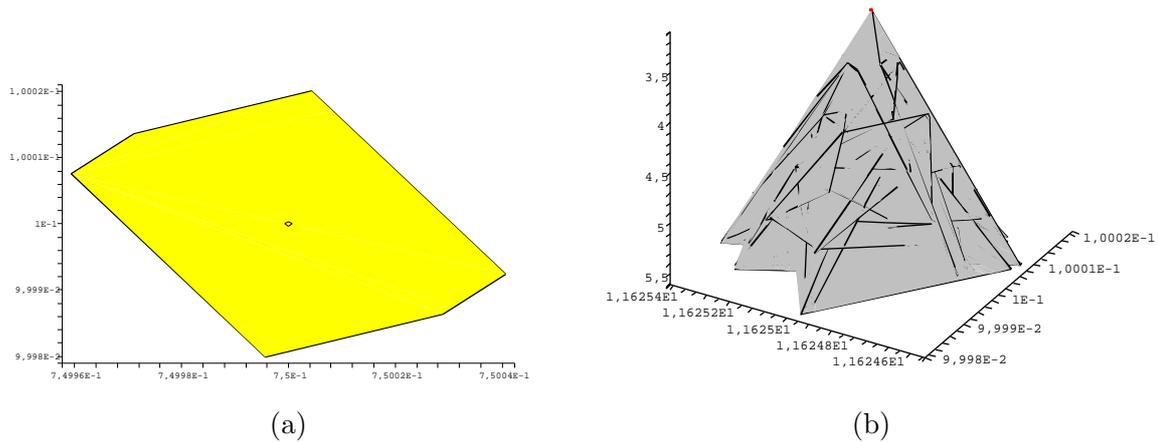


FIG. 8.4 – Ensembles atteignables vis-à-vis (a) de l'excentricité $e = (e_x, e_y)$ (b) des paramètres (P, e_y, L) , à l'échelle la plus grossière ($h = 6$).

Grâce aux formules (8.5) de changement de repère, ces résultats nous donnent une approximation du domaine atteignable à partir de X_0 dans le repère cartésien (r_1, r_2) (cf figure 8.5).

8.2.3 Recherche de trajectoires optimales

Dans le chapitre 7, nous avons étudié la forme générale des solutions optimales d'un problème de contrôle associé au modèle hybride du chapitre 3 : nous avons montré que les contrôles optimaux doivent être cherchés sous forme de boucle de rétroaction affinement dépendante par morceaux de la position du système.

De plus, nous avons vu que la difficulté réside dans l'initialisation de l'état adjoint dont dépend l'optimalité des extrémales calculées par l'algorithme 13.

Pour l'instant, nous disposons d'un algorithme de simulation des extrémales du problème de contrôle considéré. Nous retrouvons bien évident la trajectoire de référence du satellite (l'orbite initiale) comme extrémale (non optimale) par défaut du problème (cf figure 8.6).

Cependant, sans une "bonne" initialisation de l'état adjoint, nous ne pouvons calculer les extrémales candidates à l'optimalité. Ce problème s'inscrit naturellement dans les perspectives des travaux présentés dans cette thèse.

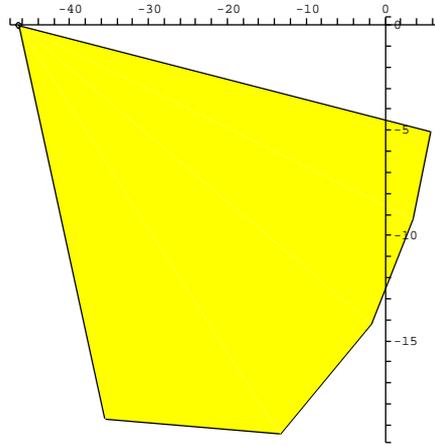


FIG. 8.5 – Approximation du domaine atteignable du satellite à l'échelle $h = 6$ dans un repère cartésien (r_1, r_2)

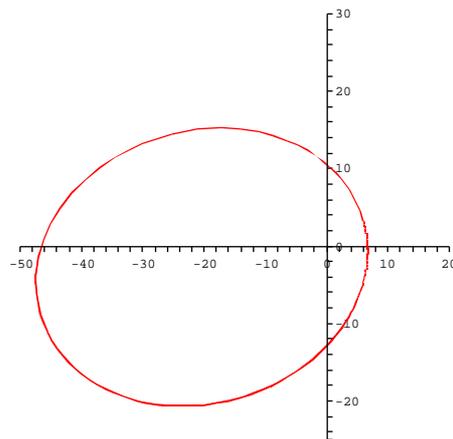


FIG. 8.6 – Extrémale de référence du problème de transfert d'un satellite

Conclusion et perspectives

Cette thèse est consacrée à l'étude des problèmes de contrôle optimal non linéaires par des méthodes de calcul hybride. Dans ce manuscrit, nous avons plus spécifiquement examiné deux problèmes centraux en théorie du contrôle : la contrôlabilité à la cible choisie et la recherche de solutions optimales. Les contributions de ces travaux se situent aussi bien au niveau théorique qu'au niveau algorithmique. Nous nous sommes en effet attachés à établir un cadre théorique solide justifiant rigoureusement notre démarche, tout en concevant et implémentant des algorithmes permettant de traiter efficacement des problèmes en grande dimension.

La partie I de ce manuscrit est consacrée à la définition d'une nouvelle classe de systèmes hybrides affines par morceaux adaptés à la modélisation des systèmes de contrôle non linéaires.

Dans le chapitre 2, nous avons proposé une généralisation des techniques de linéarisation par morceaux aux systèmes de contrôle non linéaires. Notre principale contribution dans ce chapitre réside dans l'évaluation de la qualité de l'approximation affine par morceaux. Nous présentons sur ce point une étude approfondie de l'erreur d'interpolation et de la convergence de l'approximation en fonction des hypothèses de régularité du champ modélisé. Nous proposons également un nouvel algorithme de calcul de la forme canonique de Kalman pour les systèmes linéaires permettant d'améliorer la complexité des calculs et de traiter plus efficacement des problèmes en grande dimension.

Dans le chapitre 3, nous introduisons la classe des systèmes hybrides affines par morceaux dans chaque état discret comme outil de modélisation des systèmes de contrôle non linéaires. Nous proposons dans ce chapitre des algorithmes permettant le calcul à la volée d'un maillage de l'espace état-contrôle et la manipulation efficace du modèle hybride en toute dimension.

La partie II de ce manuscrit aborde le problème de la contrôlabilité à l'origine des systèmes de contrôle non linéaires.

Dans le chapitre 4, nous nous intéressons à l'approche hybride de la contrôlabilité des systèmes de contrôle non linéaires. Pour cela, nous caractérisons la contrôlabilité en termes d'appartenance au domaine contrôlable. Dans un premier temps, nous étudions l'approximation du domaine contrôlable des systèmes non linéaires par le domaine contrôlable hybride et nous donnons une borne sur l'erreur d'approximation commise. Nous développons ensuite une approche constructive de la contrôlabilité par l'étude du domaine contrôlable défini sur des chemins d'états discrets du modèle hybride construit au chapitre 3.

Dans la continuité du chapitre 4, nous développons au chapitre 5 un algorithme de calcul d'une approximation convexe du domaine contrôlable hybride sur des chemins finis de cellules.

La partie III traite le problème de la recherche de solutions optimales des problèmes de contrôle non linéaires.

Dans le chapitre 6, nous nous intéressons à deux approches des problèmes de contrôle : la première est basée sur l'utilisation du principe du maximum hybride, la seconde sur la résolution des équations d'Hamilton-Jacobi-Bellman. Pour chacune de ces approches, nous justifions la pertinence de notre approche hybride et étudions la convergence des solutions du problème hybride vers les solutions du problème initial. Les résultats remarquables de ce chapitre sont l'obtention d'un principe du maximum hybride adapté au modèle hybride proposé ainsi que les résultats de convergence uniforme sur tout compact des problèmes hybrides vers le problème non linéaire modélisé.

Le chapitre 7 est consacré à la résolution des problèmes de contrôle optimal associés aux systèmes hybrides introduits au chapitre 3. Le résultat clé de ce chapitre est que l'exploitation des conditions nécessaires d'optimalité fournies par le principe du maximum hybride nous permet d'en déduire la structure générale des contrôles optimaux sous forme de boucle de rétroaction affine par morceaux.

Dans le huitième et dernier chapitre, nous présentons les bibliothèques Maple que nous avons développées pour la modélisation et l'étude des problèmes de contrôle non linéaires. Tous les algorithmes ont été implémentés en toute dimension et reposent sur la manipulation de polyèdres. Pour conclure ce chapitre, nous testons ces algorithmes sur un exemple concret ce qui nous permet ainsi d'établir une première comparaison de nos méthodes avec les méthodes existantes.

En conclusion, par le biais du calcul hybride, nous avons proposé dans cette thèse une méthodologie pour la modélisation et l'analyse des problèmes de contrôle non linéaires et hybrides. Les outils ainsi développés forment une base mathématique et algorithmique qui devrait servir de support à l'étude de tout problème de contrôle (avec diverses conditions aux bords) et les perspectives de ces travaux sont nombreuses.

Une première perspective de ce travail serait d'appliquer les techniques présentées à des problèmes de contrôle dont les conditions initiales et finales sont plus générales. En effet, dans ce manuscrit, nous nous sommes intéressés aux problèmes de contrôle vers un point cible donné ; nous sommes convaincus que les outils que nous avons développés dans cette thèse nous permettraient de traiter n'importe quel autre problème de contrôle avec des conditions initiales et finales plus générales ce qui nous donnerait juste des conditions de transversalité sur l'état adjoint.

Une autre perspective est l'extension des algorithmes de contrôlabilité et de calcul de solutions optimales aux systèmes hybrides polyédraux affines par morceaux. En effet, le fait que les cellules du maillage de l'espace d'état soient des simplexes n'intervient pas dans la conception des algorithmes de calcul du domaine contrôlable hybride et des solutions optimales, excepté pour le calcul des approximations linéaires locales du champ non linéaire. L'unique difficulté se situe donc au niveau de la description du système hybride considéré et plus précisément au niveau de la manipulation de polyèdres quelconques. Une solution serait d'utiliser une structure de données spécifique permettant pour un polyèdre donné de calculer ses sommets, ses faces, ses voisins, etc.

Pour la contrôlabilité approchée des systèmes non linéaires via leur approximation hybride, il nous semble qu'une approche multi-échelle devrait permettre d'améliorer les approximations

tout en évitant d'avoir à explorer tout l'espace d'état (cf chapitre 4, partie 4.3.3). Le succès de cette approche repose sur l'établissement de relations (typiquement de convexité du domaine contrôlable) entre les modèles hybrides à différentes échelles.

Concernant la recherche de solutions optimales, une première chose serait d'étendre les techniques des chapitres 6 et 7 à des problèmes où la fonction coût dépend explicitement du contrôle. Le cas où le coût est linéaire par rapport au contrôle ne pose pas de problème (l'hamiltonien reste affine par rapport au contrôle). Dans le cas où le coût est non linéaire, une piste serait de construire une approximation affine par morceaux par rapport au contrôle définissant ainsi dans chaque état discret de l'automate hybride une fonction coût locale à optimiser. Le choix des approximations est déterminant dans le sens où une approximation trop grossière risque de faire apparaître des problèmes de minima locaux et donc de fausser la résolution optimale.

Enfin il nous semble important de poursuivre la comparaison de nos méthodes hybrides avec les méthodes existantes d'une part dans le but de tester la stabilité et l'efficacité des algorithmes proposés et d'autre part pour enrichir nos méthodes de résolution en les confrontant à des applications concrètes. Une idée notamment pour résoudre le problème de l'initialisation de l'état adjoint dans une cellule donnée serait de combiner le calcul de trajectoires optimales du chapitre 7 avec une méthode directe nous permettant ainsi d'obtenir une première approximation de l'état adjoint à l'instant initial.

Connaissant la structure a priori des trajectoires optimales hybrides grâce aux résultats du chapitre 7, une dernière perspective prometteuse est de développer une méthode de tir adaptée au modèle hybride présenté au chapitre 3. La modélisation hybride développée dans cette thèse nous permettrait ainsi d'approcher n'importe quel problème de contrôle non linéaire par un problème hybride affine par morceaux pour lequel nous avons des outils de résolution efficaces en toute dimension et avec une grande précision.

Annexe A

Quelques rappels de géométrie

Cette annexe regroupe un certain nombre de rappels de géométrie sur les polytopes et les maillages. Les démonstrations et des compléments des résultats énoncés dans cette partie peuvent être trouvés dans [88]. Les notions de polytope et de maillage sont utilisées tout au long du manuscrit et plus particulièrement dans la partie I de ce manuscrit pour la construction du modèle hybride affine par morceaux.

A.1 Polyèdres et polytopes

Un polyèdre est l'intersection d'un nombre fini de demi-espaces fermés de \mathbb{R}^d du type : $\{X \in \mathbb{R}^d / AX + b \leq 0\}$ où $A \in \mathbb{M}_{k,d}(\mathbb{R})$ et $b \in \mathbb{R}^k$. Un polytope est un polyèdre borné.

Proposition A.1.1 ([88, prop 2.2.]). *Soit $P \subset \mathbb{R}^d$ un polytope.*

1. *Tout polytope est l'enveloppe convexe de ses sommets : $P = \text{Conv}(\text{vert}(P))$.*
2. *Si un polytope est l'enveloppe convexe d'un nombre fini de points, alors cet ensemble contient tous les sommets du polytope : $P = \text{Conv}(V) \Rightarrow \text{vert}(P) \subset V$.*

Définition A.1.1 (Diamètre d'un ensemble borné). *Soit $D \subset \mathbb{R}^d$ un domaine borné. Le diamètre de D , noté $\text{diam}(D)$ est défini par :*

$$\text{diam}(D) = \sup_{x,y \in D} \|x - y\|$$

Proposition A.1.2. *Soit $P \subset \mathbb{R}^d$ un polytope défini comme l'enveloppe convexe de ses sommets : $P = \text{Conv}(s_1, \dots, s_k)$. Le diamètre de P , noté $\text{diam}(P)$, est défini comme suit :*

$$\text{diam}(P) = \max_{1 \leq i \neq j \leq k} d(s_i, s_j)$$

Preuve. Soient $x, y \in P$. Par définition de P , on a :

$$\exists (\alpha_i)_{i=1..k}, (\beta_i)_{i=1..k} \in [0, 1]^k, \sum_{i=1}^k \alpha_i = \sum_{i=1}^k \beta_i = 1, \begin{cases} x = \sum_{i=1}^k \alpha_i s_i \\ y = \sum_{i=1}^k \beta_i s_i \end{cases}$$

Soit $i \in \llbracket 1, k \rrbracket$. Calculons $d(x, s_i)$:

$$\begin{aligned} d(x, s_i) &= \|x - s_i\|_2 = \left\| \sum_{j=1}^k \alpha_j (s_j - s_i) \right\|_2 \leq \sum_{j=1}^k \alpha_j \|s_j - s_i\|_2 \\ &\leq \left(\sum_{j=1}^k \alpha_j \right) \max_{i \neq j} d(s_i, s_j) = \max_{i \neq j} d(s_i, s_j) \end{aligned}$$

On en déduit alors :

$$\begin{aligned} d(x, y) &= \|x - y\|_2 = \left\| \sum_{j=1}^k \beta_j (x - s_j) \right\| \\ &\leq \sum_{j=1}^k \beta_j \|x - s_j\|_2 \leq \left(\sum_{j=1}^k \beta_j \right) \max_{i \neq j} d(s_i, s_j) = \max_{i \neq j} d(s_i, s_j) \end{aligned}$$

D'où : $\text{diam}(P) \leq \max_{i \neq j} d(s_i, s_j)$. Par ailleurs, $\{d(s_i, s_j) / i \neq j\}$ est fini donc il existe $(i, j) \in \llbracket 1, k \rrbracket^2$, tels que : $d(s_i, s_j) = \max_{k \neq k'} d(s_k, s_{k'})$. On a donc égalité. □

Définition A.1.2 (*d*-simplexe). Un *d*-simplexe de \mathbb{R}^n est l'enveloppe convexe de $d+1$ points de \mathbb{R}^n affinement indépendants.

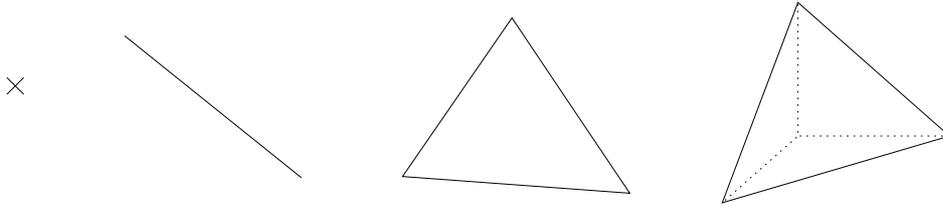


FIG. A.1 – Exemples de *d*-simplexes de \mathbb{R}^3 pour $d = 0, 1, 2, 3$.

Remarquons qu'un *d*-simplexe est également un polytope à $d + 1$ sommets.

A.2 Maillage d'un domaine en polytopes

Définition A.2.1 (Maillage). $(\Delta_i)_{i \in I}$ est un maillage d'un domaine $E \subset \mathbb{R}^d$ si et seulement si :

1. $\bigcup_{i \in I} \Delta_i = E$
2. Pour tous $i, j \in I$, $i \neq j$, l'intersection de Δ_i avec Δ_j est soit vide, soit égale à l'enveloppe convexe des sommets de Δ_i communs à Δ_j .

Le maillage $(\Delta_i)_{i \in I}$ est dit simplicial si, pour tout $i \in I$, Δ_i est un *d*-simplexe de \mathbb{R}^d .

On peut alors introduire la notion de *taille* d'un maillage :

Définition A.2.2 (Taille d'un maillage). Soit $\Delta = (\Delta_i)_{i \in I}$ un maillage d'un domaine $E \subset \mathbb{R}^d$. On définit la *taille* h du maillage Δ par :

$$h = \sup_{i \in I} h_i \text{ avec : } h_i = \text{diam}(\Delta_i)$$

Annexe B

Preuves des algorithmes pour le calcul de la forme de Kalman

Cette annexe regroupe les démonstrations des algorithmes de calcul de la forme canonique de Kalman présentés au chapitre 2, paragraphe 2.2.2. Une description plus complète des techniques et des algorithmes présentés ici est faite dans [66].

B.1 Calcul de la forme de Kalman par l’algorithme de Keller-Gehrig

B.1.1 Algorithme

Le principe de l’algorithme présenté au chapitre 2 est de calculer la matrice compressée de Krylov de la matrice B relativement à la matrice A grâce à l’algorithme de Keller-Gehrig, puis de compléter les vecteurs colonnes de cette matrice en une base de l’espace $W(A, B) = \text{Vect}(B, AB, \dots, A^{n-1}B)$. La matrice T ainsi construite est la matrice de passage permettant d’obtenir la forme canonique de Kalman donnée par le théorème 2.2.1.

On veut maintenant démontrer le théorème 2.2.2 rappelé ci-après :

Théorème. *Soient A une matrice de taille $n \times n$ et B une matrice de taille $n \times m$. L’algorithme 1 est correct et nécessite $\mathcal{O}(n^\omega \log n)$ opérations arithmétiques en utilisant la compression de Keller-Gehrig.*

B.1.2 Démonstration du théorème 2.2.2

L’algorithme de Keller-Gehrig construit la matrice compressée de Krylov \overline{K} de la matrice B relativement à A . On peut démontrer que cette matrice vérifie la relation suivante :

$$A\overline{K} = \overline{K}H,$$

où H est sous la forme polycyclique de Hessenberg [5] i.e. triangulaire supérieure par blocs avec des blocs compagnons sur la diagonale et des blocs supérieurs nuls sauf sur la dernière

Algorithme 14 Forme de Kalman par blocs**Données :** A une matrice $n \times n$, B une matrice $n \times m$, définies sur un corps**Sortie :** r, T, A_1, A_2, A_3, B_1 (cf théorème 2.2.1)

- 1: $(\overline{K}, r) = \text{CompressedKrylovMatrix}(A, B)$ {par l'algorithme de Keller-Gehrig}
- 2: **si** ($r=n$) **alors**
- 3: Renvoyer $(n, Id, A, \emptyset, \emptyset, B)$
- 4: **sinon**
- 5: $(L, [U_1 U_2], P) = \text{LUP}(\overline{K}^T)$
- 6: $T = \left[\overline{K} \mid P^T \begin{bmatrix} 0 \\ I_{n-r} \end{bmatrix} \right]$
- 7: $B_1 = L^{-T} U_1^{-T} P B$
- 8: $A' = P A^T P^T = \begin{bmatrix} A'_{11} & A'_{12} \\ A'_{21} & A'_{22} \end{bmatrix}$
- 9: $C_1 = L^{-T} U_1^{-T} A'_{12}$; $C_2 = A'_{22} - U_2^T U_1^{-T} A'_{12}$
- 10: **pour tout** j **faire**
- 11: Soit t_j les indices des colonnes dans \overline{K} de la dernière itérée l_j du $j^{\text{ème}}$ bloc.
- 12: $m_j = l_j L_{1\dots t_j, 1\dots t_j}^{-1}$
- 13: **fin pour**
- 14: Construire la matrice polycyclique H en plaçant chaque vecteur colonne m_j à l'indice de colonne t_j et en ajoutant 1 sur la sous-diagonale de toutes les autres colonnes.
- 15: Renvoyer (r, T, H, C_1, C_2, B_1)
- 16: **fin si**

colonne :

$$\overline{K}^{-1} A \overline{K} = \left[\begin{array}{ccc|ccc} \boxed{\begin{matrix} 0 & & * \\ 1 & 0 & * \\ & \ddots & \ddots & * \\ & & 1 & * \end{matrix}} & & & & * \\ & & & & * \\ & & & & * \\ & & & & * \\ & & & & * \\ & & \ddots & & \\ & & & \boxed{\begin{matrix} 0 & & * \\ 1 & 0 & * \\ & \ddots & \ddots & * \\ & & 1 & * \end{matrix}} & * \end{array} \right]. \quad (\text{B.1})$$

On introduit maintenant la matrice T construite de la façon suivante :

$$T = \left[\overline{K} \mid P^T \begin{bmatrix} 0 \\ I_{n-r} \end{bmatrix} \right],$$

où P est la matrice de permutation intervenant dans la décomposition LUP de \overline{K}^T . On veut montrer que T est la matrice de passage permettant d'obtenir la décomposition canonique de Kalman.

Calculons alors : $AT = \left[A \overline{K} \mid A P^T \begin{bmatrix} 0 \\ I_{n-r} \end{bmatrix} \right]$, et il reste à calculer C_1 et C_2 telles que :

$$AT = T \begin{bmatrix} H & C_1 \\ 0 & C_2 \end{bmatrix} \quad \text{i.e.} \quad T \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = AP^T \begin{bmatrix} 0 \\ I_{n-r} \end{bmatrix}.$$

En écrivant : $A' = PAP^T = \begin{bmatrix} A'_{11} & A'_{12} \\ A'_{21} & A'_{22} \end{bmatrix}$, on obtient :

$$T \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = P^T A' \begin{bmatrix} 0 \\ I_{n-r} \end{bmatrix} = P^T \begin{bmatrix} A'_{12} \\ A'_{22} \end{bmatrix}$$

Cependant, $T = P^T \begin{bmatrix} U_1^T & 0 \\ U_2^T & I_{n-r} \end{bmatrix} \begin{bmatrix} L^T & 0 \\ 0 & I_{n-r} \end{bmatrix}$, si bien que :

$$\begin{aligned} \begin{bmatrix} U_1^T & 0 \\ U_2^T & I_{n-r} \end{bmatrix} \begin{bmatrix} L^T & 0 \\ 0 & I_{n-r} \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} &= \begin{bmatrix} A'_{12} \\ A'_{22} \end{bmatrix} \\ \begin{bmatrix} U_1^T & 0 \\ U_2^T & I_{n-r} \end{bmatrix} \begin{bmatrix} L^T C_1 \\ C_2 \end{bmatrix} &= \begin{bmatrix} A'_{12} \\ A'_{22} \end{bmatrix} \end{aligned}$$

Nous pouvons alors en déduire le système suivant :

$$\begin{cases} U_1^T L^T C_1 &= A'_{12} \\ U_2^T L^T C_1 + C_2 &= A'_{22} \end{cases}$$

dont les solutions sont :

$$\begin{cases} C_1 &= L^{-T} U_1^{-T} A'_{12} \\ C_2 &= A'_{22} - U_2^T U_1^{-T} A'_{12} \end{cases}$$

Par conséquent, la matrice A s'écrit bien sous la forme de Kalman cherchée :

$$T^{-1}AT = \begin{bmatrix} H & C_1 \\ 0 & C_2 \end{bmatrix},$$

De plus, par construction, les colonnes de la matrice B doivent être des combinaisons linéaires des colonnes de \overline{K} , d'où le système :

$$B = T \begin{bmatrix} B_1 \\ 0 \end{bmatrix}.$$

dont la solution est : $B_1 = L^{-T} U_1^{-T} P B$.

La complexité de la compression est en $\mathcal{O}(n^\omega \log(n))$: les étapes de l'algorithme 1 comportent en effet $n \times n$ factorisations LUP, multiplications et résolutions de systèmes triangulaires, ce qui nécessite au plus $\mathcal{O}(n^\omega)$ opérations.

B.2 Calcul de la forme de Kalman par LU-Krylov

B.2.1 Algorithme et principe de démonstration

L'algorithme 2 rappelé ci-après, permet le calcul de la matrice de passage du théorème de structure canonique de Kalman.

Le principe de l'algorithme est le suivant : à l'étape i de l'algorithme, on construit la matrice compressée de Krylov K_i du premier vecteur colonne b_i de la matrice courante relativement à la matrice A_i obtenue après i itérations. On construit alors récursivement une matrice \tilde{K} que l'on complète en une matrice inversible de taille $n \times n$ et qui nous donne ainsi la matrice de passage du théorème de Kalman.

Algorithme 15 Matrice de passage de Kalman par LU-Krylov

Données : A une matrice $n \times n$, B une matrice $n \times m$ définies sur un corps

Sortie : $(\tilde{K}, \text{rang}(\tilde{K}))$

- 1: Sélectionner la première colonne b de B .
 - 2: $\left\{ \begin{array}{l} K_1 = [b \quad Ab \quad A^2b \quad \dots] \\ (L, [U_1|U_2], P) = \text{LUP}(K_1^T), r_1 = \text{rang}(K_1) \end{array} \right.$
 {La matrice K_1 est calculée à la volée : on calcule ainsi au plus $2r_1$ colonnes. La matrice U_1 est de taille $r_1 \times r_1$.}
 - 3: **si** $(r_1 = n)$ **alors**
 - 4: Renvoyer (K_1, r_1)
 - 5: **sinon**
 - 6: $A' = PAP^T = \begin{bmatrix} A'_{11} & A'_{12} \\ A'_{21} & A'_{22} \end{bmatrix}$ où A'_{11} est $r_1 \times r_1$.
 - 7: $A_R = A'_{22} - U_2^T U_1^{-T} A'_{12}$
 {La matrice A_R est appelée *complément de Schur* de A'_{11} dans A .}
 - 8: $B' = \begin{bmatrix} L^{-T} U_1^{-T} & 0 \\ -U_2^T U_1^{-T} & I \end{bmatrix} PB$
 - 9: Calculer la matrice de permutation Q telle que $B'Q = \begin{bmatrix} X & Y \\ 0 & B_R \end{bmatrix}$
 - 10: Appel récursif de $(K_2, r_2) = \text{LUCKM}(A_R, B_R)$
 - 11: $\tilde{K} = \left[K_1 \left| P^T \begin{bmatrix} 0 \\ K_2 \end{bmatrix} \right. \right]$
 - 12: Renvoyer $(\tilde{K}, r_1 + r_2)$
 - 13: **fin si**
-

En pratique, plutôt que d'appliquer l'algorithme 2 puis l'algorithme de calcul de la forme de Kalman, on peut s'apercevoir qu'à chaque étape de calcul de \tilde{K} , la matrice courante K_i permet de définir une matrice $T_i = \left[K_i \left| P^T \begin{bmatrix} 0 \\ I_{n-r} \end{bmatrix} \right. \right]$, telle que $T_i^{-1} A_i T_i$ et $T_i^{-1} B_i$ soient bien triangulaires supérieures par blocs. À la dernière itération, on obtient la décomposition canonique de Kalman.

B.2.2 Démonstration

L'étape 2 de l'algorithme 2 construit la matrice :

$$K_1 = [b \mid Ab \mid \dots \mid A^{r_1-1}b],$$

où r_1 est le rang du sous-espace $Vect(b, Ab, \dots, A^{r_1-1}b)$. On effectue la décomposition LUP de la matrice K_1^T ainsi formée : il existe donc une matrice inversible L de taille $r_1 \times r_1$, triangulaire inférieure par blocs, une matrice $U = [U_1 \mid U_2]$ de taille $r_1 \times n$ et une permutation P telles que :

$$K_1^T = L[U_1 \mid U_2]P,$$

où U_1 est une matrice inversible, triangulaire supérieure de taille $r_1 \times r_1$.

Un résultat important pour la suite de la démonstration est le suivant :

$$K_1^T A^T = C_1^T K_1^T, \quad (\text{B.2})$$

où C_1 désigne le bloc compagnon de taille $r_1 \times r_1$, défini de la façon suivante :

$$C_1 = \begin{bmatrix} 0 & & & m_0 \\ 1 & 0 & & m_1 \\ & \ddots & \ddots & \\ & & 1 & m_{r_1-1} \end{bmatrix}.$$

En effet : $AK_1 = A[b \mid Ab \mid \dots \mid A^{r_1-1}b] = [Ab \mid A^2b \mid \dots \mid A^{r_1}b]$. Or, par construction, le vecteur $A^{r_1}b$ est linéairement dépendant des premières itérées $A^i b$; il existe donc des réels $(m_i)_{i=0 \dots r_1-1}$, tels que : $A^{r_1}b = \sum_{i=0}^{r_1-1} m_i A^i b$. On obtient alors :

$$AK_1 = \sum_{i=0}^{r_1-1} m_i [Ab \mid A^2b \mid \dots \mid A^i b] = [b \mid Ab \mid \dots \mid A^{r_1-1}b] C_1 = K_1 C_1.$$

Par construction, les vecteurs colonnes de K_1 forment une base du sous-espace A -invariant engendré par b , i.e. de $Vect(b, Ab, \dots, A^{r_1-1}b)$. L'idée est maintenant de compléter cette base pour obtenir une base de l'espace complet. Pour cela, on définit la matrice inversible suivante :

$$T_1^T = \left[\begin{array}{c|c} L & 0 \\ \hline 0 & I_{n-r_1} \end{array} \right] \left[\begin{array}{c|c} U_1 & U_2 \\ \hline 0 & I_{n-r_1} \end{array} \right] P = \left[\begin{array}{c|c} K_1^T & \\ \hline [0 \mid I_{n-r_1}] P \end{array} \right]$$

Calculons maintenant le produit $T_1^T A^T T_1^{-T}$:

$$\begin{aligned} T_1^T A^T T_1^{-T} &= \left[\begin{array}{c|c} K_1^T & \\ \hline [0 \mid I_{n-r_1}] P \end{array} \right] A^T T_1^{-T} = \left[\begin{array}{c|c} K_1^T A^T & \\ \hline [0 \mid I_{n-r_1}] P A^T \end{array} \right] T_1^{-T} \\ &= \left[\begin{array}{c|c} C_1^T K_1^T & \\ \hline [0 \mid I_{n-r_1}] P A^T \end{array} \right] T_1^{-T} \end{aligned}$$

De plus, on a :

$$K_1^T T_1^{-T} = L[U_1 \mid U_2] P P^{-1} \left[\begin{array}{c|c} U_1^{-1} & -U_1^{-1} U_2 \\ \hline 0 & I_{n-r_1} \end{array} \right] \left[\begin{array}{c|c} L^{-1} & 0 \\ \hline 0 & I_{n-r_1} \end{array} \right] = [I_{r_1} \mid 0], \quad (\text{B.3})$$

ce qui nous permet d'écrire :

$$T_1^T A^T T_1^{-T} = \left[\begin{array}{c|c} C_1^T & 0 \\ \hline [0 \mid I_{n-r_1}] P A^T T_1^{-T} \end{array} \right]$$

En écrivant : $A' = P A P^{-1} = \left[\begin{array}{c|c} A'_{1,1} & A'_{1,2} \\ \hline A'_{2,1} & A'_{2,2} \end{array} \right]$, on obtient :

$$\begin{aligned} [0 \mid I_{n-r_1}] P A^T T_1^{-T} &= [A'_{1,2}{}^T \mid A'_{2,2}{}^T] \left[\begin{array}{c|c} U_1^{-1} & -U_1^{-1} U_2 \\ \hline 0 & I_{n-r_1} \end{array} \right] \left[\begin{array}{c|c} L^{-1} & 0 \\ \hline 0 & I_{n-r_1} \end{array} \right] \\ &= [A'_{1,2}{}^T U_1^{-1} L^{-1} \mid A'_{2,2}{}^T - A'_{1,2}{}^T U_1^{-1} U_2] \end{aligned}$$

On introduit alors le *complément de Schur*, noté A_R , de $A'_{1,1}$ dans A' : $A_R = A'_{2,2} - U_2^T U_1^{-T} A'_{1,2}$. Nous avons donc démontré que la matrice A est réduite à une matrice bloc triangulaire supérieure :

$$T_1^{-1} A T_1 = \left[\begin{array}{c|c} C_1 & L^{-T} U_1^{-T} A'_{1,2} \\ \hline 0 & A_R \end{array} \right]$$

Or la matrice \tilde{K} cherchée est formée des itérées $A^i b_j$ (linéairement indépendantes) des colonnes de B . De plus, appliquer A aux vecteurs colonnes de B , revient à appliquer $A T_1$ à ceux de $T_1^{-1} B$. Calculons donc la matrice $B' = T_1^{-1} B$:

$$B' = T_1^{-1} B = \left[\begin{array}{c|c} L^{-T} & 0 \\ \hline 0 & I_{n-r_1} \end{array} \right] \left[\begin{array}{c|c} U_1^{-T} & 0 \\ \hline -U_2^T U_1^{-T} & I_{n-r_1} \end{array} \right] P B = \left[\begin{array}{c|c} L^{-T} U_1^{-T} & 0 \\ \hline -U_2^T U_1^{-T} & I_{n-r_1} \end{array} \right] P B$$

Tout vecteur colonne v de B linéairement dépendant des itérées $b, Ab, \dots, A^{r_1-1}b$, s'écrit comme combinaison linéaire de ces vecteurs, i.e. des vecteurs colonnes de K_1 . Or, d'après (B.3), nous savons que : $T_1^{-1} K_1 = \left[\begin{array}{c} I_{r_1} \\ 0 \end{array} \right]$. Par conséquent, l'image par T_1^{-1} du vecteur v est

de la forme : $T_1^{-1} v = \left[\begin{array}{c} V_1 \\ 0 \end{array} \right]$, où le bloc de 0 est un vecteur colonne de dimension $n - r_1$. Il existe donc une permutation Q sur les colonnes de B qui nous permet d'écrire B sous la forme $[B_1 \mid B_2]$, où B_1 est la matrice formée des vecteurs colonnes de B , linéairement dépendants des $A^i b$, $i = 0, \dots, r_1 - 1$. La matrice $T_1^{-1} B Q$ est donc de la forme :

$$B' Q = T_1^{-1} B Q = \left[\begin{array}{c|c} X & Y \\ \hline 0 & B_R \end{array} \right], \text{ où } B_R \text{ est une matrice à } (n - r_1) \text{ lignes.}$$

De façon récursive, il nous suffit alors de ré-appliquer cet algorithme au couple (A_R, B_R) . On fait ainsi apparaître à chaque étape un bloc de zéro de plus dans les matrices A et B , jusqu'à obtenir des matrices triangulaires supérieures par blocs de la dimension cherchée. Les matrices K_i successives définissent alors la matrice \tilde{K} comme suit :

$$\tilde{K} = \left[\begin{array}{c|c} K_1 & P^T \left[\begin{array}{c} 0 \\ K_2 \end{array} \right] \end{array} \right]$$

Bibliographie

- [1] A.A. Agrachev and Y.L. Sachkov. *Control Theory from the Geometric Viewpoint*, volume 87 of *Encyclopaedia of Mathematical Sciences*. Springer, 2004.
- [2] R. Alur, T. Dang, and F. Ivancic. Reachability analysis of hybrid systems via predicate abstraction. In *Fifth International Workshop on Hybrid Systems : Computation and Control*, 2002.
- [3] E. Asarin and T. Dang. Abstraction by projection and application to multi-affine systems. In R. Alur and G.J. Pappas, editors, *Hybrid Systems : Computation and Control*, volume 2993 of *LNCS*, pages 32–47. Springer, 2004.
- [4] E. Asarin, T. Dang, and A. Girard. Reachability of non-linear systems using conservative approximations. In O. Maler and A. Pnueli, editors, *Hybrid Systems : Computation and Control*, volume 2623 of *LNCS*, pages 22–35. Springer, 2003.
- [5] D. Augot and P. Camion. On the computation of minimal polynomials, cyclic vectors, and Frobenius forms. *Linear Algebra and its Applications*, 260(1–3) :61–94, July 1997.
- [6] C.B. Barber, D.P. Dobkin, and H.T. Huhdanpaa. The Quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 1996.
- [7] M. Bardi and I. Capuzzo-Dolcetta. *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, volume 17 of *Systems & Control : Foundations & Applications*. Birkhauser, 1997.
- [8] G. Barles. *Solutions de viscosité des équations de Hamilton-Jacobi*, volume 17 of *Mathématiques et Applications*. Springer-Verlag, 1994.
- [9] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.
- [10] D.P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 1 & 2. Athena Scientific, 1984.
- [11] J.F. Bonnans and S. Maurin. An implementation of the shooting algorithm for solving optimal control problems. *Technical Report RT-0240, INRIA*, 2000.
- [12] B. Bonnard and J.-B. Caillau. *Advanced topics in control systems theory*, volume 328 of *Lecture Notes in Control and Information Sciences*, chapter 1 : Introduction to nonlinear optimal control, pages 1–60. Springer-Verlag, A. Loria, F. Lamnabhi and E. Panteley edition, 2006.
- [13] B. Bonnard, J.-B. Caillau, and E. Trélat. Geometric optimal control of elliptic keplerian orbits. *Discrete and Continuous Dynamical Systems - Series B*, 4(5) :929–956, 2005.
- [14] B. Bonnard and M. Chyba. The role of singular trajectories in control theory. *Mathematical Monographs, Springer-Verlag*, 2003.

-
- [15] M.S. Branicky, V.S. Borkar, and S.K. Mitter. A Unified Framework for Hybrid Control : Model and Optimal Control Theory. *IEEE Transactions on Automatic Control*, 43(1) :31–45, 1998.
- [16] A.E. Bryson and Y. Ho. *Applied Optimal Control*. Hemisphere, 1975.
- [17] J.-B. Caillaud and J. Noailles. Coplanar control of a satellite around the earth. *ESAIM : Control, Optimisation and Calculus of Variations*, 6 :239–258, 2001.
- [18] I. Capuzzo-Dolcetta. *On a Discrete Approximation of the Hamilton-Jacobi Equation of Dynamic Programming*, volume 10 of *Applied Mathematics and Optimization*, pages 367–377. Springer-Verlag, 1983.
- [19] H. Cartan. *Calcul différentiel*. Collection Méthodes. Hermann Paris, 1967.
- [20] M.-J. Chien and E.S. Kuh. Solving nonlinear resistive networks using piecewise-linear analysis and simplicial subdivision. *IEEE Transactions on Circuits and Systems*, 24(6) :305–317, 1977.
- [21] F.H. Clarke. *Optimization and Nonsmooth Analysis*. SIAM Classics in Applied Mathematics, 1990.
- [22] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*, 9 :251–280, 1990.
- [23] M.G. Crandall, L.C. Evans, and P.-L. Lions. Some properties of viscosity solutions of hamilton-jacobi equations. *Transactions of the American Mathematical Society*, 282(2) :487–502, 1984.
- [24] M.G. Crandall, H. Ishii, and P.-L. Lions. Uniqueness of viscosity solutions revisited. *Journal of Mathematical Society*, 39 :581–596, 1987.
- [25] M.G. Crandall, H. Ishii, and P.-L. Lions. A user’s guide to viscosity solutions. *Bulletin A.M.S., N.S.*, 27 :1–67, 1992.
- [26] M.G. Crandall and P.-L. Lions. Viscosity solutions of hamilton-jacobi equations. *Transactions of the American Mathematical Society*, 277(1) :1–42, 1983.
- [27] T. Dang. *Vérification et synthèse des systèmes hybrides*. PhD thesis, Institut National Polytechnique de Grenoble, 2000.
- [28] D.F. Delchamps. *State Space and Input-Output Linear Systems*. Springer-Verlag, 1988.
- [29] J. Della Dora, A. Maignan, M. Mirica-Ruse, and S. Yovine. Hybrid computation. In *Proceedings of the 2001 International Symposium on Symbolic and Algebraic Computation*. Bernard Mourrain editor, ACM Press, July 2001.
- [30] J. Dieudonné. *Calcul infinitésimal*. Collection Méthodes. Hermann Paris, 1968.
- [31] J.-G. Dumas, T. Gautier, and C. Pernet. Finite field linear algebra subroutines. In Teo Mora, editor, *Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*, pages 63–74. ACM Press, New York, 2002.
- [32] J.-G. Dumas, P. Giorgi, and C. Pernet. FFPACK : Finite Field Linear Algebra Package. In Jaime Gutierrez, editor, *Proceedings of the 2004 International Symposium on Symbolic and Algebraic Computation*, pages 140–147. ACM Press, New York, 2004.
- [33] J.-G. Dumas, C. Pernet, and Z. Wan. Efficient computation of the characteristic polynomial. In Manuel Kauers, editor, *Proceedings of the 2005 International Symposium on Symbolic and Algebraic Computation*, pages 277–284. ACM Press, New York, 2005.

- [34] J.-G. Dumas and A. Rondepierre. Modeling the electrical activity of a neuron by a continuous and piecewise affine hybrid system. In *Hybrid Systems : Computation and Control*, volume 2623 of *LNCS*, pages 156–171. Springer, 2003.
- [35] J.-G. Dumas and A. Rondepierre. Algorithms for symbolic/numeric control of affine dynamical systems. In Manuel Kauers, editor, *Proceedings of the 2005 International Symposium on Symbolic and Algebraic Computation*, pages 277–284. ACM Press, New York, 2005.
- [36] J.-G. Dumas and A. Rondepierre. Hybrid optimal control. *International Journal of Control*, soumis, 2005.
- [37] G. Ewald. *Combinatorial Convexity and Algebraic Geometry*, volume 168 of *Graduate Texts in Mathematics*. Springer-Verlag, 1996.
- [38] R. Fierro, A. K. Das, V. Kumar, and J. P. Ostrowski. Hybrid control of formations of robots. In *IEEE Proceedings of the International Conference on Robotics and Automation*, 2001.
- [39] A.F. Filippov. *Differential equations with discontinuous righthand sides*, volume 18 of *Mathematics and their Applications (Soviet series)*. Kluwer Academic, 1988.
- [40] H. Freudenthal. Simplizialzerlegungen von beschraeukter flachheit. *Annals of Math. in Science and Engin.*, 43 :580–582, 1942.
- [41] A.T. Fuller. Relay control systems optimized for various performance criteria. In *Proceedings of the First World Congress IFAC*, pages 510–519. Butterworth, 1960.
- [42] R.V. Gamkrelidze. Discovery of the maximum principle. *Journal of Dynamical and Control Systems*, 5(4) :437–451, 1999.
- [43] A. Girard. Approximate solutions of odes using piecewise linear vector fields. In *Proceedings of the 2002 Computer Algebra in Scientific Computing*, pages 107–120. Springer Verlag, September 2002.
- [44] A. Girard. *Analyse algorithmique des systèmes hybrides*. PhD thesis, Institut National Polytechnique de Grenoble, 2004.
- [45] G.H. Golub and C.F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. The Johns Hopkins University Press, Baltimore, MD, USA, third edition, 1996.
- [46] T. Guilbaud. *Méthodes numériques en commande optimale*. PhD thesis, Université Paris VI, 2002.
- [47] L.C.G.J.M. Habets and J.H. van Schuppen. Control of piecewise-linear hybrid systems on simplices and rectangles. In M.D. Di Benedetto and A.L. Sangiovanni-Vincentelli, editors, *Hybrid Systems : Computation and control*, volume 2034 of *LNCS*, pages 261–274. Springer, 2001.
- [48] R. Hartl, S. Sethi, and R. Vickson. A survey of the maximum principles for optimal control problems with state constraints. *SIAM Review*, 37(2) :181–218, 1995.
- [49] S. Hedlund and A. Rantzer. Optimal control of hybrid systems. In *Proceedings 38th IEEE Conference on Decision and Control*, 1999.
- [50] O.H. Ibarra, S. Moran, and R. Hui. A generalization of the fast lup matrix decomposition algorithm and applications. *Journal of Algorithms*, 3 :45–56, 1982.

- [51] A. Isidori. *Nonlinear Control Systems*. Springer, 3rd edition, 2001.
- [52] M. Johansson. *Piecewise Linear Control Systems*, volume 284 of *Lectures Notes in Control and Information Sciences*. Springer, 2003.
- [53] C.D. Johnson and J.E. Gibson. Singular solutions in problems of optimal control. *IEEE Transactions on Automatic Control*, 8 :4–15, 1963.
- [54] P. Julian, A. Desages, and O. Agamennoni. High-level canonical piecewise linear representation using a simplicial partition. *IEEE Transactions on Circuits and Systems - I : Fundamental Theory and Applications*, 46(4) :463–480, 1999.
- [55] V. Jurdjevic. *Geometric Control Theory*. Cambridge University Press, 1997.
- [56] R.E. Kalman. Canonical structure of linear dynamical systems. In *Proceedings of the National Academy of Sciences*, pages 596–600, 1961.
- [57] R.E. Kalman. Mathematical description of linear dynamical systems. *SIAM Journal on Control*, 1 :152–192, 1963.
- [58] R.E. Kalman, P.A. Falb, and M. Arbib. *Topics in Mathematical System Theory*. New York : McGraw-Hill, 1969.
- [59] W. Keller-Gehrig. Fast algorithms for the characteristic polynomial. *Theoretical computer science*, 36 :309–317, 1985.
- [60] H.J. Kelley, R. Kopp, and H. G. Moyer. *Singular Extremals*, pages 63–101. Academic Press, 1967.
- [61] M.M. Lee and L. Markus. *Foundations of Optimal Control Theory*. Wiley, New York, 1967.
- [62] C. Lobry. Contrôlabilité des systèmes non linéaires. *SIAM Journal of Control*, 8 :573–605, 1970.
- [63] A.S. Matveev and A.V. Savkin. *Qualitative Theory of Hybrid Dynamical Systems*. Control Engineering. Birkhäuser, 2000.
- [64] G.S. Minimis and C.C. Paige. A direct algorithm for pole assignment of time-invariant multi-input linear systems using state feedback. *Automatica*, 24(3) :343–356, 1988.
- [65] M. Mirica-Ruse. *Contribution à l'étude des systèmes hybrides*. PhD thesis, Université Joseph Fourier, Grenoble, 2002.
- [66] C. Pernet, A. Rondepierre, and G. Villard. Computing the Kalman form. Technical report, IMAG-ccsd-00009558, arXiv cs.SC/0510014, oct 2005.
- [67] H.J. Pesch. A practical guide to the solutions of real-life optimal control problems. *Parametric Optimization. Control Cybernet*, 23 :7–60, 1994.
- [68] E.R. Pinch. *Optimal Control and the Calculus of Variations*. Oxford University Press, 1993.
- [69] E. Plahte1 and S.W. Omholt T. Mestl. Global analysis of steady points for systems of differential equations with sigmoid interactions. *Dynamics and Stability of Systems*, 9(4) :275–291, 1994.
- [70] L. Pontriaguine, V. Boltiansky, R. Gamkrelidze, and E. Michtchenko. *The mathematical theory of optimal processes*. Editions de Moscou, 1962.
- [71] P. Riedinger, C. Iung, and F. Kratz. An optimal control approach for hybrid systems. *European Journal of Control*, 9 :449–458, 2003.

- [72] H.M. Robbins. A generalized Legendre-Clebsch condition for the singular cases of optimal control. *IBM Journal of Research and Development*, 11(4) :361–372, 1967.
- [73] A. Rondepierre. Piecewise affine systems controllability and hybrid optimal control. In *Proceedings of the 2005 International Conference on Informatics in Control, Automation and Robotics*, 2005.
- [74] E. Sontag. *Mathematical Control Theory : Deterministic Finite Dimensional Systems*. Graduate Texts in Mathematics. Springer, second edition, 1998.
- [75] E.D. Sontag. Nonlinear regulation : The piecewise linear approach. *IEEE Transactions on Automatic Control*, 26(2) :346–358, 1981.
- [76] M. Storace and O. De Feo. Piecewise-linear approximation of nonlinear dynamical systems. *IEEE Transactions on Circuits and Systems - I :Fundamental Theory and Applications*, 51(4) :830–842, 2004.
- [77] H.J. Sussmann. A maximum principle for hybrid optimal control problems. In *IEEE Conference on Decision and Control*, pages 425–430, 1999.
- [78] H.J. Sussmann. Piecewise-linear approximation of nonlinear dynamical systems. *IEEE Transactions on Circuits and Systems - I :Fundamental Theory and Applications*, 51(4) :830–842, 2004.
- [79] P. Tabuada and G.J. Pappas. Hybrid abstractions that preserve timed languages. In M.D. Di Benedetto and A.L. Sangiovanni-Vincentelli, editors, *Hybrid Systems : Computation and Control*, volume 2034 of *LNCS*, pages 501–514. Springer, 2001.
- [80] L. Tournier. *Étude et modélisation mathématique de réseaux de régulation génétique et métabolique*. PhD thesis, Institut National Polytechnique de Grenoble, 2005.
- [81] E. Trélat. *Contrôle optimal : théorie et applications*. Collection Mathématiques Concrètes. Vuibert, 2005.
- [82] J.H. van Schuppen. A sufficient condition for controllability of a class of hybrid systems. In T.A. Henzinger and S. Sastry, editors, *Hybrid Systems : Computation and control*, volume 1386 of *LNCS*, pages 374–383. Springer, 1998.
- [83] G. Villard. Communication personnelle.
- [84] X. Xu and P.J. Antsaklis. Optimal control of switched systems via non-linear optimization based on direct differentiations of value functions. *International Journal of Control*, 75(16–17) :1406–1426, 2002.
- [85] J. Zaytoon. *Systèmes dynamiques hybrides*. Systèmes automatisés. Hermès Science, 2001.
- [86] M.I. Zelikin and V.F. Borisov. Optimal chattering feedback control. *Journal of Mathematical Sciences*, 114(3) :1227–1344, 2003.
- [87] J. Zhang, K.H. Johansson, J. Lygeros, and S. Sastry. Zeno hybrid systems. *International Journal of Robust and Nonlinear Control*, 11 :435–451, 2001.
- [88] G.M. Ziegler. *Lectures on Polytopes*. Graduate Texts in Mathematics. Springer-Verlag, 1994.

Résumé : Cette thèse est consacrée à la résolution des problèmes de contrôle non linéaires par des méthodes de calcul hybride. L'idée défendue est que la modélisation par les systèmes hybrides permet la résolution approchée des problèmes non linéaires sans connaissance a priori du comportement du système étudié. Dans un premier temps, nous nous intéressons à la modélisation des systèmes de contrôle non linéaires par une nouvelle classe de systèmes hybrides affines par morceaux. Un soin particulier est apporté à l'étude de l'erreur et de la convergence de l'approximation hybride. La deuxième partie est consacrée au problème de la contrôlabilité à l'origine des systèmes non linéaires. Après avoir quantifié l'erreur commise entre le domaine contrôlable non linéaire et son approximation hybride, nous proposons une approche constructive pour le calcul du domaine contrôlable, permettant de réduire l'exploration des états discrets de l'automate hybride. La dernière partie est dédiée à la recherche de solutions optimales des problèmes de contrôle non linéaires et hybrides. Nous justifions tout d'abord la pertinence de la modélisation hybride à travers deux approches : le principe du maximum de Pontryagin et les solutions de viscosité des équations d'Hamilton-Jacobi-Bellman. Nous énonçons en particulier un principe du maximum hybride qui nous permet alors de déterminer la structure du contrôle optimal hybride. Ces trois parties répondent à un objectif principal : développer par le calcul hybride combinant analyse numérique et calcul formel, des outils mathématiques et algorithmiques efficaces pour l'étude de dynamiques contrôlées non linéaires.

Mots-clés : contrôle optimal, systèmes hybrides affines par morceaux, contrôlabilité et domaine contrôlable, principe du maximum hybride, équations d'Hamilton-Jacobi-Bellman.

Abstract : This thesis is devoted to the analysis of non linear control problems by hybrid computation methods. We defend the idea that the use of hybrid systems allows the approximate resolution of non linear control problems without any a priori knowledge of the behavior of the considered systems. In the first part we focus on the modelling of non linear control systems by a new class of piecewise affine hybrid systems. Particularly, we propose a full analysis of the error and of the convergence of the hybrid approximation. The second part deals with the controllability to the origin of non linear systems. After providing an analysis of the approximation error between the non linear controllable domain and its hybrid approximation, we then propose a constructive approach for the computation of the controllable domain which allows to reduce the exploration of the discrete states of the hybrid automaton. The last part is devoted to the search of optimal solutions of the non linear and hybrid optimal control problems. We first justify the relevance of our hybrid model thanks to two approaches : the Pontryagin maximum principle and viscosity solutions of Hamilton-Jacobi-Bellman equations. Particularly, we state a hybrid maximum principle which provides us the structure of the hybrid optimal control. These three parts answer to the main purpose : to develop by the hybrid computation combining numerical analysis and computer algebra, some mathematical and algorithmic efficient tools for the analysis of non linear control systems.

Keywords : optimal control, piecewise affine hybrid systems, controllability and controllable domain, hybrid maximum principle, Hamilton-Jacobi-Bellman equations.