



Surveillance préventive des systèmes hybrides à incertitudes bornées

Moussa Maiga

► **To cite this version:**

Moussa Maiga. Surveillance préventive des systèmes hybrides à incertitudes bornées. Autre. Université d'Orléans, 2015. Français. <NNT : 2015ORLE2010>. <tel-01288466>

HAL Id: tel-01288466

<https://tel.archives-ouvertes.fr/tel-01288466>

Submitted on 15 Mar 2016

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.

ÉCOLE DOCTORALE
MATHÉMATIQUES, INFORMATIQUE, PHYSIQUE THÉORIQUE ET INGÉNIERIE DES
SYSTÈMES

LABORATOIRES PRISME et LAAS CNRS

THÈSE présentée par :
Moussa MAÏGA

soutenue le : *02 juillet 2015 à 10 heures*
Salle Sigaud de Lafond, IUT de Bourges

pour obtenir le grade de : **Docteur de l'université d'Orléans**
Discipline : SCIENCES ET TECHNOLOGIES INDUSTRIELLES

**Surveillance préventive des systèmes
hybrides à incertitudes bornées**

THÈSE dirigée par :

Nacim RAMDANI

Professeur des universités, Université d'Orléans, Bourges

Louise TRAVÉ-MASSUYÈS

Directeur de recherche CNRS, LAAS Toulouse

RAPPORTEURS :

Didier MAQUIN

Professeur des universités, Université de Lorraine, Nancy

Michel KIEFFER

Professeur des universités, Université Paris XI

JURY :

Didier MAQUIN

Professeur des universités, Université de Lorraine, Nancy

Michel KIEFFER

Professeur des universités, Université Paris XI

Vicenç PUIG

Professeur des universités, Univ. Politècnica de Catalunya, *Président du jury*

Christophe COMBASTEL

Maître de Conférences, ENSEA Cergy-Pontoise

Louise TRAVÉ-MASSUYÈS

Directeur de recherche CNRS, LAAS Toulouse

Nacim RAMDANI

Professeur des universités, Université d'Orléans, Bourges

TABLE DES MATIÈRES

1	Introduction	9
1.1	Contexte et motivation	9
1.2	Structure de la thèse	12
2	Notions formelles et outils mathématiques	13
2.1	Fondamentaux et problèmes traités	13
2.1.1	Définition d'un système dynamique hybride	14
2.1.2	Formalisme d'un système dynamique hybride	14
2.1.3	Atteignabilité hybride	17
2.1.4	Observation ensembliste des systèmes hybrides	19
2.1.5	Estimation ensembliste de paramètres des systèmes hybrides	20
2.2	Outils mathématiques pour le calcul ensembliste	20
2.2.1	Arithmétique des intervalles	21
2.2.2	Résolution des Problèmes de satisfaction de contraintes par arithmétique des intervalles	23
2.2.3	Notion de contracteur : Hc4-revise	23
2.3	Inversion ensembliste	25
2.3.1	Opérations sur les zonotopes	27
2.4	Intégration numérique garantie d'équations différentielles : tube de trajectoires	30
2.4.1	Méthode de Lohner : factorisation QR	33
2.4.2	Représentation zonotopique et domaine de solution	36
3	Atteignabilité hybride. État de l'art	39
3.1	Introduction	39
3.2	Atteignabilité continue	40
3.3	Transition discrète	42
3.4	Cas où la garde est linéaire	43
3.4.1	Méthodes par approximation d'ensembles pour l'atteignabilité hybride linéaire	43
3.5	Cas où la garde est non linéaire	47
3.5.1	Méthode par approximation polynomiale de la garde	48
3.5.2	Méthodes par intervalles	49
3.6	Conclusion	50

4	Méthodes ensemblistes rapides pour l'atteignabilité hybride	51
4.1	Introduction	52
4.2	Méthodes et algorithmes pour le problème de passage des gardes	53
4.2.1	Garde non linéaire et résolution par CSP	53
4.2.2	Garde linéaire	56
4.3	Méthode ensembliste pour le saut discret	68
4.3.1	Fonction de reset linéaire	68
4.3.2	Fonction de reset non linéaire	69
4.4	Méthodes pour la fusion des trajectoires	70
4.4.1	Calcul des sommets des sous-tubes	71
4.4.2	Enveloppe d'un nuage de points par un zonotope	73
4.4.3	Construction d'une MSBP à partir du zonotope englobant	76
4.4.4	Analyse de complexité	78
4.5	Évaluation de la méthode d'atteignabilité hybride sur des exemples simples	80
4.5.1	Exemple 1 : Balle rebondissante	80
4.5.2	Exemple 2 : Système masse-ressort à commutation	84
4.5.3	Exemple 3 : Balle rebondissante sur une surface sinusoïdale	87
4.6	Évaluation de la méthode d'atteignabilité hybride sur des benchmarks	91
4.6.1	Benchmark 1 : Manœuvre anti-collision	92
4.6.2	Benchmark 2 : Intégrateur non-holonyme	94
4.6.3	Benchmark 3 : Contrôle de la glycémie de patients diabétiques	95
4.7	Conclusion	100
5	Méthodes ensemblistes pour l'estimation de paramètres et d'état pour les systèmes hybrides dans le contexte erreur bornée	107
5.1	Motivations : détection de défauts et diagnostic	107
5.1.1	Approche prédiction/correction	109
5.1.2	Approche paramétrique	110
5.2	Observation ensembliste des SHIB	112
5.2.1	Approche prédiction-correction pour les SHIB	112
5.2.2	Exemple numérique 1 : système masse-ressort hybride	120
5.2.3	Exemple numérique 2 : système à deux réservoirs	124
5.2.4	Exemple de scénario de défaut	125
5.3	Méthode ensembliste pour l'estimation paramétrique des SHIB	126
5.3.1	Tests d'inclusion	127
5.3.2	SIVIA pour système hybride (SIVIA-H)	130
5.3.3	Exemple numérique : système masse-ressort hybride	132
5.3.4	Exemple de scénarios de défaut	134
5.4	Conclusion	136
	Conclusion et perspectives	139
	Bibliographie	163

TABLE DES FIGURES

1.1	Surveillance des systèmes dynamiques hybrides dans un contexte à erreur bornée	11
2.1	Système hybride.	15
2.2	Balle rebondissante en 1D modélisée comme un automate hybride, l'ensemble atteignable dans l'espace $x \times v$ en 2D et dans l'espace $x \times v \times t$ en 3D en un temps fini.	16
2.3	Ensemble atteignable par un système hybride en temps fini (2.1–2.2–2.3).	17
2.4	Résolution de l'intersection du tube de trajectoires et de la garde. L'objectif est de résoudre un CSP de manière à déterminer l'ensemble des conditions initiales z_j^* qui conduit le vecteur $z(t^*; t_j, z_j^*)$ à intersecter la garde à l'instant t^* . L'ensemble solution du CSP est donné par toutes les combinaisons possibles de $z_j^* \times t^*$ (Ramdani and Nedialkov, 2011).	18
2.5	Observateur de système hybride basé sur la prédiction/correction; où $\mathcal{Z}^F(t_{j+1})$ est la prédiction à un pas de l'ensemble $\mathcal{Z}(t_j)$, $\mathcal{Z}^{inv}(t_{j+1})$ est l'ensemble obtenu par inversion ensembliste à l'instant t_{j+1} à partir de \mathbb{Y}_{j+1} , $\mathcal{Z}^c(t_{j+1})$ est l'ensemble corrigé avec les mesures à l'instant t_{j+1} .	20
2.6	Exemple d'un sous-pavage.	22
2.7	Évaluation et phase de projection de la procédure de HC4-Revise à la contrainte $(x - y)^2 = z$ avec $[x] = [8, 10]$, $[y] = [0, 4]$ et $[z] = [25, 36]$.	24
2.8	Illustration de SIVIA	26
2.9	Illustration de SIVIA-C	27
2.10	Exemple de zonotope.	28
2.11	Exemple de zonotope obtenu avec la toolbox PNPMP de MATLAB [®] (Riverso et al., 2013).	29
2.12	Illustration du tube de trajectoires (boîte en bleu) entre deux pas de temps t_j, t_{j+1} ; $[\tilde{z}_j]$ représente la solution a priori, $[z_j]$ représente la solution à l'instant t_j , et $[z_{j+1}]$ représente la solution à l'instant t_{j+1} .	31
2.13	Illustration de la factorisation QR.	34
2.14	Somme de Minkowski du paralléloétope $A[r]$ et de la boîte alignée avec les axes $[v]$.	37
3.1	Encadrement de l'ensemble qui intersecte la condition de garde par un zonotope (Althoff, 2010)	44

3.2	Conversion zonotope \rightarrow polytope et intersection polytope, polytope (Althoff, 2010)	45
3.3	Exemple d'un polyèdre D_q (Girard, 2004)	46
3.4	Notion de fonction support	47
3.5	Intersection ensemble (fonction support) et condition de garde (hyperplan) (Guer- nic and Girard, 2009b).	47
4.1	Illustration de l'algorithme de calcul d'intersection	54
4.2	Intersection zonotope/hyperplan $Z_1 \cap \mathcal{H} \subseteq Z_\cap$	59
4.3	Ensemble atteignable calculé pour (4.21). Les figures (a-c-e) illustrent la méthode de résolution par intersection zonotope/hyperplan, CPU times=0.16s ; les figures (b-d-f) illustrent la méthode de résolution par CSP, CPU times=0.20s.	60
4.4	Le "support" strip d'un zonotope	61
4.5	Illustration sur l'exemple 4.21 des trois méthodes de passage ensembliste des gardes, les figures (a-d-g-j) illustrent la méthode de résolution par intersection zonotope/hyperplan, les figures (b-e-h-k) illustrent la méthode de résolution par CSP, les figures (c-f,i,l) illustrent la méthode de résolution par intersec- tion zonotope/demi-espace, illustration 3D dans l'espace $x_1 \times x_2 \times t$ pour les trois méthodes	64
4.6	Illustration de la méthode de passage ensembliste d'une garde de type polytope, Temps CPU=0.108s	67
4.7	Construction de l'ensemble des vecteurs $c_\ell + Z^\pm(R_\ell)$	73
4.8	Étapes principales pour construire le zonotope englobant réalisée par l'algorithme10 cloud2zonotope dans le cas d'un nuage de point en 2D en entrée et une MSBP en sortie , (a) centrage du nuage de points et compression le long de la première direction principale, (b) trouver la seconde direction de compression comme la première direction principale du nuage restant, (c) englober le reste du nuage dans une boîte alignée avec les axes, (d) calcul de la MSBP englobant le nuage de points.	76
4.9	Automate hybride de la balle rebondissante	81
4.10	L'ensemble atteignable obtenu pour la balle rebondissante. Historique des va- riables p_x, p_y, v_y en fonction du temps	82
4.11	(a-b) : L'ensemble atteignable obtenu pour la balle rebondissante. Projection dans l'espace $p_y \times p_x$, (a) : sans la fusion des trajectoires, et (b) : avec fusion des trajectoires, (c-d-e-f-g-h-i) : les zonotopes obtenus en 2D (projection dans dans l'espace $p_x \times p_y$ en fonction du ratio r à $t=2s$) pour la balle rebondissante . . .	83

4.12	L'ensemble atteignable du système masse-ressort. De gauche à droite, et de haut en bas, la version continue (CPU 0.028s), la version commutation sans la fusion de trajectoires (CPU 1.884s), la version commutation avec la fusion des trajectoires en utilisant le critère <i>P-radius</i> (CPU 0.196s), en utilisant le critère longueur des segments (CPU 0.164s), en utilisant le critère volume (CPU 0.2s), quand on utilise la fusion avec $ratio=0$ (CPU 0.308s), et la méthode naïve qui consiste à prendre l'enveloppe convexe des boîtes solutions. La dernière figure compare les MSBP obtenus au temps final pour les différentes expérimentations.	85
4.13	L'ensemble atteignable du système masse-ressort en 3D ($x_2 \times x_1 \times t$), à gauche sans la fusion de trajectoires et à droite avec la fusion des trajectoires pour le critère <i>P-radius</i>	86
4.14	A droite, automate hybride modélisant la balle rebondissante, et à gauche la balle rebondissante sur une surface non linéaire avec deux directions possible selon les positions initiales pour une hauteur donnée	87
4.15	Capture des sous-trajectoires droites et gauche après le premier rebond. Les points en noir représentent les sommets des MSBP (nuage de points) obtenues après le saut, l'unique MSBP est obtenu avec chacun des trois critères, ainsi que la valeur optimale pour <i>ratio</i>	88
4.16	La frontière, c'est-à-dire l'ensemble solution à chaque point de la grille temporelle dans l'espace $p_y \times p_x$ pour la balle rebondissante. De gauche à droite, l'ensemble solutions obtenu sans la fusion des trajectoires, avec la fusion des trajectoires en utilisant le critère <i>P-radius</i> , puis la longueur des segments, et le volume.	88
4.17	L'ensemble atteignable (à gauche sans la fusion, et à droite avec la fusion avec le critère de longueur des segments) obtenu pour la balle rebondissante sur la surface sinusoïdale. Historique des variables p_x, p_y, v_x, v_y en fonction du temps	89
4.18	L'ensemble atteignable de la balle rebondissante sur une surface sinusoïdale en 3D, (a) : à gauche sans la fusion de trajectoires et (b) : à droite avec la fusion des trajectoires pour le critère longueur des segments.	90
4.19	Automate hybride du système de manœuvre anti-collision	92
4.20	Résultats système de manœuvre anti-collision	93
4.21	L'ensemble atteignable obtenu pour l'intégrateur non-holonome. Historique des variables x, y, z en fonction du temps	94
4.22	Résultats intégrateur non-holonome	95
4.23	Automate hybride-bench3-cas 1	96
4.24	Automate hybride-bench3-cas 2	97
4.25	Composante <i>G</i> de l'ensemble atteignable pour les cas 1 et 2 du benchmark Glycémie.	97
4.26	Les figures (a-c) obtenus avec Flow* et les figures (b-d) obtenus avec notre outil	99
5.1	Prédiction et atteignabilité hybride	112
5.2	Prédiction-correction hybride	113
5.3	Projection du zonotope $Z(c + \blacksquare(R))$ et de l'origine sur l'axe orienté suivant w	115
5.4	Critère pour trouver un vecteur de séparation	116

5.5	Automate hybride du système masse-ressort et Schéma du fonctionnement du système masse-ressort hybride	121
5.6	Expérience 1 : Estimation du mode (a), l'historique des variables x_1, x_2 estimées en fonction du temps (b)-(c) et la projection de l'ensemble atteignable dans le plan de phase (d) $x_2 \times x_1$, temps CPU=8.012s	122
5.7	Expérience 2 : Estimation du mode (a), l'historique des variables x_1, x_2 estimées en fonction du temps (b)-(c) et la projection de l'ensemble atteignable dans le plan de phase (d) $x_2 \times x_1$, temps CPU=83.888s	123
5.8	Le système à deux réservoirs	124
5.9	Estimation du mode (a), l'historique des variables x_1, x_2 estimées en fonction du temps (b)-(c) et la projection de l'ensemble atteignable dans le plan de phase (d) $x_2 \times x_1$, Temps CPU=1092.66s	125
5.10	Détection de défaut par approche prédiction/correction	126
5.11	le support strip d'un zonotope.	130
5.12	(en haut) Test d'inclusion par arithmétique des intervalles (a-b-c) pour la frontière F_j à l'instant t_j , (en bas) test d'inclusion avec les zonotopes (d-e-f) pour la frontière F_j à l'instant t_j	131
5.13	Approximation intérieure de la solution (boîte rouge) dans l'espace des paramètres $p_2 \times p_1$, la couche d'incertitude (boîte bleu) et la projection de l'ensemble atteignable dans le plan de phase $x_2 \times x_1$	134
5.14	Approximation intérieure de la solution (boîte rouge) dans l'espace des paramètres $p_3 \times p_1$, la couche d'incertitude (boîte bleu) et la projection de l'ensemble atteignable dans le plan de phase $x_2 \times x_1$	135
5.15	Détection de défaut par estimation de paramètres, (a) : scénario de non défaut, intersection non vide entre l'ensemble des paramètres nominaux et estimés ; (b) : scénario de défaut, intersection vide entre l'ensemble des paramètres nominaux et estimés	136

PUBLICATIONS DE L'AUTEUR

COMMUNICATIONS DANS DES CONGRÈS INTERNATIONAUX AVEC ACTES ET COMITÉ DE LECTURE

1. M. Maïga, N. Ramdani, and L. Travé-Massuyès. Robust fault detection in hybrid systems using set-membership parameter estimation. IFAC International (*Symposium on Fault Detection Supervision and Safety of Technical Processes SAFEPROCESS'2015*). Article accepté.
2. M. Maïga, C. Combastel, N. Ramdani, and L. Travé-Massuyès. Nonlinear hybrid reachability using set integration and zonotope enclosures. Strasbourg, France, 2014. In Proc. of 13th *European Control Conference, ECC 2014*. pages 234–239.
3. M. Maïga, N. Ramdani, and L. Travé-Massuyès. A fast method for solving guard set intersection in nonlinear hybrid reachability. Florence, Italy, 2013. In Proc. of 52nd *IEEE Conference on Decision and Control, CDC 2013*. pages 508–513.
4. M. Maïga, E. Chanthery, and L. Travé-Massuyès. Hybrid system diagnosis : Test of the diagnoser HYDIAG on a benchmark of the international diagnostic competition DXC'2011. IFAC International (*Symposium on Fault Detection Supervision and Safety of Technical Processes SAFEPROCESS'2012*), Mexico (Mexico), 29-31 Août 2012, pp.271-276

REVUES INTERNATIONALES

1. M. Maïga, N. Ramdani, L. Travé-Massuyès and C. Combastel. Reachability analysis of uncertain nonlinear hybrid systems. *IEEE Transactions on Automatic Control*. Article soumis septembre 2014.
2. M. Maïga, N. Ramdani, L. Travé-Massuyès and C. Combastel. A CSP versus a zonotope-based method for solving guard set intersection in nonlinear hybrid reachability. *Mathematics in Computer Science*, 2014. Volume 8, pages 407-423.

WORKSHOP (PRÉSENTATION ORALE)

1. M. Maïga, N. Ramdani, and L. Travé-Massuyès. Improvements on flow/guard intersection for non linear hybrid reachability . Manifestation sans acte : *Small Workshop on Interval Methods (SWIM 2013)*, Brest (France), 5-7 Juin 2013.
2. M. Maïga, N. Ramdani, and L. Travé-Massuyès. First results on nonlinear hybrid reachability combining interval Taylor method and IBEX library. Manifestation sans acte : *Small Workshop on Interval Methods (SWIM 2012)*, Oldenburg (Allemagne), 4-6 Juin 2012.

CHAPITRE 1

INTRODUCTION

1.1 CONTEXTE ET MOTIVATION

Le développement du diagnostic de défauts à base de modèle démarre simultanément dans plusieurs équipes de recherche au début des années 1970. L'intérêt pour les méthodes de détection et localisation de défauts dans les systèmes dynamiques n'a cessé de croître au cours des deux dernières décennies du fait de la forte nécessité d'accroître la robustesse des systèmes. Cet impératif est motivé non seulement par les contraintes liées à la sécurité mais aussi par des critères économiques.

Les méthodes de détection et de localisation de défauts à base de modèles se fondent sur la détection de divergences entre le comportement du système et de son modèle. La modélisation des systèmes dynamiques requiert la prise en compte d'incertitudes liées à l'existence inévitable de bruits (bruits de mesure, bruits sur la dynamique), à la méconnaissance de certains phénomènes perturbateurs mais également de la valeur exacte des paramètres (spécification dans certaines tolérances, phénomène de vieillissement). Alors que certaines de ces incertitudes se prêtent bien à une modélisation de type statistique, d'autres se caractérisent par des bornes, sans autre attribut.

Ainsi les méthodes ensemblistes, permettant de manipuler directement des sous-ensembles de nombres réels, se révèlent pertinentes dans ce contexte à incertitude bornée. Ces dernières années, les méthodes ensemblistes ont significativement progressé d'un point de vue fondamental, notamment en ce qui concerne l'intégration de modèles différentiels et les calculs d'atteignabilité. Elles ont également montré leur intérêt dans de nombreux domaines : analyse de stabilité, synthèse de lois de commande, validation, sûreté de fonctionnement, estimation, diagnostic, etc.

Les systèmes dynamiques hybrides sont des systèmes complexes pour lesquels coexistent des dynamiques continues et des dynamiques événementielles (Alur et al., 1995). Les systèmes

embarqués en sont un exemple en mettant en œuvre des systèmes de commande numérique temps-réel les faisant interagir avec leur environnement, régi par des lois physiques de nature continue. Des processus continus sont en interaction avec des processus discrets qui assurent leur supervision. Utilisés pour la supervision et la planification du système à l'échelle globale, les processus discrets déclenchent des changements de modes opératoires qui peuvent aller jusqu'à la modification de la structure du système, alors que les processus continus expriment la commande et la réaction du système sur une échelle de temps continu. Les systèmes complexes exhibant des dynamiques continues par morceaux peuvent être modélisés comme des systèmes dynamiques hybrides, et des systèmes multi-physiques complexes peuvent être modélisés de manière plus simple sous la forme de systèmes continus par morceaux, à commutation ou hybrides. On trouve ainsi des systèmes à dynamiques hybrides dans des domaines aussi variés que les réseaux de transports (Alur and Pappas, 2004), l'industrie automobile (Johansson, 2003; Pepyne and Cassandra, 2000), la robotique (Ames et al., 2009), la médecine et la biologie (Dounias and Linkens, 2004; Fisher, 1991; Furler et al., 1985; Piazza et al., 2005).

Ainsi, pour un grand nombre d'applications, Il est nécessaire d'implanter un système de surveillance afin de détecter, d'isoler voire d'identifier tout dysfonctionnement à court terme. Le système de surveillance doit réaliser les trois tâches suivantes :

- La détection : elle consiste à prendre une décision binaire : soit le système fonctionne correctement, soit un défaut (défaillance) s'est produit. Le cas échéant, la procédure doit déterminer l'instant d'occurrence du défaut.
- La localisation : elle consiste à déterminer le composant défectueux.
- L'identification : elle consiste à déterminer la nature exacte de la défaillance en vue de déterminer le type de maintenance ou de correction (accommodation, reconfiguration) à réaliser sur l'installation.

Les méthodes de détection de défauts et diagnostic des systèmes hybrides sont généralement des extensions de celles existant pour les systèmes continus ou les systèmes à événements discrets. Parmi les extensions des approches issues des systèmes continus, nous pouvons citer l'estimation paramétrique, les relations de redondance analytique ou les observateurs. Un défaut sera détecté si l'état hybride reconstruit à partir des mesures devient incohérent avec l'état prédit par un modèle de fonctionnement nominal, ce qui se traduit par un «résidu» non nul, et par l'estimation d'un mode discret correspondant à un mode de faute.

Dans le cadre de cette thèse, nous nous intéresserons à la surveillance préventive des systèmes hybrides dans un contexte à «erreurs inconnues» mais bornées (voir figure 1.1), contexte dans lequel nous faisons l'hypothèse que toutes les perturbations agissant sur le système et toutes les incertitudes affectant le modèle sont comprises dans un ensemble défini par des bornes connues, mais sans autre connaissance quant à leur distribution dans cet ensemble (Milanese et al., 1996; Schweppe, 1968). On parle alors de *systèmes à incertitude bornée* ou SIB. La solution du problème d'estimation correspond à un ensemble (éventuellement formé d'une union de sous-ensembles non connexes) de vecteurs d'état hybrides cohérents avec le modèle retenu, les mesures et les bornes d'erreurs a priori.

Nous proposons de fonder la détection et la localisation des défauts sur une estimation de l'état hybride incluant une estimation ensembliste de la composante continue. Par ailleurs, des travaux récents ont montré qu'il était possible d'estimer les configurations discrètes d'un automate hybride par le biais de méthodes ensemblistes (Benazera and Travé-Massuyès, 2009) et que cette représentation de l'incertitude permet de réduire significativement le nombre d'hypothèses de diagnostic.

Le calcul de l'ensemble atteignable par un système à dynamique hybride, c'est-à-dire la détermination de la partie de l'espace d'état hybride atteignable par le système à partir d'un domaine initial, ensemble de volume non nul, tout en tenant compte de manière explicite de toutes les erreurs et perturbations agissant sur le système et des possibles variations des paramètres caractérisant le système étudié, est au cœur des méthodes ensemblistes d'estimation et d'observation des systèmes continus et hybrides (par exemple (Meslem and Ramdani, 2011; Meslem et al., 2010a)). Les analyses d'atteignabilité concernant un système dynamique permettent également de procéder à la vérification de propriétés telles que la stabilité.

La nature garantie des estimations réalisées par les approches à erreur bornée a l'avantage d'aboutir à une vérification fiable et garantie, qui s'avère très pertinente en termes de sécurité et performance puisque l'on peut alors théoriquement étudier tous les comportements possibles du système. On peut s'assurer par exemple que le système ne peut pas atteindre un ensemble d'états «dangereux». Le lecteur peut se référer par exemple aux articles (Althoff and Krogh, 2012; Alur et al., 1995; Chutinan and Krogh, 2003; Guéguen et al., 2009; Tomlin et al., 2003) et aux références qu'ils citent. Un état «dangereux» pourrait être par exemple un état dans lequel une distance critique entre deux robots mobiles n'est plus respectée ; ou un état dans lequel une concentration à ne pas excéder pour certains produits chimiques est atteinte dans un réacteur. D'autres applications possibles de l'analyse d'atteignabilité, tout aussi importante que les vérifications liées à la sécurité, peuvent être citées, sans être exhaustives :

- Ordonnancement : l'analyse d'atteignabilité peut vérifier si l'ordonnancement optimal d'un système (généralement un système de production) est assuré sous toutes les conditions, ou si par exemple le contrôleur de supervision peut fonctionner en mode de récu-

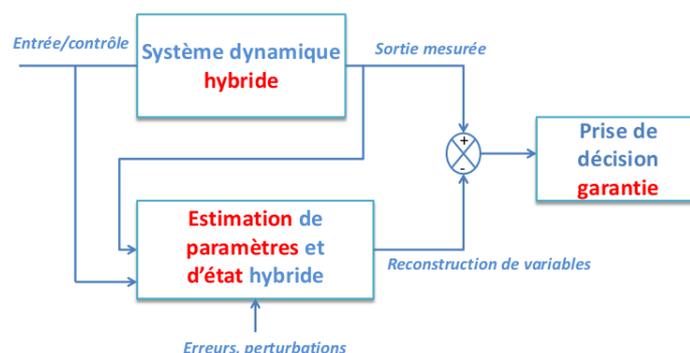


FIGURE 1.1 – Surveillance des systèmes dynamiques hybrides dans un contexte à erreur bornée

pération (Althoff, 2010).

- Synthèse de contrôleurs : l'analyse d'atteignabilité peut être utilisée pour trouver l'ensemble de paramètres de contrôle qui satisfont des contraintes de stabilité ou de performance (par exemple (Donzé et al., 2009)).
- Blocages : l'analyse d'atteignabilité peut déterminer si un système peut rester bloqué dans une certaine région de l'espace d'état continu ou dans un mode de fonctionnement donné (Althoff, 2010).

Des outils logiciels existent, dont certains en *open source ou logiciel libre*, pour réaliser l'analyse d'atteignabilité pour des systèmes hybrides. Pour les systèmes hybrides linéaires, on peut citer entre autres HyTech (Henzinger et al., 1997) SpaceX (Frehse et al., 2011) et PHAVer (Frehse, 2005) ; et pour les systèmes hybrides non linéaires, d/dt (Asarin et al., 2001) qui est fondé sur une hybridation linéaire sur des grilles, CheckMate (Chutinan and Krogh, 2003), fondé sur Matlab (et par conséquent non garanti), HySon (Bouissou et al., 2012), HydLogic (Ishii et al., 2011) et Flow* (Chen et al., 2012a) qui est un outil fondé sur des modèles de Taylor intervalles.

1.2 STRUCTURE DE LA THÈSE

Après ce premier chapitre introductif qui présente le contexte du travail et motive le sujet, le chapitre 2 présente les différentes problématiques abordées dans la thèse, à savoir l'atteignabilité hybride non linéaire, l'estimation de paramètres, l'observation d'état hybride et le diagnostic des systèmes hybrides. Il rappelle ensuite les outils mathématiques utilisés : l'analyse par intervalles et les zonotopes, les méthodes d'intégration numérique garantie des équations différentielles ordinaires incertaines ainsi que l'inversion ensembliste. Le chapitre 3 est consacré à l'état de l'art des méthodes d'atteignabilité linéaire et non linéaire. Nous évoquons les différentes méthodes de résolution du passage des gardes¹ qui sont présentes dans la littérature. Le chapitre 4 regroupe la première contribution de la thèse. Celle-ci consiste en une méthode efficace et rapide pour l'atteignabilité hybride non linéaire. Elle repose sur l'utilisation de deux méthodes robustes : une méthode pour le passage ensembliste des gardes et une méthode de fusion de trajectoires. Le chapitre 5 aborde la deuxième contribution, à savoir l'estimation ensembliste de paramètres et l'observation hybride (mode et état continu). Pour l'estimation d'état hybride, nous avons proposé une méthode basée sur un contacteur construit au dessus de notre méthode d'atteignabilité hybride non linéaire. En utilisant notre méthode d'atteignabilité hybride non linéaire et un algorithme d'inversion ensembliste SIVIA-H, notre contribution permet de calculer de manière garantie l'ensemble des paramètres cohérents avec le modèle hybride, l'ensemble des mesures et les bornes d'erreurs. Enfin dans la dernière partie, nous concluons la thèse avec un résumé des résultats et une perspective sur les travaux futurs.

1. conditions qui définissent les transitions entre modes d'un système hybride

CHAPITRE 2

NOTIONS FORMELLES ET OUTILS MATHÉMATIQUES

Sommaire

2.1 Fondamentaux et problèmes traités	13
2.1.1 Définition d'un système dynamique hybride	14
2.1.2 Formalisme d'un système dynamique hybride	14
2.1.3 Atteignabilité hybride	17
2.1.4 Observation ensembliste des systèmes hybrides	19
2.1.5 Estimation ensembliste de paramètres des systèmes hybrides	20
2.2 Outils mathématiques pour le calcul ensembliste	20
2.2.1 Arithmétique des intervalles	21
2.2.2 Résolution des Problèmes de satisfaction de contraintes par arithmétique des intervalles	23
2.2.3 Notion de contracteur : Hc4-revise	23
2.3 Inversion ensembliste	25
2.3.1 Opérations sur les zonotopes	27
2.4 Intégration numérique garantie d'équations différentielles : tube de trajectoires	30
2.4.1 Méthode de Lohner : factorisation QR	33
2.4.2 Représentation zonotopique et domaine de solution	36

2.1 FONDAMENTAUX ET PROBLÈMES TRAITÉS

2.1.1 DÉFINITION D'UN SYSTÈME DYNAMIQUE HYBRIDE

Les systèmes dynamiques hybrides continu-discret (SDH) (Alur et al., 1995; Antsaklis and Nerode, 1998; Van Der Schaft et al., 2000) sont des systèmes dynamiques faisant intervenir explicitement et simultanément des phénomènes ou des modèles de type dynamique continu et événementiel. Ils peuvent être constitués de processus continus interagissant avec des processus discrets ou encore supervisés par ces derniers. Ils peuvent aussi être le fruit de l'interaction entre des algorithmes continus, utilisés pour la commande de sous-systèmes dynamiques et d'algorithmes discrets, utilisés pour la supervision et la planification à l'échelle globale du système. Par ailleurs, un grand nombre de systèmes dynamiques exhibant des dynamiques discontinues, c'est-à-dire continues par morceaux, peuvent être modélisés par des SDH. Une des modélisations possibles qui tient compte de ce double aspect, continu et discret, consiste à modéliser les SDH par un automate hybride, c'est-à-dire un ensemble d'équations différentielles gouvernées par un automate à états finis (Alur et al., 1995, 1992; Guéguen and Zaytoon, 2004).

2.1.2 FORMALISME D'UN SYSTÈME DYNAMIQUE HYBRIDE

Dans ce mémoire, nous modéliserons un système dynamique hybride par un automate hybride (voir la figure 2.1) (Alur et al., 1995) défini par $H = (\mathcal{Q}, \mathcal{D}, \mathcal{P}, \Sigma, \mathcal{A}, \text{Inv}, \mathcal{F})$

- \mathcal{Q} est l'ensemble des modes discrets (de fautes ou nominaux) de l'automate. Pour un mode q fixé, la dynamique continue, et donc les transitions continues, sont décrites par des équations différentielles non autonomes, non linéaires, $f_q \in \mathcal{F}$ de la forme

$$\text{flow}(q) : \quad \dot{x}(t) = f_q(x, p, t) \quad (2.1)$$

où $f_q : \mathcal{D} \times \mathcal{P} \times \mathbb{R}^+ \mapsto \mathcal{D}$ est non linéaire, et continûment dérivable plusieurs fois sur $\mathcal{D} \subseteq \mathbb{R}^n$, la dimension n pouvant dépendre de q , et $p \in \mathcal{P}$, où \mathcal{P} est un domaine d'incertitude pour le vecteur de paramètres p .

- Inv est un invariant¹, qui attribue un domaine à la composante continue de l'état pour chaque mode. Il est défini par le système d'inégalités suivantes :

$$\text{Inv}(q) : \quad v_q(x(t), p, t) < 0, \quad (2.2)$$

où $v_q : \mathcal{D} \times \mathcal{P} \times \mathbb{R}^+ \mapsto \mathbb{R}^m$ est une fonction vectorielle non linéaire, (l'inégalité s'appliquant composante par composante), et le nombre m d'inégalités pouvant ainsi dépendre de q .

- \mathcal{A} est l'ensemble des transitions discrètes $\{e = q \rightarrow q'\}$, caractérisée chacune par le 5-uplet $(q, \text{guard}, \sigma, \rho, q')$, où q et q' représentent chacun le mode de départ et d'arrivée après la transition discrète ; guard est une condition donnée sous la forme du système d'équations :

$$\text{guard}(e) : \quad \gamma_e(x(t), p, t) = 0; \quad (2.3)$$

σ est un événement, et ρ est une fonction de réinitialisation (ou de reset).

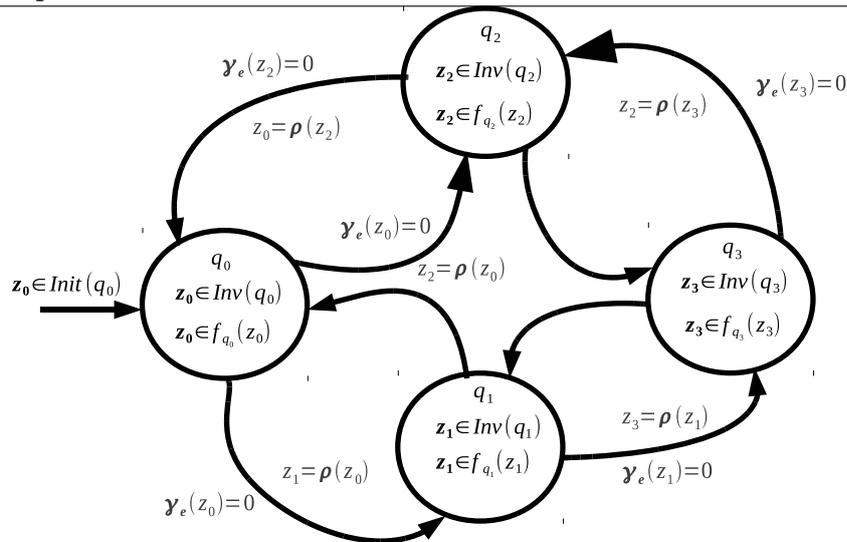


FIGURE 2.1 – Système hybride.

Une transition discrète $q \rightarrow q'$ a lieu si le flux d'état continu satisfait les conditions de garde(2.3).

En introduisant une variable d'état $z = (x, p, t)$ avec $\dot{z} = (\dot{x}, 0, 1)$, et en définissant $\mathcal{Z} = \mathcal{D} \times \mathcal{D} \times \mathbb{R}^+$, les équations (2.1–2.2–2.3) peuvent être réécrites sous la forme :

$$\text{flow}(q) : \dot{z}(t) = f_q(z), \quad (2.4)$$

$$\text{Inv}(q) : \nu_q(z(t)) < 0 \quad (2.5)$$

$$\text{guard}(e) : \gamma_e(z(t)) = 0. \quad (2.6)$$

Ainsi toutes les quantités incertaines sont incorporées dans le vecteur d'état initial.

La figure 2.2 illustre l'automate modélisant la dynamique hybride de la balle rebondissante en 1D. Dans cet exemple, un seul mode avec un invariant $x \geq 0$ et des équations du flux $\dot{x} = v$ et $\dot{v} = -g$. La position x et la vitesse v sont les variables continues du système et dépendant du temps. La balle chute librement jusqu'à ce qu'elle atteigne le sol ($x = 0$). Si cela se produit avec une vitesse négative, la balle rebondit et la fonction de reset est appliquée : la vitesse change de direction et la balle perd une partie de son énergie, la vitesse après réinitialisation est $v' = -e.v$, $e \in [0, 1]$, où e et g sont des constantes. La figure 2.2 montre aussi le résultat de l'analyse d'atteignabilité de la balle rebondissante et illustre l'interdépendance du flux et des transitions discrètes.

1. Ici un invariant désigne une fonction qui définit la frontière de l'évolution du vecteur d'état continu, souvent spécifié sur des considérations pratiques

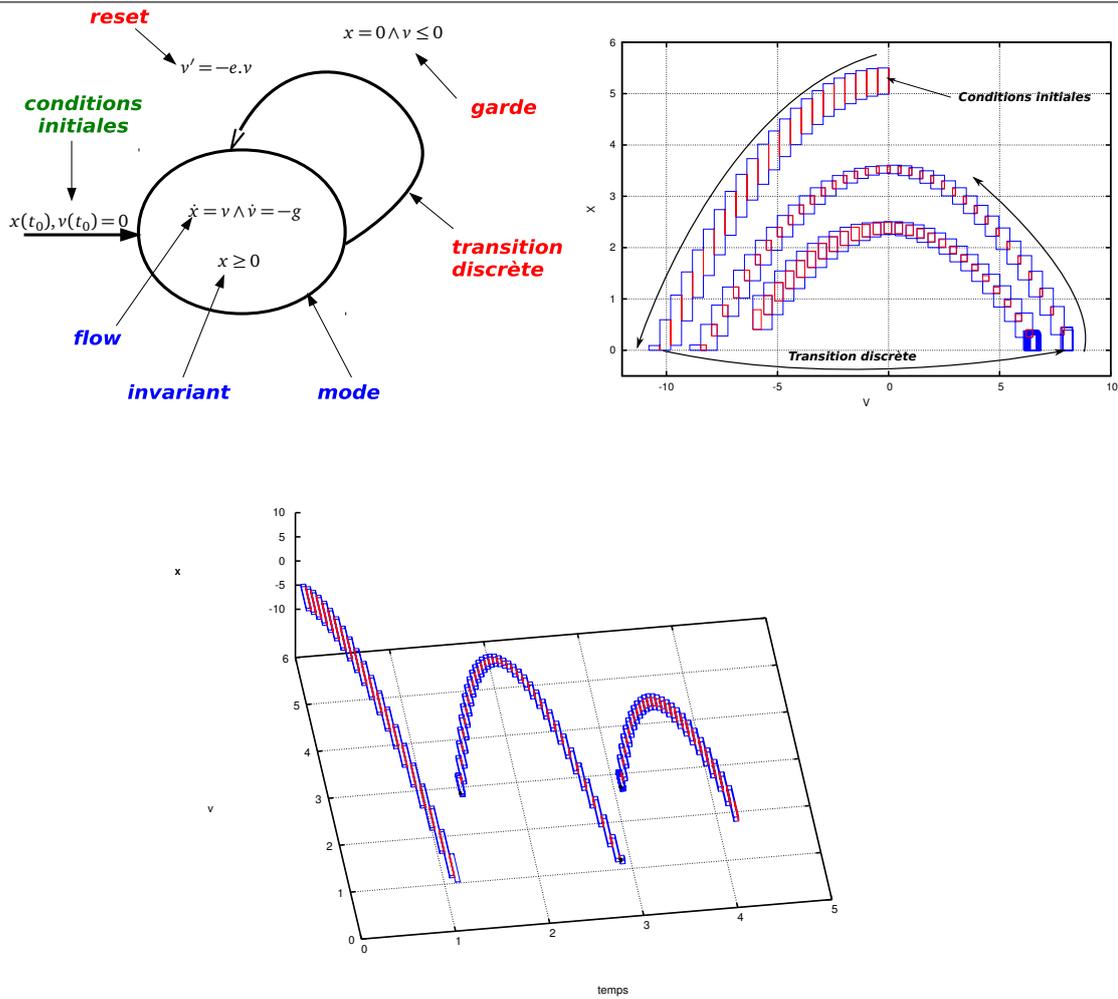


FIGURE 2.2 – Balle rebondissante en 1D modélisée comme un automate hybride, l'ensemble atteignable dans l'espace $x \times v$ en 2D et dans l'espace $x \times v \times t$ en 3D en un temps fini.

Lorsque nous étudierons l'estimation (paramétrique et/ou d'état hybride) des systèmes hybrides, nous considérerons l'équation de mesure suivante :

$$y(t) = g_q(x(t), p, t), \quad (2.7)$$

où la fonction $g_q : \mathbb{D} \times \mathbb{P} \times \mathbb{R}^+ \mapsto \mathbb{R}^m$ peut être non linéaire. En faisant l'hypothèse que les mesures du vecteur de sortie y_j (2.7) sont accessibles sur une grille temporelle $t_0 < t_1 < t_2 < \dots < t_n$, et le pas de temps de mesure n'est pas nécessairement constant. Toutes les incertitudes et perturbations seront prises inconnues mais bornées. On notera \mathbb{Y}_i le domaine admissible du vecteur de sortie y_i à l'instant t_i .

$$\mathbb{Y}_i = y_i + \mathbb{E}_i \quad (2.8)$$

où \mathbb{E}_i le domaine admissible des erreurs à l'instant t_i .

2.1.3 ATTEIGNABILITÉ HYBRIDE

Dans cette partie, nous nous intéresser au calcul explicite de l'ensemble atteignable par un système hybride non linéaire à incertitudes bornées (2.1–2.2–2.3). L'ensemble atteignable du système (voir la figure 2.3) (2.1–2.2–2.3) sur un horizon temporel $[t_0, t]$ est défini comme suit :

$$\mathcal{R}([t_0, t]; \mathcal{X}_0) \equiv \left\{ \mathcal{X}(t; t_0, \mathcal{X}_0) \mid \dot{x}(t) = f_q(x, p, t) \wedge x(t) \in \text{Inv}(q) \right. \\ \left. \wedge \gamma_e(x(t), p, t) = 0 \wedge (x(t_0) \times p \times t_0) \in \mathcal{X}_0 \right\}, \quad (2.9)$$

où $\mathcal{X}(t; t_0, \mathcal{X}_0)$ est l'ensemble solution de (2.1–2.2–2.3) à l'instant t provenant de chaque condition initiale dans $\mathcal{X}_0 = [\underline{z}_0, \bar{z}_0]$ à t_0 .

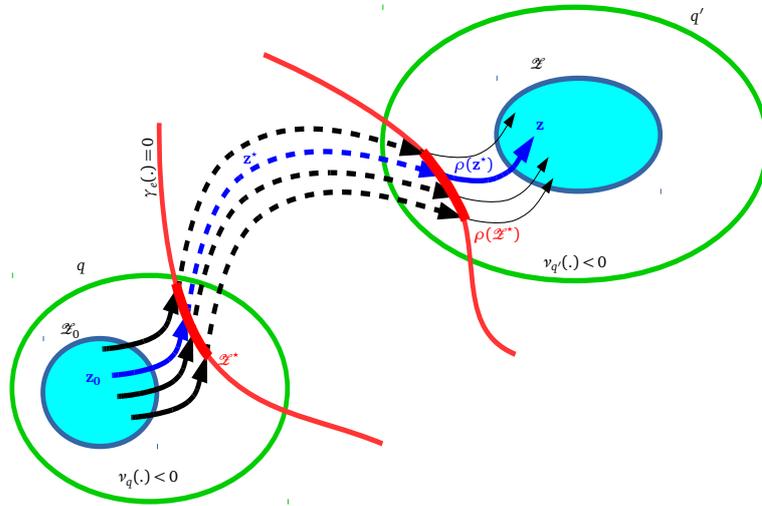


FIGURE 2.3 – Ensemble atteignable par un système hybride en temps fini (2.1–2.2–2.3).

Il s'agit principalement de combiner les méthodes de Taylor intervalle pour le calcul de l'expansion continue d'un ensemble (détaillées dans la partie 2.4) avec les techniques de résolution de problèmes de satisfaction de contraintes (rappelées dans la partie 2.2.2) pour détecter et localiser l'intersection du flux continu avec les frontières de l'invariant et avec les conditions de garde.

La transition discrète $e = q \rightarrow q'$ a lieu lorsque le flux continu $z(t)$ vérifie la condition de garde (2.3) à t_e . L'idée est de trouver toutes les combinaisons $t_e \times z(t_e)$ solutions de (2.3). Nous disposons par le biais de notre solveur d'équations différentielle, noté $\varphi^{QR}()$ (voir l'algorithme 3), d'une expression analytique de la solution telle que fournie par le développement en série de Taylor (2.24) pour l'intervalle $[t_j, t_{j+1}]$. Il suffit de trouver toutes les conditions initiales $z_j^* \in z_j$ qui conduisent à un vecteur $z(t_e; t_j, z_j^*)$ solution de (2.3) à $t_e = t^*$ (voir figure 2.4) ; ceci n'est rien d'autre que l'ensemble

$$[\underline{t}^*, \bar{t}^*] \times \mathcal{Z}_j^*$$

où

$$\mathcal{Z}_j^* = \bigcup_{j=1} (z_j^*) \subset \mathcal{Z}_j$$

est l'ensemble initial des vecteurs d'état initiaux z_j^* qui conduisent à $z(t_e; t_j, z_j^*)$ et qui satisfont (2.3) à $t_e \in [t_j, t_{j+1}]$. Par conséquent, il suffit de filtrer le domaine $[z_j]$, en y retirant toutes les

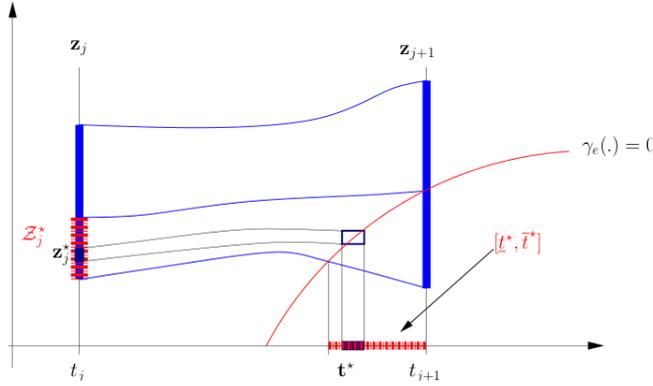


FIGURE 2.4 – Résolution de l'intersection du tube de trajectoires et de la garde. L'objectif est de résoudre un CSP de manière à déterminer l'ensemble des conditions initiales z_j^* qui conduit le vecteur $z(t^*; t_j, z_j^*)$ à intersecter la garde à l'instant t^* . L'ensemble solution du CSP est donné par toutes les combinaisons possibles de $z_j^* \times t^*$ (Ramdani and Nedialkov, 2011).

parties qui génèrent une trajectoire qui n'intersecte pas la garde pour tout $t_e \in [t_j, t_{j+1}]$. On peut donc obtenir facilement l'ensemble $[\underline{t}^*, \bar{t}^*] \times \mathcal{Z}_j^*$ en résolvant un problème de satisfaction de contraintes (CSP) (détaillé dans la partie 2.2.2).

L'algorithme d'atteignabilité est au cœur de mon travail de thèse et de la majorité des méthodes d'estimation ensemblistes. Nous allons donner l'idée principale de chaque méthode proposée :

- Nous avons proposé une méthode de complexité polynomiale pour la résolution du passage des gardes ensemblistes, dans le cas où la garde est linéaire. Cette méthode est basée sur des opérations géométriques d'intersection entre zonotope et hyperplan. L'avantage de cette méthode réside dans le fait qu'elle est réduite à des calculs d'algèbre linéaire simples (principalement, l'évaluation du noyau) qui assure une complexité polynomiale et des temps de calcul réduits qui favorisent le passage à l'échelle.
- Si la première méthode est restreinte aux gardes linéaires, la deuxième est beaucoup plus générale (gardes non linéaires). Nous avons donc comparé cette première méthode

avec une deuxième basée sur la propagation de contraintes pour la résolution des problèmes de satisfaction de contraintes. Cette méthode consiste à procéder à la bisection dans la seule direction du temps et à utiliser un contracteur, HC4_Revise de IBEX (Charbert, 2007), pour réduire le domaine des vecteurs d'état localisés sur la garde, durant la tranche de temps étudiée. En d'autres termes, nous réalisons la bisection uniquement dans la direction de la variable temporelle pour améliorer la précision de la détection de l'instant de passage de garde. En bisectant dans une seule direction, nous pouvons ainsi assurer une complexité polynomiale pour l'algorithme d'atteignabilité hybride. Finalement, le contracteur est utilisé sur la tranche du tube de trajectoire correspondant à un intervalle de temps que nous pouvons rendre aussi fin que possible.

- Ensuite, nous avons proposé une méthode générale d'atteignabilité hybride qui inclut la résolution rapide et efficace du passage des gardes ensemblistes, la fusion des trajectoires, et l'amélioration du saut discret à travers la fonction de reset. Notre méthode de passage rapide et efficace des gardes ensemblistes ajoute à la bisection dans la seule direction du temps la nouveauté de faire collaborer plusieurs contracteurs simultanément, pour mieux réduire le domaine des vecteurs d'état localisés sur la garde, durant la tranche de temps étudiée.

Nous allons revenir plus en détails sur ces contributions majeures de notre algorithme dans le chapitre 4.

2.1.4 OBSERVATION ENSEMBLISTE DES SYSTÈMES HYBRIDES

L'observation des systèmes hybrides vise à reconstruire simultanément le mode discret et la composante continue de l'état hybride, et donc à reconstruire la séquence de commutation de l'automate hybride.

L'estimation à «erreur bornée» du système hybride calcule une sur-approximation des domaines faisables de la séquence de commutation et de la composante continue de l'état hybride, cohérents avec le modèle hybride retenu, les hypothèses a priori sur les erreurs et les domaines faisables des mesures (voir la figure 2.5). De manière formelle, étant données deux mesures incertaines \mathbb{Y}_j et \mathbb{Y}_{j+1} collectées à t_j et t_{j+1} , le problème de l'estimation consiste à effectuer simultanément les deux opérations suivantes, pour tout j , $0 \leq j \leq n_T - 1$:

- (i) Détecter et localiser toutes les transitions possibles (t^*, e) qui peuvent survenir à l'instant $t^* \in [t_j, t_{j+1}]$. Compte tenu de la présence d'incertitudes dans les conditions initiales, les dynamiques continues et les données mesurées, il peut exister un continuum d'instantants de commutation. Il peut aussi exister plusieurs sous-intervalles de commutation dans $[t_j, t_{j+1}]$.
- (ii) Reconstruire les ensembles de vecteurs d'état hybrides $(q(t_j), \mathcal{X}^c(t_j))$ et $(q(t_{j+1}), \mathcal{X}^c(t_{j+1}))$ cohérents avec la séquence de commutation reconstruite dans (i), la modélisation (1)–(5), et les inclusions $y(t_j) \in \mathbb{Y}_j$ et $y(t_{j+1}) \in \mathbb{Y}_{j+1}$.

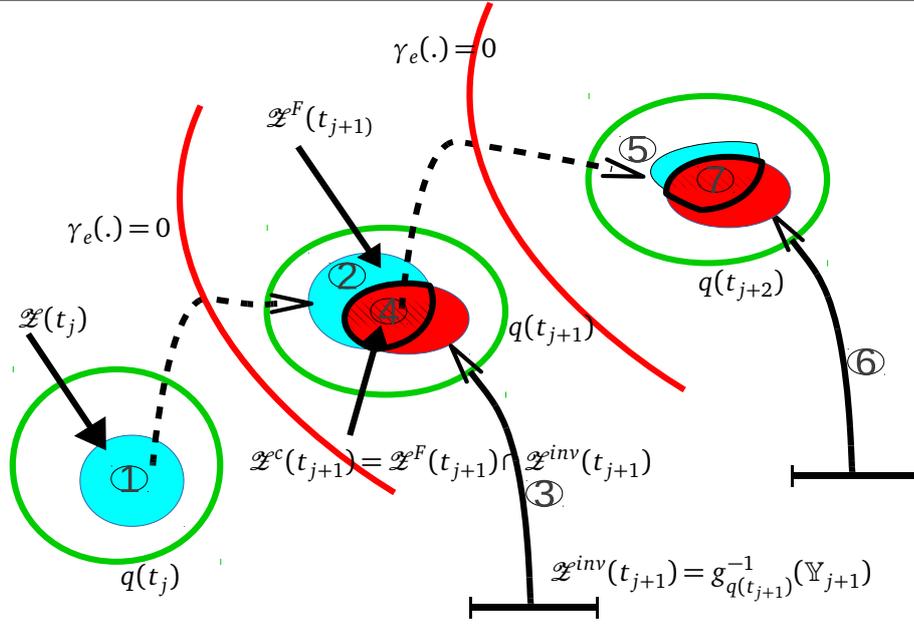


FIGURE 2.5 – Observateur de système hybride basé sur la prédiction/correction ; où $\mathcal{Z}^F(t_{j+1})$ est la prédiction à un pas de l'ensemble $\mathcal{Z}(t_j)$, $\mathcal{Z}^{inv}(t_{j+1})$ est l'ensemble obtenu par inversion ensembliste à l'instant t_{j+1} à partir de \mathbb{Y}_{j+1} , $\mathcal{Z}^c(t_{j+1})$ est l'ensemble corrigé avec les mesures à l'instant t_{j+1} .

2.1.5 ESTIMATION ENSEMBLISTE DE PARAMÈTRES DES SYSTÈMES HYBRIDES

Étant donné le modèle hybride (2.1–2.2–2.3), les mesures (2.7) et les bornes des incertitudes de mesure, le but de l'estimation de paramètres consiste à estimer le vecteur de paramètres \mathbf{p} en déterminant l'ensemble \mathcal{S} de tous les paramètres acceptables.

$$\mathcal{S} = \{\mathbf{p} \in \mathcal{P}_0 \mid (\forall t \in [t_0, T_{end}], \mathcal{R}([t_0, T_{end}]; \mathcal{Z}_0) \wedge (\forall t_j \in \{t_1, t_2, \dots, t_n\}, g_q(x, \mathbf{p}, t_j) \in \mathbb{Y}_j)\} \quad (2.10)$$

où l'ensemble \mathcal{P}_0 est un ensemble de recherche a priori.

La caractérisation de l'ensemble solution \mathcal{S} est un problème d'inversion ensembliste c'est à dire qu'à partir de l'ensemble des mesures \mathbb{Y} , nous voudrions reconstruire l'ensemble des paramètres \mathcal{S} .

2.2 OUTILS MATHÉMATIQUES POUR LE CALCUL ENSEMBLISTE

Dans cette section, nous allons présenter les concepts clés concernant les méthodes fondées sur l'arithmétique des intervalles que nous utilisons pour calculer les intersections du tube

de trajectoires avec les invariants ou conditions gardées, et pour évaluer les fonctions de reset. Les zonotopes sont également introduits parce que nous utilisons des zonotopes particuliers pour limiter la sur-approximation résultant de l'enveloppement d'un ensemble de forme arbitraire par une boîte alignée avec les axes. Nous allons aussi présenter les méthodes d'intégration numérique garantie des équations différentielles ordinaires incertaines ainsi que les méthodes d'inversion ensembliste.

2.2.1 ARITHMÉTIQUE DES INTERVALLES

L'objectif de l'arithmétique des intervalles est de manipuler des intervalles de nombres réels. On parle alors de calculs garantis ou validés. Dans cette thèse, les intervalles sont utilisées pour prendre en compte les incertitude sur les états continus et les paramètres d'un système dynamique.

Un intervalle $[u] = [\underline{u}, \bar{u}]$ est un fermé borné de \mathbb{R} où \underline{u} représente la borne inférieure et \bar{u} représente la borne supérieure. L'intervalle unitaire est noté $\mathbf{B} = [-1, +1]$.

Pour un intervalle $[u]$ donné, on définit :

- le point milieu : $mid([u]) = (\bar{u} + \underline{u})/2$
- la largeur : $w([u]) = \bar{u} - \underline{u}$
- son rayon : $rad([u]) = (\bar{u} - \underline{u})/2$

Deux intervalles $[u]$ et $[v]$ sont égaux si et seulement si $\underline{u} = \underline{v}$ et $\bar{u} = \bar{v}$. Les opérations arithmétiques réelles peuvent être étendues aux intervalles (Moore, 1996) et définies par :

$$\circ \in \{+, -, *, /\}, [u] \circ [v] = [\{x \circ y \mid x \in [u], y \in [v]\}].$$

On les appelle les extensions naturelles des opérations classiques.

2.2.1.1 VECTEURS ET MATRICES INTERVALLES

Un vecteur intervalle (boîte ou pavé) $[X]$ est le produit cartésien de n intervalles :

$$[X] = [x_1] \times [x_2] \times \dots \times [x_n]$$

Une boîte unitaire notée \mathbf{B}^n , est une boîte de n intervalles unitaires. De même, nous notons par $[A] \in \mathbb{IR}^{n \times m}$ une matrice intervalle de n lignes et de m colonnes dont les éléments sont des intervalles $[a_{i,j}]$, $i = 1, \dots, n$ et $j = 1, \dots, m$.

2.2.1.2 SOUS-PAVAGES ET PAVAGES

Un sous-pavage est défini comme une union de pavés. En particulier, un sous-pavage régulier est constitué d'un ensemble de pavés disjoints ou de pavés qui ne partagent que leurs frontières (Jaulin et al., 2001).

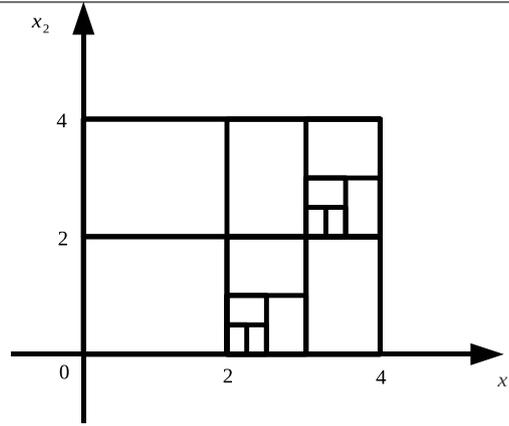


FIGURE 2.6 – Exemple d'un sous-pavage.

Sur la figure 2.6, un exemple de sous-pavage recouvrant le domaine $[0, 4] \times [0, 4]$ est : $[0, 2] \times [0, 2]$, $[2, 4] \times [0, 2]$, $[2, 3] \times [0, 1]$. Lorsqu'un sous-pavage \mathcal{K} recouvre tout le pavé $[x]$, \mathcal{K} est appelé pavage de $[x]$.

2.2.1.3 OPÉRATION DE BISSECTION

Soit un pavé $[X] \in \mathbb{I}\mathbb{R}^n : x_1] \times [x_2] \dots \times [x_n]$. Définissons les pavés :

$$\begin{cases} L[x] = [x_1] \times [x_2] \times \dots \times \underline{[x_j, x_{oj}]} \times \dots \times [x_n] \\ R[x] = [x_1] \times [x_2] \times \dots \times [x_{oj}, \overline{x_j}] \times \dots \times [x_n] \\ j \in [1 : n] \end{cases} \quad (2.11)$$

Alors, l'opération qui consiste à obtenir les pavés $L[x]$ et $R[x]$ est appelée bisection (Jaulin et al., 2001).

2.2.1.4 FONCTION D'INCLUSION

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, l'extension de la fonction f sur le vecteur intervalle $[U]$ est donnée par :

$$f([U]) = \{f(x) | x \in [U]\}.$$

La fonction intervalle $[f] : \mathbb{I}\mathbb{R}^n$ vers $\mathbb{I}\mathbb{R}^m$ est une fonction d'inclusion de f si :

$$\forall [U] \in \mathbb{I}\mathbb{R}^n, f([U]) \subseteq [f]([U]).$$

Une fonction d'inclusion de f peut être obtenue en remplaçant chaque occurrence de variable réelle par l'intervalle correspondant. En pratique la fonction d'inclusion n'est pas unique, elle dépend de l'expression formelle de f .

2.2.2 RÉOLUTION DES PROBLÈMES DE SATISFACTION DE CONTRAINTES PAR ARITHMÉTIQUE DES INTERVALLES

Considérons le système de m (in)égalités mettant en jeu n variables $z \in \mathbb{R}^n$:

$$\mathcal{C} : \bigwedge_{1 \leq i \leq m} h_i(z) \prec 0, \prec \in \{<, =\}, \quad (2.12)$$

ici l'opérateur \bigwedge désigne une conjonction de contraintes. On note \mathcal{X}_0 le domaine initial associé aux variables z . Le système (2.12) définit un problème de satisfaction de contraintes, que nous noterons :

$$\text{CSP} : (z, \mathcal{X}, \mathcal{C}). \quad (2.13)$$

En notant \mathcal{S} l'ensemble de ses solutions, on a :

$$\mathcal{S} = \{z \in \mathcal{X}_0 \mid \bigwedge_{1 \leq i \leq m} h_i(z) \prec 0\}. \quad (2.14)$$

Une approximation extérieure $\overline{\mathcal{S}} \supseteq \mathcal{S}$ et une approximation intérieure $\underline{\mathcal{S}} \subseteq \mathcal{S}$ peuvent être calculées de manière fiable et garantie en utilisant des algorithmes de recherche exhaustive combinant un algorithme de bisection, l'arithmétique des intervalles, et des fonctions de réduction (contacteurs intervalles). On pourra se rapporter aux références ((Chabert and Jaulin, 2009; Cleary, 1987; Davis, 1987; Jaulin et al., 2001; Waltz, 1975)) pour plus de détails.

2.2.3 NOTION DE CONTRACTEUR : HC4-REVISE

L'idée principale des contracteurs est de réduire la taille d'une boîte $[z]$ pour qu'elle sur estime de manière minimale l'ensemble solution d'un ensemble de contraintes \mathcal{C} sans avoir recours à la bisection. La réduction peut être faite par des opérateurs intervalles de réduction, appelés contracteurs sur $[z]$, que nous écrivons

$$[z]' = \text{Contractor}(\mathcal{C}, [z]).$$

Etant donné l'ensemble des contraintes \mathcal{C} donné par (2.12), cet opérateur supprime de $[z]$ un sous-ensemble qui ne contient pas de solutions en satisfaisant les propriétés suivantes : $[z]' \subseteq [z]$, et $[z]' \cap \mathcal{S} = [z] \cap \mathcal{S}$, où \mathcal{S} est l'ensemble solution défini par (2.12). La majorité des contracteurs utilisent des techniques de consistance (par exemple (Collavizza et al., 1999), voir aussi (Neumaier, 2004)) et/ou la propagation de contraintes (Chabert and Jaulin, 2009; Jaulin et al., 2001). Les techniques de propagation de contraintes sont basées sur l'extension intervalle du filtre local de Waltz (Cleary, 1987; Davis, 1987; Waltz, 1975). Les techniques de filtrage reposent sur les propriétés de consistance. HC4-Revise (Benhamou et al., 1999) effectue la projection automatique d'une contrainte c sur toutes ses variables en évitant la décomposition

de c en contraintes primitives². L'algorithme utilise une représentation en arbre de la contrainte : les feuilles sont des constantes ou variables, et les nœuds internes correspondent aux opérateurs de base.

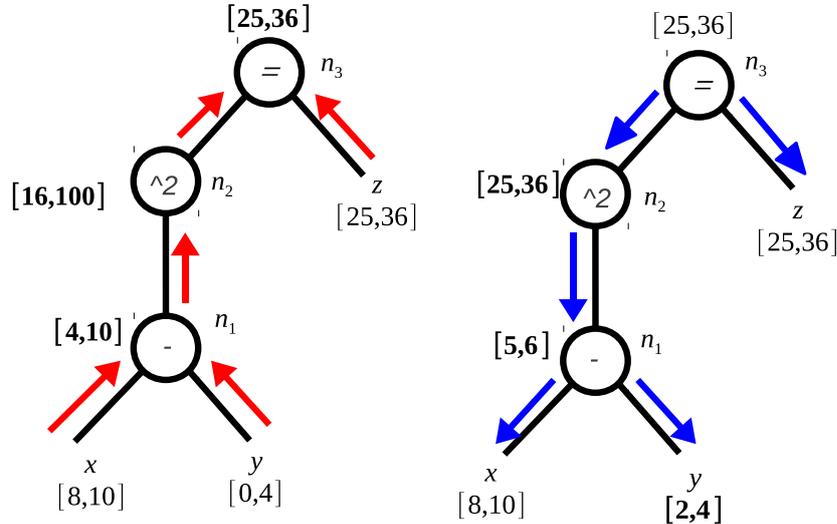


FIGURE 2.7 – Évaluation et phase de projection de la procédure de HC4-Revise à la contrainte $(x - y)^2 = z$ avec $[x] = [8, 10]$, $[y] = [0, 4]$ et $[z] = [25, 36]$.

Un intervalle est associé à chaque nœud. HC4-Revise fonctionne en deux phases :

1. La phase d'évaluation (Forward) est effectuée de manière récursive à partir des feuilles de l'arbre vers la racine (voir figure 2.7). Cette phase calcule, à l'aide des opérateurs d'arithmétique d'intervalles, l'évaluation des sous-expressions naturelles représentées par les nœuds de l'arbre. Considérons par exemple le nœud $n_2 = n_1^2$, avec $n_1 = x - y$, l'évaluation de ce nœud vaut : $[n_1]^2 = [4, 10]^2 = [16, 100]$.
2. La phase de réduction (Backward) parcourt l'arbre à partir de la racine jusqu'aux feuilles (voir figure 2.7) et applique à tous les nœuds l'opérateur de réduction. L'opérateur réduit la largeur des intervalles du nœud en éliminant les valeurs inconsistantes.

Exemple 1 *HC4-revise* appliqué à $(x - y)^2 = z$. Déroulement de la méthode :

La phase d'évaluation \uparrow (Forward) :

1. évaluation du nœud $n_1 = [8, 10] - [0, 4] = [4, 10]$
2. évaluation du nœud $n_2 = [4, 10]^2 = [16, 100]$
3. évaluation du nœud $n_3 = ([16, 100] - [25, 36]) \cap [25, 36] = [25, 36]$

2. Une contrainte primitive est une contrainte binaire ou bien ternaire, pour laquelle chacune des deux ou trois variables n'apparaît qu'une seule fois dans l'expression de la contrainte. Par exemple $x_1 = 3 + x_2 x_3$, $x_1 = 3 \sin(x_2)$, sont deux contraintes primitives

La phase de réduction \downarrow (Backward) :

1. évaluation du noeud $n_3 = [25, 36]$
2. évaluation du noeud $n_2 = [16, 100] \cap [25, 36] = [25, 36]$
3. évaluation du noeud $n_1 = \sqrt{([25, 36])} = [5, 6]$
4. $[y] = y \cap (x - n_1) = [0, 4] \cap ([8, 10] - [5, 6]) = [2, 4]$
5. $[x] = x \cap (n_1 + y) = [8, 10] \cap ([5, 6] + [0, 4]) = [8, 10]$

L'algorithme HC4-reverse est implémenté dans IBEX (Chabert, 2013).

2.3 INVERSION ENSEMBLISTE

On considère le problème de la détermination de l'ensemble solution pour les grandeurs inconnues \mathbf{u} définies par

$$\mathbb{S} = \{\mathbf{u} \in U \mid \Phi(\mathbf{u}) \in [y]\} = \Phi^{-1}([y]) \cap U \quad (2.15)$$

où $[y]$ est connu a priori, U est le domaine de recherche initial pour \mathbf{u} choisi a priori et Φ une fonction non linéaire non nécessairement inversible au sens classique. Le problème (2.15) est un problème d'inversion ensembliste, car on cherche à reconstruire par la fonction Φ un sous-ensemble de l'ensemble de départ U en connaissant l'ensemble d'arrivée $[y]$. Ce problème peut être résolu de manière fiable par SIVIA (Set Inversion Via Interval Analysis). SIVIA (Jaulin and Walter, 1993) est un algorithme récursif qui explore tout l'espace de recherche sans perdre aucune solution. L'algorithme SIVIA Jaulin et al. (2001), basé sur l'arithmétique des intervalles, permet d'obtenir une approximation garantie de cette image réciproque. En effet, en utilisant un test d'inclusion, SIVIA calcule une approximation intérieure $\underline{\mathbb{S}}$ et une approximation extérieure $\overline{\mathbb{S}}$ de l'ensemble \mathbb{S} tel que :

$$\underline{\mathbb{S}} \subseteq \mathbb{S} \subseteq \overline{\mathbb{S}} \quad (2.16)$$

L'approximation (ou encadrement) intérieure $\underline{\mathbb{S}}$ est constituée des pavés acceptables (faisables) : pour prouver qu'une boîte $[\mathbf{u}]$ est acceptable, il suffit de prouver que $\Phi([\mathbf{u}]) \subseteq [y]$; à l'inverse, si on peut prouver que $\Phi([\mathbf{u}]) \cap [y] = \emptyset$ ($\neg \overline{\mathbb{S}}$ est l'ensemble des pavés inacceptables, qui n'est rien d'autre que le complémentaire de $\overline{\mathbb{S}}$), alors le pavé $[\mathbf{u}]$ est inacceptable. Le cas échéant, aucune conclusion ne peut être atteinte et le pavé $[\mathbf{u}]$ est dit ambigu : il est bissecté et les deux sous-pavés testés à nouveau jusqu'à ce que leur taille atteigne un seuil $\epsilon > 0$ fixé par l'utilisateur ; en procédant de la sorte, on assure que l'algorithme SIVIA converge en un nombre fini d'itérations (Jaulin and Walter, 1993). L'approximation extérieure $\overline{\mathbb{S}}$ est définie par

$$\overline{\mathbb{S}} = \underline{\mathbb{S}} \cup \Delta \mathbb{S} \quad (2.17)$$

où $\Delta \mathbb{S}$ est une couche d'incertitude formée par l'union de tous les pavés ambigus dont la taille est en deçà de ϵ .

Algorithme 1 : Algorithme SIVIA.

```

input :  $[\mathbf{u}], \Phi, \epsilon, [y]$ 
output :  $\bar{\mathbb{S}}, -\bar{\mathbb{S}}, \Delta\mathbb{S}$ 
1  $\mathcal{L}_{run} \leftarrow [\mathbf{u}]$ ;
3 while  $\mathcal{L}_{run} \neq \emptyset$  do
4   if  $\Phi([\mathbf{u}]) \cap [y] = \emptyset$  then
5      $-\bar{\mathbb{S}} \leftarrow [\mathbf{u}]$ ;
6   else if  $\Phi([\mathbf{u}]) \subseteq [y]$  then
7      $\bar{\mathbb{S}} \leftarrow [\mathbf{u}]$ ;
8   else if  $w([\mathbf{u}]) \leq \epsilon$  then
9      $\Delta\mathbb{S} \leftarrow [\mathbf{u}]$ 
10  else
11     $([\mathbf{u}_1], [\mathbf{u}_2]) \leftarrow \text{Bissect}([\mathbf{u}])$ ;
12     $\mathcal{L}_{run} \leftarrow [\mathbf{u}_1]$ ;
13     $\mathcal{L}_{run} \leftarrow [\mathbf{u}_2]$ ;
14  end if
15 end while

```

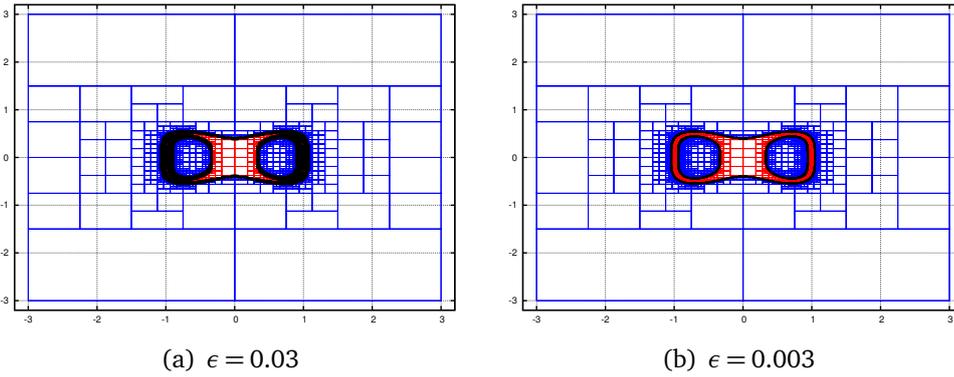


FIGURE 2.8 – Illustration de SIVIA .

Exemple 2 Considérons : $\Phi = u_1^4 - u_1^2 + 4u_2$ où $y = [-0.1, 0.1]$ et $u_1 \times u_2 = ([-3, 3]; [-3, 3])$. Pour obtenir la solution \mathbb{S} , SIVIA peut être utilisé et le résultat est illustré sur la figure 2.8 pour deux valeurs de ϵ . Les régions en noir sont les pavés indéterminés et les régions en rouge l'approximation intérieure.

On peut améliorer les performances de l'algorithme précédent en réduisant le nombre de bisections, ce qui est possible en ajoutant un contracteur avant les tests d'inclusion. Cette nouvelle version est connue sous le nom SIVIA-C. Cela revient à supprimer tout simplement le test de la ligne 5 de l'**algorithme 2** puisque le contracteur retire toutes les parties inconsistantes (voir le tableau comparatif 2.1 entre les deux versions de SIVIA pour un seuil de bisection $\epsilon = 0.003$).

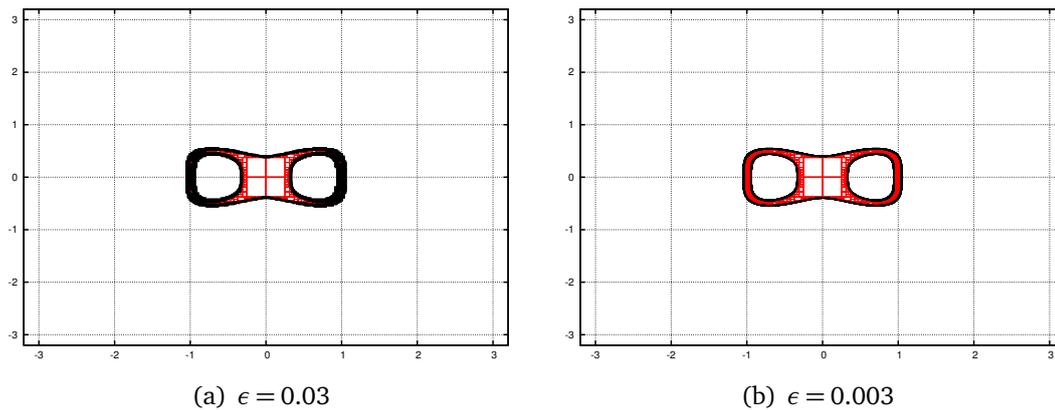


FIGURE 2.9 – Illustration de SIVIA-C .

TABLE 2.1 – Comparaison du nombre de boîtes et du nombre de bisection entre les deux versions de SIVIA ($\epsilon = 0.003$)

	N de bisections	N de boîtes (\underline{S})	N de boîtes (ΔS)	N de boîtes ($-\overline{S}$)
SIVIA	24407	7320	9136	7952
SIVIA-C	17231	7852	6876	0

Algorithme 2 : Algorithme SIVIA-C

```

input :  $[\mathbf{u}], \Phi, \epsilon, [y]$ 
output :  $\underline{S}, -\overline{S}, \Delta S$ 
1  $\mathcal{L}_{run} \leftarrow [\mathbf{u}]$ ;
3 while  $\mathcal{L}_{run} \neq \emptyset$  do
4    $[\mathbf{u}'] = \text{Contractor}(\mathcal{C}, [\mathbf{u}])$ ;
5   if  $\Phi([\mathbf{u}']) \subseteq [y]$  then
6      $\underline{S} \leftarrow [\mathbf{u}']$ ;
7   else if  $w([\mathbf{u}']) \leq \epsilon$  then
8      $\Delta S \leftarrow [\mathbf{u}']$ ;
9   else
10     $([\mathbf{u}_1], [\mathbf{u}_2]) \leftarrow \text{Bisect}([\mathbf{u}'])$ ;
11     $\mathcal{L}_{run} \leftarrow [\mathbf{u}_1]$ ;
12     $\mathcal{L}_{run} \leftarrow [\mathbf{u}_2]$ ;
13  end if
14 end while

```

2.3.1 OPÉRATIONS SUR LES ZONOTOPES

2.3.1.1 ZONOTOPE : DÉFINITIONS ÉLÉMENTAIRES

Les zonotopes constituent une classe particulière de polytopes définis comme l'image d'un hypercube par une application linéaire Lalami (2008). La projection d'un cube de dimension 3 sur un espace de dimension 2 est un zonotope comme le montre la figure 2.10. Le zonotope

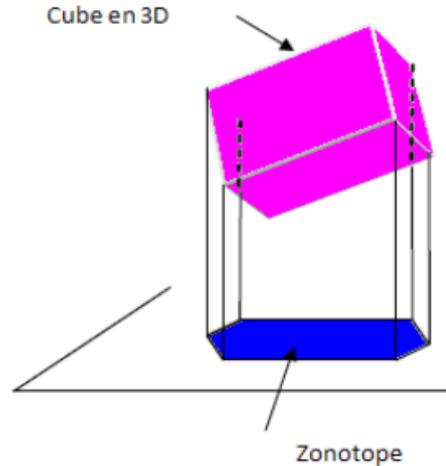


FIGURE 2.10 – Exemple de zonotope.

retrouvé par la projection sur un plan 2D, est représenté par une matrice intervalle 2 lignes (projection dans l'espace 2D) et 3 colonnes (c'est l'image linéaire d'un cube en 3D). Cette propriété décrivant la transformation linéaire par des matrices permet de justifier l'application des zonotopes dans le domaine ensembliste et permet de faire des manipulations matricielles. Les zonotopes sont souvent utilisés pour représenter les domaines atteignables des systèmes dynamiques à entrées bornées. Plus généralement, ils sont utilisés pour caractériser les relations de dépendance linéaire entre les variables incertaines (Lalami, 2008).

2.3.1.2 DEUX DÉFINITIONS ÉQUIVALENTES D'UN ZONOTOPE

Zonotope (Définition 1) – Un p -zonotope $Z \subset \mathbb{R}^n$ est, à une translation près correspondant à son centre $c \in \mathbb{R}^n$, l'image d'un hypercube unitaire de dimension p par une application linéaire. Soit $R \in \mathbb{R}^{n \times p}$ la matrice décrivant cette application linéaire³, on a $Z = c + \blacksquare(R)$ où l'opérateur $\blacksquare(\cdot)$ est défini par :

$$\blacksquare(R) = \{x = R \cdot s \mid s \in \mathbf{B}^p\} \quad (2.18)$$

Somme de Minkowski – La somme de Minkowski de deux ensembles $P \subset \mathbb{R}^n$ et $Q \subset \mathbb{R}^n$ est l'ensemble noté $P \oplus Q$ et défini comme suit :

$$P \oplus Q = \{p + q \mid p \in P, q \in Q\} \quad (2.19)$$

3. En général $n \leq p$, un zonotope pour lequel $n = p$ est un parallélotope

Segment de droite – Un segment de droite $S \subset \mathbb{R}^n$ est l'enveloppe convexe de deux points p_1 et p_2 de \mathbb{R}^n . $c = (p_1 + p_2)/2 \in \mathbb{R}^n$ est le centre de S . $r = (p_2 - p_1)/2 \in \mathbb{R}^n$ est le rayon de S . On a :

$$S = c + r\mathbf{B} \quad (2.20)$$

Un segment de droite est un 1-zonotope et réciproquement.

Zonotope (Définition 2) – Un p -zonotope $Z \subset \mathbb{R}^n$ est la somme de Minkowski de p segments de droite de \mathbb{R}^n .

Exemple 3 Considérons un 4-zonotope Z en 3D dont le centre c et la matrice R donnant la forme du zonotope est donné : Ce zonotope est illustré par la figure 2.11.

$$c = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 2 & 3 & 0.5 & 0 \\ 3 & 2 & 1 & 0 & -0.5 \\ 0.1 & 0.5 & 0.6 & 0.7 & -0.1 \end{bmatrix}$$

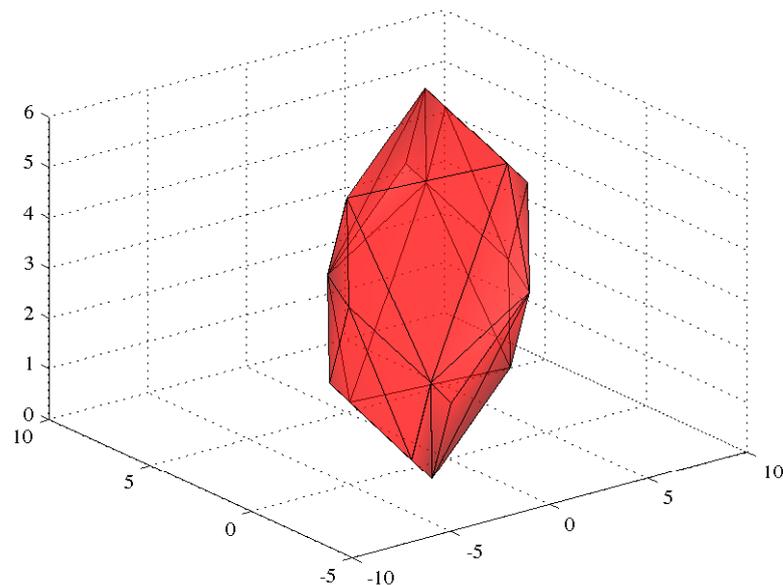


FIGURE 2.11 – Exemple de zonotope obtenu avec la toolbox PNPMP de MATLAB[®] (Riverso et al., 2013).

2.3.1.3 PROPRIÉTÉS DES ZONOTOPES

Théorème 1 (Inclusion d'une famille de zonotopes (Alamo et al., 2005; Raimondo et al., 2011))
 Considérons une famille de zonotopes représentée par $Z = c \oplus [M]\mathbf{B}^p$, où $c \in \mathbb{R}^n$, et $[M] \in \mathbb{R}^{n \times p}$ est

une matrice intervalle. Le zonotope suivant

$$Z \subseteq \diamond(Z) = c \oplus J\mathbf{B}^{p+n},$$

est une approximation extérieure de la famille de zonotopes, où la matrice $J \in \mathbb{R}^{n \times (n+p)}$ est définie par

$$J = [\text{mid}([M]) | G]. \quad (2.21)$$

$[\cdot | \cdot]$ dénote la concaténation classique de matrices et $G \in \mathbb{R}^{n \times n}$ est une matrice diagonale qui satisfait

$$G_{ii} = \sum_{j=1}^m \text{rad}([M]_{ij}), \quad i = 1, \dots, n. \quad (2.22)$$

Théorème 2 (Extension d'un zonotope (Alamo et al., 2005; Kühn, 1998; Raimondo et al., 2011))

Considérons une fonction $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ avec des dérivées continues, un zonotope $Z = c \oplus R\mathbf{B}^p$, et une matrice intervalle $[M] \in \mathbb{I}\mathbb{R}^{n \times p}$. Nous avons

$$\nabla F(Z)R \subseteq [M] \Rightarrow F(Z) \subseteq F(c) \oplus \diamond([M]\mathbf{B}^p). \quad (2.23)$$

ici l'opérateur ∇ désigne la jacobienne.

2.4 INTÉGRATION NUMÉRIQUE GARANTIE D'ÉQUATIONS DIFFÉRENTIELLES : TUBE DE TRAJECTOIRES

Considérons le système dynamique *incertain* décrit par (2.4) avec $z(t_0) \in \mathcal{X}_0$ à l'instant $t_0 \geq 0$ et $\mathcal{X}(t; t_0, \mathcal{X}_0)$ l'ensemble des solutions de (2.4) à l'instant t provenant de chaque condition initiale dans \mathcal{X}_0 à t_0 . On considère une grille temporelle $t_0 < t_1 < t_2 < \dots < t_n$, avec un pas de temps non nécessairement constant, et on suppose $\mathcal{X}_0 = [z_0] = [\underline{z}_0, \bar{z}_0]$. L'intégration numérique garantie par les méthodes de Taylor intervalle calcule des vecteurs intervalle $[z_j], j = 1, \dots, n$ qui contiennent de manière garantie l'ensemble $\mathcal{X}(t_j; t_0, \mathcal{X}_0)$ des solutions de (2.4) à l'instant t_j , $j = 1, \dots, n_T$.

Ces méthodes se déroulent généralement en deux étapes :

- Existence et Unicité : cette étape consiste à vérifier l'existence et l'unicité de la solution pour chaque condition initiale $z_j \in [z_j]$ en utilisant le théorème du point fixe de Banach et l'opérateur de Picard-Lindelöf ((Moore, 1996), (Nedialkov et al., 1999)) et à calculer la solution a priori $[\tilde{z}_j]$ qui vérifie $[\tilde{z}_j] \supseteq \mathcal{X}(t)$ pour tout t dans $[t_j, t_{j+1}]$ ((Nedialkov, 1999a) ; (Berz et al., 1998)).
- Contraction : cette étape consiste à fournir un encadrement garanti de l'ensemble solution $[z_{j+1}]$ de (2.4) à t_{j+1} en utilisant soit un développement en série de Taylor à l'ordre k ou des contracteurs à point fixe (Hansen and Walster, 2004) et la solution à t_j . $[\tilde{z}_j]$ est utilisée pour encadrer le reste du développement.

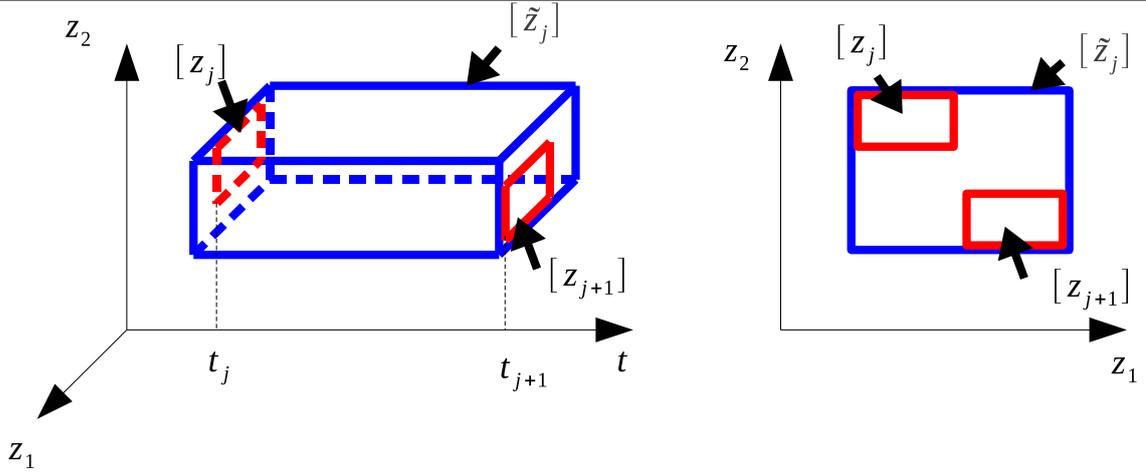


FIGURE 2.12 – Illustration du tube de trajectoires (boîte en bleu) entre deux pas de temps t_j, t_{j+1} ; $[\tilde{z}_j]$ représente la solution a priori, $[z_j]$ représente la solution à l'instant t_j , et $[z_{j+1}]$ représente la solution à l'instant t_{j+1} .

Les auteurs (Eijgenraam, 1981; Lohner, 1987b; Moore, 1996; Nedialkov et al., 1999; Rihm, 1994a, 1992, 1993, 1994b) ont proposé plusieurs algorithmes et méthodes pour rendre la méthode de Taylor applicable en pratique, car il y a bien évidemment des problèmes de sur-approximation.

(Ramdani and Nedialkov, 2011) ont fait une extension de la méthode Taylor afin d'avoir une expression analytique de la solution du tube de trajectoires (2.24) sur tout l'horizon temporel $[t_j, t_{j+1}]$. A partir d'un point courant t_j de la grille temporelle, on a la solution $[z_j]$, on peut calculer la solution garantie $[z_{j+1}]$ (voir figure 2.12) au pas de temps suivant t_{j+1} , en utilisant le développement en série de Taylor (2.24) dans lequel le reste ($[\tilde{z}_j]$) est encadré dans une boîte contenant de manière garantie toute la trajectoire entre le début et la fin du pas d'intégration $[t_j, t_{j+1}]$.

La solution analytique du tube de trajectoires pour tout $t \in [t_j, t_{j+1}]$ est donc donnée par :

$$\mathcal{X}(t; t_j, z_j) \subseteq [z](t; t_j, [z_j]) = [z_j] + \sum_{i=1}^{k-1} (t-t_j)^i f_q^{[i]}([z_j]) + (t-t_j)^k f_q^{[k]}([\tilde{z}_j]), \quad (2.24)$$

L'équation (2.24) est une expression analytique de la frontière du tube de trajectoires de (2.4). Les coefficients $f_q^{[i]}$ de Taylor sont obtenus par différentiation automatique (Bendtsen and Stauning, 1996).

$$\begin{cases} f_q^{[1]} = f \\ f_q^{[i]} = \frac{1}{i} \frac{\partial f_q^{[i-1]}}{\partial z} f \end{cases} \quad (2.25)$$

Algorithme 3 : Algorithme φ^{QR} .

input : $[z_j], t_j, t, [\tilde{z}_j]$
output : $[x](t)$

- 1 $[v](t) := \hat{z}_j + \sum_{i=1}^{k-1} (t-t_j)^i f_q^{[i]}(\hat{z}_j) + (t-t_j)^k f_q^{[k]}([\tilde{z}_j]);$
- 2 $[S](t) := \mathbb{I} + \sum_{i=1}^{k-1} (t-t_j)^i \frac{\partial f_q^{[i]}}{\partial z}([z_j]);$
- 3 $[q](t) := ([S](t)A_j)[r_j] + [S](t)([v_j] - \hat{z}_j);$
- 4 $[z](t) := [v](t) + [q](t);$
- 5 obtenir $A(t)$ via QR-factorization of $\text{mid}([S](t)A_j)$ (Lohner, 1987a) (voir **algorithme 4**);
- 6 $[r](t) := A(t)^{-1}([S](t)A_j)r_j + (A(t)^{-1}[S])([v_j] - \hat{z}_j);$
- 7 $\hat{z}(t) := \text{mid}([v](t));$
- 8 $[x](t) := \{[z](t), \hat{z}(t), [v](t), [r](t), A(t)\};$

L'encadrement a priori $[\tilde{z}_j]$ est utilisé afin d'évaluer l'erreur de troncature du développement en série de Taylor. Ce dernier terme étant implicitement proportionnel à $1/k!$, le pessimisme induit peut être contrôlé en choisissant k grand.

Le pessimisme introduit par le développement de Taylor (2.24) peut être évalué en calculant la taille w du pavé $[z_{j+1}]$. La taille d'un pavé est toujours positive, il est montré que $w([z_{j+1}]) > w([z_j])$ (Raïssi, 2012a).

Ainsi, le schéma numérique (2.24) est numériquement instable et diverge au bout de quelques pas, il ne peut donc pas être utilisé directement.

Les méthodes pratiques utilisent la forme centrée, un pré-conditionnement matriciel et des transformations linéaires (Ramdani and Nedialkov, 2011; Ramdani, 2012). Dans la suite, on pourra utiliser la représentation suivante pour contrôler le pessimisme. A chaque instant t_j , l'encadrement de la solution est calculé sous la forme :

$$\mathcal{X}(t_j; t_0, z_0) \in \{v_j + A_j r_j \mid v_j \in [v_j], r_j \in [r_j]\},$$

où la matrice A_j est généralement obtenue par factorisation QR (voir la section 2.4.1 pour plus de détails).

Soit la forme composée (ligne 8 de l'**algorithme 3**)

$$[x](t) \equiv \{[z](t), \hat{z}(t), [v](t), [r](t), A(t)\}, \quad (2.26)$$

où $\hat{z}(t) := \text{mid}([z](t))$

Nous rappelons la méthode de Lohner que nous avons implémentée pour contrôler l'effet d'enveloppement dans les méthodes de Taylor intervalle, dans l'algorithme d'atteignabilité hybride non linéaire initialement publié dans (Ramdani and Nedialkov, 2011).

2.4.1 MÉTHODE DE LOHNER : FACTORISATION QR

Le principe fondamental de la méthode développée par Lohner (Lohner, 1987b; Nedialkov et al., 1999) pour contrôler l'effet d'enveloppement à chaque pas d'intégration, consiste à effectuer un changement de coordonnées du système pour contrôler la sur-approximation de la vraie solution. L'idée exploitée est de propager dans le processus d'intégration des parallélépipèdes, qui encadrent au mieux l'ensemble solution à chaque pas d'intégration, au lieu de pavés alignés avec les axes du système original. Il s'agit donc de définir, à chaque pas d'intégration, une matrice inversible A_j et un vecteur intervalle $[r_j]$ tels que la boîte (2.24), solution de (2.4) puisse être caractérisée par :

$$[z_j] = \{v_j + A_j r_j \mid r_j \in [r_j], v_j \in [v_j]\} \quad (2.27)$$

Comme mentionné dans (Ramdani and Nedialkov, 2011), il est possible d'obtenir une solution explicite pour la solution de (2.4) pour tout $t \in [t_j, t_{j+1}]$, en combinant la méthode Lohner et une méthode de Taylor intervalle, comme indiquée dans l'**algorithme** φ^{QR} (**algorithme 3**, page 32), où $[\chi]$ désigne les variables $\{[z], \hat{z} = \text{mid}([z]), [v], [r], A(t)\}$, et où $[\tilde{z}_j]$ désigne la solution a priori encadrant le tube de trajectoire sur tout l'intervalle $[t_j; t_{j+1}]$. Ainsi l'encadrement de la solution à un instant $t \in [t_j, t_{j+1}]$ est donné par

$$[\chi](t) = \varphi^{QR}([\chi_j], t_j, t, [\tilde{z}_j]).$$

et l'encadrement de la solution à l'instant t_{j+1} par

$$[\chi_{j+1}] = \varphi^{QR}([\chi_j], t_j, t_{j+1}, [\tilde{z}_j]).$$

Dans l'**algorithme** $\varphi^{QR}(\cdot)$, on utilise la méthode de Lohner pour choisir la matrice $A(t)$ (ligne 5 de l'**algorithme 3**). Ce choix est fondé sur la factorisation QR, qui joue un rôle essentiel dans le contrôle de l'effet d'enveloppement en choisissant une transformation de coordonnées appropriée. Il s'agit d'effectuer une rotation des axes du repère de façon à ce que la première composante de Q soit parallèle au plus long des cotés du parallélépipède $A[r]$ (où $[r]$ est calculé ligne 6 de l'**algorithme 3**). Lohner a montré que cette opération réduit la sur-approximation de la solution d'une manière significative. En pratique, nous calculons les longueurs de chaque coté du parallélépipède $A[r]$ par la formule :

$$l_i = \|A_{j+1}^i\|_2 \cdot w([r_j]^i), \quad (2.28)$$

où A_{j+1}^i est la i ème colonne de A , $[r_j]^i$ la i ème composante de r_j , $w(\cdot)$ la largeur d'un intervalle, et $\|\cdot\|_2$ représente la norme Euclidienne. La méthode de Lohner est résumée dans l'**algorithme 4** suivant : on pose $\hat{A}_{j+1} = \tilde{A}_{j+1} \cdot P_{j+1}$, avec P_{j+1} une matrice de permutation. Nous avons dans le cas général :

$$\hat{A} \in \{S_j \cdot A_j \mid S_j \in [S_j]\} \quad (2.29)$$

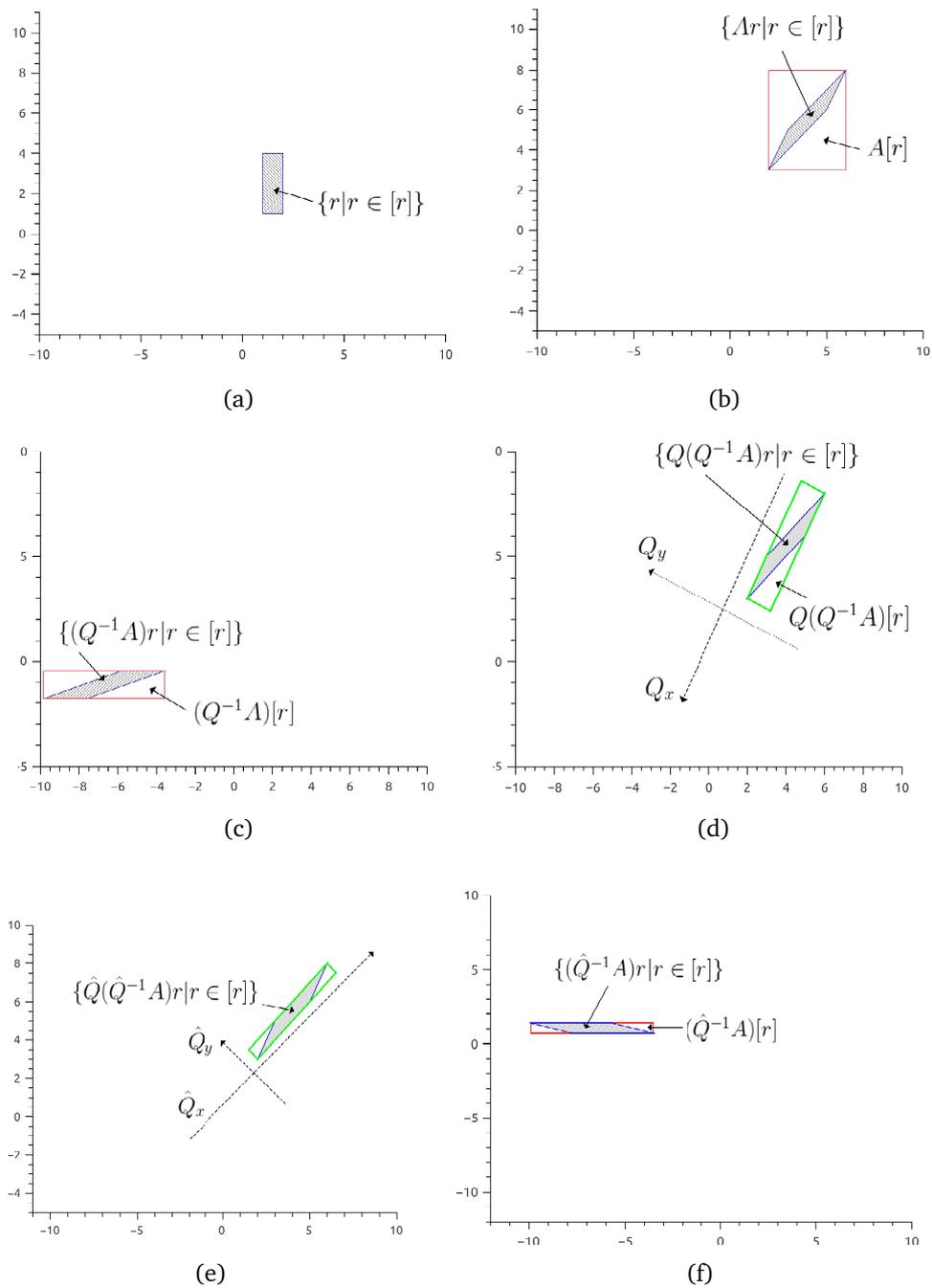


FIGURE 2.13 – Illustration de la factorisation QR.

Dans ce mémoire, nous prenons (voir ligne 5 de l'**algorithme** φ^{QR}) :

$$\tilde{A}_{j+1} \in \text{mid}(S_j \cdot A_j), \text{ avec } S_j \in [S_j].$$

Pour illustrer la méthode de Lohner, considérons l'exemple extrait de la thèse de Nedialkov

Algorithme 4 : Algorithme l_{qr} Factorisation QR.

input : A_j, S_j, r_j
output : $A(t), l_i$

- 1 $A(t) := mid((S_j)A_j)$;
- 2 Compute l_i via (2.28);
- 3 Sort(l_i) and $A_{perm}(t) := Permute(A(t))$;
- 4 Compute $Q(t) \leftarrow A_{perm}(t) = Q(t)R(t)$;
- 5 Choose $A(t) := Q(t)$;

(Nedialkov et al., 1999) et que nous avons reproduit. Dans cet exemple $A = \tilde{A}$, soit la matrice A , et le vecteur intervalles $[r]$

$$A = \begin{pmatrix} 1 & 1 \\ 2 & 1 \end{pmatrix} \quad [r] = \begin{pmatrix} [1, 2] \\ [2, 4] \end{pmatrix}$$

La factorisation QR de la matrice A est

$$A = \left(-\frac{1}{\sqrt{5}} \begin{pmatrix} 1 & 1 \\ 2 & -1 \end{pmatrix}\right) \left(-\frac{1}{\sqrt{5}} \begin{pmatrix} 5 & 3 \\ 0 & 1 \end{pmatrix}\right) = QR \quad (2.30)$$

Considérons tout d'abord l'ensemble

$$\{Ar \mid r \in [r]\} \quad (2.31)$$

La figure 2.13-a montre la représentation graphique du pavé initial $[r]$ dans le plan (x, y) . Une application linéaire, caractérisée par la matrice A , le transforme en un parallélogramme tracé dans la figure 2.13-b. Sur cette figure, la boîte englobante (en ligne rouge) représente une sur-approximation de ce parallélogramme obtenue par un simple calcul intervalle $A[r]$. En revanche, un changement orthogonal de coordonnées, engendré par la matrice Q , permet d'améliorer nettement la précision de l'encadrement du parallélogramme $\{Ar \mid r \in [r]\}$ dans cette nouvelle base de coordonnées (Q_x, Q_y) comme le montre la figure 2.13-d.

Maintenant, considérons l'ensemble

$$\{Q^{-1}Ar \mid r \in [r]\} \quad (2.32)$$

La matrice Q^{-1} , calculée à partir de (2.30), transforme le parallélogramme (2.31) en un autre parallélogramme (2.32) dont le côté court est parallèle à l'axe original x . Son encadrement est $(Q^{-1}A)[r]$ représenté sur la figure 2.13-c.

Maintenant en calculant les longueurs de chaque coté du parallélogramme suivant la formule (2.28), nous avons :

$$l_1 = \sqrt{(1^2 + 2^2)} \cdot (2 - 1); \quad l_2 = \sqrt{(1^2 + 1^2)} \cdot (4 - 1)$$

Nous avons $l_2 > l_1$, ainsi, selon le critère de Lohner, on permute les colonnes 1 et 2 de la matrice \tilde{A} , en utilisant la matrice de permutation P ,

$$P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \Rightarrow \hat{A} = \begin{pmatrix} 1 & 1 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$$

La factorisation QR de la nouvelle matrice \hat{A} est

$$\hat{A} = \left(-\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\right) \left(-\frac{1}{\sqrt{2}} \begin{pmatrix} 2 & 3 \\ 0 & -1 \end{pmatrix}\right) = \hat{Q}\hat{R}$$

La figure 2.13-e montre que l'encadrement de l'ensemble (2.31) dans le plan $(\hat{Q}_x^{-1}, \hat{Q}_y^{-1})$ est plus précis que celui obtenu dans le plan (Q_x^{-1}, Q_y^{-1}) . De même, le pavé $(\hat{Q}^{-1}A)[r]$ est de taille inférieure par rapport à celle du pavé $(Q^{-1}A)[r]$ comme l'illustre la figure 2.13-f. En effet, le côté le plus long du parallélépipède, $\{Ar | r \in [r]\}$ transformé par Q^{-1} , est parallèle à l'axe original x .

2.4.2 REPRÉSENTATION ZONOTOPIQUE ET DOMAINE DE SOLUTION

Nous pouvons remarquer que le domaine de solution (2.27) est la somme de Minkowski d'un parallélotope, c'est à dire une boîte orientée, et d'une boîte alignée avec les axes, que nous désignons par MSBP (Minkowski Sum of a Parallelotope and an axis aligned Box).

$$\mathcal{X}(t) = A(t)[r](t) \oplus [v](t). \quad (2.33)$$

Une MSBP est un zonotope particulier généré par $2n$ segments de droite (voir la figure 2.14) :

$$\mathcal{X}(t) = c(t) \oplus R(t)\mathbf{B}^{2n}, \quad (2.34)$$

où, pour tout t , le vecteur ponctuel $c(t) \in \mathbb{R}^n$ et la matrice ponctuelle $R(t) \in \mathbb{R}^{n \times 2n}$ satisfont :

$$c(t) = A(t)\text{mid}([r](t)) + \text{mid}([v](t)), \quad (2.35)$$

$$R(t) = [A(t)\text{dr}([r](t)) | \text{dr}([v](t))]. \quad (2.36)$$

où $|$ dénote la concaténation de matrices, et $\text{dr}(\cdot)$ est la notation simplifiée de $\text{diag}(\text{rad}(\cdot))$, c'est à dire une matrice diagonale de nombres réels dont chaque élément correspond au rayon d'un intervalle.

Il est donc évident de voir que, pour tout t :

$$\mathcal{X}(t) = \{A(t)r + v | v \in [v](t), r \in [r](t)\} = A(t)[r](t) \oplus [v](t).$$

Remarque 1 Le domaine de solution est un zonotope particulier (MSBP), et notre méthode de fusion de trajectoires est basée sur une approche zonotopique sur laquelle nous reviendrons plus en détail dans la section 4.4.

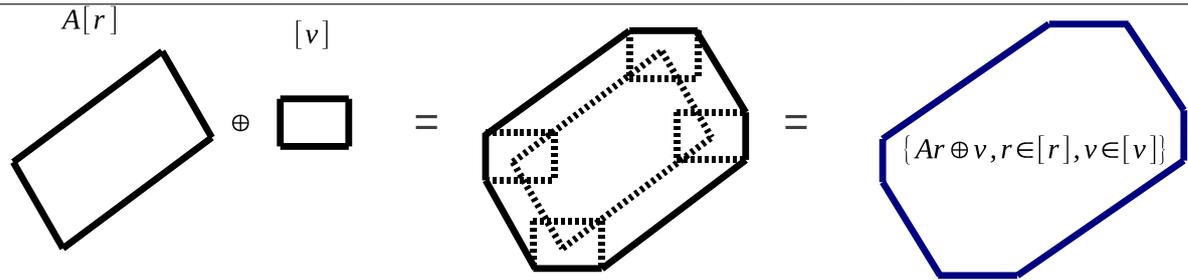


FIGURE 2.14 – Somme de Minkowski du paralléloptope $A[r]$ et de la boîte alignée avec les axes $[v]$.

CONCLUSION

Ce chapitre a été consacré à la présentation des différentes problématiques abordées dans la thèse : l'atteignabilité hybride non linéaire, l'estimation de paramètres, l'observation d'état hybride. Il rappelle ensuite les outils mathématiques utilisés : l'arithmétique des intervalles et les zonotopes, ainsi que les méthodes d'intégration numérique garantie des équations différentielles ordinaires incertaines et les méthodes d'inversion ensembliste. Pour l'intégration numérique nous avons présenté en premier lieu le principe de base du développement en série de Taylor intervalle, puis nous avons introduit les techniques de factorisation QR de Lohner qui permettent de contrôler la propagation du pessimisme dû à l'effet d'enveloppement.

CHAPITRE 3

ATTEIGNABILITÉ HYBRIDE. ÉTAT DE L'ART

Sommaire

3.1 Introduction	39
3.2 Atteignabilité continue	40
3.3 Transition discrète	42
3.4 Cas où la garde est linéaire	43
3.4.1 Méthodes par approximation d'ensembles pour l'atteignabilité hybride linéaire	43
3.5 Cas où la garde est non linéaire	47
3.5.1 Méthode par approximation polynomiale de la garde	48
3.5.2 Méthodes par intervalles	49
3.6 Conclusion	50

3.1 INTRODUCTION

L'analyse d'atteignabilité est un défi majeur rencontré dans de nombreux problèmes de recherche, citons par exemple la commande prédictive (Bravo et al., 2006; Gonzalez et al., 2011; Kerrigan et al., 2002; Limon et al., 2005; Rakovic et al., 2012), la commande non linéaire ou le contrôle optimal (Lygeros, 2004; Lygeros et al., 1998a, 1999; Scott and Barton, 2010a,b, 2013), la théorie des jeux (de Alfaro et al., 2007; Hansen et al., 2010; Tomlin et al., 2000), la théorie de la viabilité (Gao et al., 2004, 2006), et l'estimation (Alamo et al., 2005; Benazera and Travé-Massuyès, 2009; Chernousko, 2005; Hofbaur and Williams, 2004; Le et al., 2013; Meslem et al., 2010b). Pour les systèmes hybrides (Asarin et al., 2006), elle est utilisée pour la vérification (Alur et al., 1993; Aswani et al., 2012; Bhatia and Frazzoli, 2004; Bouissou et al., 2014; Clarke et al., 2003; Guéguen et al., 2009; Guéguen and Zaytoon, 2004; Tomlin et al., 2003; Tomlin, 2011; Yazarael and Pappas, 2004) et les tâches de synthèse pour les systèmes embarqués (Alur et al., 2003; Donzé et al., 2009; Lygeros et al., 1998b). L'atteignabilité hybride

articule l'atteignabilité continue dans les différents modes, et les transitions discrètes c'est-à-dire, le calcul de l'intersection entre le tube de trajectoires et la condition de garde. Nous allons donc étudier les différentes méthodes présentes dans la littérature pour l'atteignabilité continue et les transitions discrètes.

3.2 ATTEIGNABILITÉ CONTINUE

Nous étudierons les méthodes de la littérature permettant de résoudre le problème de valeur initiale (PVI) des équations différentielles ordinaires (EDO). Ceci revient à calculer l'ensemble atteignable par un système dynamique incertain non autonome décrit par une équation différentielle non linéaire de la forme suivante :

$$\dot{x}(t) = f(x, p, t), x(t_0) \in [\underline{x}_0, \bar{x}_0] \subseteq \mathcal{D}, p \in [\underline{p}, \bar{p}] \quad (3.1)$$

où la fonction $f : \mathcal{D} \times [\underline{p}, \bar{p}] \times \mathbb{R}^+ \mapsto \mathcal{D}$, $\mathcal{D} \subseteq \mathbb{R}^n$, $[\underline{x}_0, \bar{x}_0]$ est le domaine initial pour le vecteur d'état $x(t_0)$ à l'instant $t_0 \geq 0$ et $[\underline{p}, \bar{p}]$ est le domaine d'incertitude du vecteur de paramètres p . On désigne par $\mathcal{X}(t)$ l'ensemble des solutions de (3.1) à l'instant t obtenues pour toutes les conditions initiales prises dans $[\underline{x}_0, \bar{x}_0]$ et pour tout vecteur de paramètres pris dans $[\underline{p}, \bar{p}]$. L'ensemble atteignable du système (3.1) sur un horizon temporel $[t_0, t]$ est défini comme suit :

$$\begin{aligned} \mathcal{R}([t_0, t]; [\bar{x}_0]) &\equiv \left\{ x(\tau), t_0 \leq \tau \leq t \mid (\dot{x}(t) = f(x, p, t) \wedge x(t_0) \in [\underline{x}_0, \bar{x}_0] \wedge p \in [\underline{p}, \bar{p}]) \right\} \\ &= \bigcup_{\tau \in [t_0, t]} \mathbb{X}(\tau) \quad (3.2) \end{aligned}$$

En pratique, le calcul exact de l'ensemble atteignable par un système dynamique continu non linéaire est impossible, sauf pour le cas des systèmes dont le modèle continu est défini par $\dot{x}(t) = c$ et où les composantes du vecteur c sont des constantes réelles ou intervalle. Dans la littérature il existe deux types d'approches pour résoudre ce problème.

La première approche repose sur des méthodes de calcul ensembliste qui consistent à fournir une sur-approximation de l'espace d'état atteignable sous la forme de l'union d'ensemble ayant une même forme géométrique. Le principe fondamental de cette approche consiste à faire évoluer dans le temps un domaine d'état initial selon la dynamique du système continu. Parmi les formes géométriques qui ont déjà été utilisées dans ce contexte, il y a les parallélotopes (Büeller et al., 2000), les polytopes (Chutinan and Krogh, 2003; Hagemann, 2014), les ellipsoïdes (Chernousko, 2005; Kurzhanski and Varaiya, 2000), les zonotopes (Althoff and Krogh, 2011, 2012; Althoff et al., 2008; Girard, 2005), les rectangles orientés (Stursberg and Krogh, 2003a), les fonctions supports (Asarin et al., 2006; Frehse, 2005; Frehse and Ray, 2012; Girard and Le Guernic, 2008; Guernic and Girard, 2009b) et les vecteurs intervalles (Henzinger and Ho, 1995; Ramdani and Nedialkov, 2011). En pratique, les six premières formes ne s'appliquent que pour le cas des systèmes continus linéaires en absence d'incertitudes paramétriques. Pour les systèmes linéaires avec des entrées incertaines, les zonotopes (Girard, 2005; Girard et al.,

2006; Kühn, 1998) et les fonctions supports (Girard and Le Guernic, 2008) ont clairement montré leur efficacité par rapport aux autres méthodes, permettant la vérification des systèmes avec plus de 100 variables d'état continues (Althoff and Krogh, 2012). Cependant, la forme «vecteur intervalle», encore appelée «boîte», s'applique pour le cas des systèmes non linéaires avec incertitudes paramétriques en utilisant le développement en série de Taylor étendu aux intervalles (Moore, 1996), (Nedialkov, 1999a). Mais dans ce cas, la précision de l'approximation est conditionnée par la taille du domaine d'état initial et du domaine admissible des paramètres incertains. En effet, dans le cas général cette approximation diverge sous l'effet de l'accumulation du pessimisme dû au phénomène d'enveloppement (Moore, 1996), (Nedialkov, 1999a), (Anguelov and Markov, 1998) (problème classique d'inclusion d'un ensemble de forme quelconque dans une boîte alignée avec les axes).

Récemment, l'analyse par intervalles (Moore, 1996; Sunaga, 2009) et l'intégration numérique garantie des équations différentielles ordinaires (EDO) basée sur des modèles de Taylor intervalles (Lohner, 1987b; Nedialkov, 1999a) que nous avons présentée au chapitre 2, section 2.4 ont aussi été utilisées pour résoudre le problème d'atteignabilité des systèmes dynamiques hybrides complexes ou fortement non linéaires (Henzinger et al., 2000b; Rauh et al., 2006). Mais la surestimation de l'espace d'état atteignable obtenu par ces méthodes, engendrée par le phénomène d'enveloppement limite leur exploitation pour résoudre certains cas réels. Cependant, l'intégration ensembliste non linéaire garantie peut être traitée par l'application de théorèmes de comparaison issus du domaine des inégalités différentielles. Il a été montré dans des travaux (voir par exemple (Kieffer et al., 2006; Müller, 1927)) que cette limitation peut être évitée pour certains modèles en utilisant les théorèmes de comparaison (Ramdani et al., 2008). L'approche basée sur les théorèmes de comparaison permet de déterminer un encadrement de l'ensemble des trajectoires admissibles du système incertain (3.1). L'idée principale est de construire des modèles déterministes permettant d'encadrer les solutions de (3.1) (on parle de système englobant). Cette approche permet de s'affranchir du phénomène d'enveloppement dû à la manipulation d'intervalles dans le cas multidimensionnel.

La seconde approche consiste à partitionner l'espace d'état en plusieurs parties de mêmes propriétés vis-à-vis de la dynamique continue, puis à approcher localement la dynamique complexe par des dynamiques plus simples faciles à englober (Althoff et al., 2008; Asarin et al., 2006; Lefebvre and Guéguen, 2006). Cependant, l'application de ces méthodes est souvent limitée à cause de la dimension élevée et de la complexité des modèles représentatifs des systèmes réels. Par analogie le calcul avec des automates linéaires hybrides, les algorithmes linéaires pour systèmes continus peuvent être appliqués à des systèmes non linéaires classiques ou hybrides. Ceci est réalisé par linéarisation conservative, c'est-à-dire en tenant compte de l'erreur de linéarisation comme d'une entrée supplémentaire incertaine du système linéarisé (Althoff et al., 2008). Cette méthode est également appelée hybridation puisqu'on obtient un système hybride, où chaque région de linéarisation est soumise à une dynamique continue différente. Une des limites est l'effort en temps de calcul pour les transitions entre les régions de l'espace d'état. Pour calculer les sur-approximations de l'ensemble atteignable en utilisant les méthodes d'intégration numérique garantie des équations différentielles ordinaires (EDO) basées sur des modèles de

Taylor intervalles (Lohner, 1987b; Nedialkov, 1999a). Pour cela, l'ensemble des états initiaux doit être partitionné et les états accessibles sont calculés à partir de chaque sous-ensemble de la partition (Ramdani and Nedialkov, 2011).

Des outils logiciels existent, dont certains en *open source*, permettant de résoudre le PVI en utilisant l'arithmétique des intervalles ont été réalisés. Par exemple, les méthodes qui s'appuient sur le développement en série de Taylor par rapport au temps (Lohner, 1987a; Moore, 1996; Nedialkov, 1999b; Rihm, 1994a, 1998). Dans cette classe, plusieurs logiciels qui implémentent ces méthodes sont disponibles : AWA (Lohner, 1988), VNODE-LP (Nedialkov, 2006,?, 1999b). Il existe aussi des méthodes qui reposent sur le développement en série de Taylor par rapport au temps et aux variables (Berz and Makino, 1998a, 2006; Hoefkens et al., 2003) ; dans cette catégorie de méthodes on trouve COSY INFINITY (Berz, 2002), VSPODE (Lin and Stadtherr, 2007) et l'outil d'atteignabilité hybride non linéaire Flow* (Chen, 2013; Chen et al., 2012b, 2013). Nous comparons dans le chapitre 4 les performances de notre approche d'atteignabilité hybride non linéaire avec l'outil Flow* sur des benchmarks. Nous pouvons aussi citer le solveur IOLAVABE (Eggers et al., 2012, 2014) dans lequel, les systèmes englobants sont construits automatiquement à la volée. Pour accélérer et simplifier le re-calcul du système englobant ou les bornes de la solution du PVI pour les sous-domaines arbitraires de chaque pas de temps d'intégration sans réinitialisation coûteuse du solveur, IOLAVABE stocke aussi et réutilise les coefficients de Taylor calculés par VNODE-LP.

3.3 TRANSITION DISCRÈTE

Le passage ensembliste des gardes consiste à transférer un vecteur d'état continu incertain (caractérisé par un ensemble) d'un mode à un autre, lorsque les conditions de garde sont déclenchées par l'évolution de ce même vecteur d'état. Le tube de trajectoires ayant une taille non nulle, quand celui-ci rencontre une condition de garde décrite par des contraintes (linéaires ou non linéaires), il existe un ensemble non vide d'instantants pendant lesquels la condition de garde est vérifiée, que l'on appellera *continuum* d'instantants de commutation.

Nous décomposons le problème du passage ensembliste de gardes en trois tâches :

- T1 : Détection de la satisfaction de la condition de garde ;
- T2 : Calcul du sous-ensemble d'état intersectant la garde ;
- T3 : Calcul de l'image du sous-ensemble par la fonction de saut.

Nous ne nous intéresserons dans cette section qu'aux deux premières tâches, la dernière tâche sera plus détaillée dans la section 4.3. Il existe peu de travaux à notre connaissance qui s'intéressent à la résolution du problème d'intersection du tube de trajectoires décrit par des équations différentielles non linéaires avec une garde qui est décrite par des fonctions non linéaires. La majorité des travaux se focalisent sur les classes des systèmes hybrides dont la dynamique continue est décrite par des équations différentielles linéaires. Plusieurs algorithmes ont été développés pour détecter et localiser des événements. Ces algorithmes varient dans leur approche de détection des événements et dans la phase de localisation. Nous distinguerons

principalement dans ce mémoire :

1. Les méthodes qui utilisent le fort potentiel offert par les approximations d'ensembles par les polytopes (Girard, 2005), les zonotopes ((Girard and Guernic, 2008), (Althoff and Krogh, 2011, 2012; Althoff et al., 2008)) et les fonctions supports ((Asarin et al., 2006; Frehse, 2005; Frehse and Ray, 2012; Girard and Le Guernic, 2008; Guernic and Girard, 2009b)),
2. Les méthodes qui utilisent les notions d'approximation polynomiale de la fonction de garde ((Shampine et al., 1987),(Birta et al., 1985),(Park and Barton, 1996), (Esposito and Kumar, 2007; Esposito et al., 2001), (Girard, 2002)) combinées avec les méthodes de recherche de zéros d'un polynôme,
3. Les méthodes utilisant les notions de propagation de contraintes sur les intervalles (CSP¹ (Ramdani and Nedialkov, 2011), HCS² (Ishii et al., 2009)).

Dans les deux premières méthodes, les gardes sont linéaires (hyperplan, demi-espace), et la dernière méthode aborde les gardes non linéaires, mais reste applicable aux gardes linéaires.

3.4 CAS OÙ LA GARDE EST LINÉAIRE

3.4.1 MÉTHODES PAR APPROXIMATION D'ENSEMBLES POUR L'ATTEIGNABILITÉ HYBRIDE LINÉAIRE

3.4.1.1 ENSEMBLE ATTEIGNABLE REPRÉSENTÉ PAR DES ZONOTOPES

Considérons un système linéaire hybride où la dynamique continue est décrite par des équations différentielles incertaines (Girard, 2005; Girard and Guernic, 2008). La garde est décrite par un hyperplan, la détection de l'événement est réalisée en vérifiant à chaque étape si l'ensemble atteignable intersecte la garde ou non (équation 3.3). La garde est définie par :

$$G = \{x \in \mathbb{R}^n : d^T \cdot x = e\}$$

Soit $Z = (c, \langle g_1, \dots, g_p \rangle)$ un zonotope et G un hyperplan, l'intersection de Z et G est non vide ssi :

$$\forall x_1 \in [-1, 1], \dots, x_p \in [-1, 1], d^T c + \sum_{i=1}^{i=p} d^T g_i x_i = e$$

Ainsi, le zonotope Z intersecte l'hyperplan G , ssi :

-
1. Constraint Satisfaction Problem
 2. Hybrid Constraint System

$$(e - d^T c) \in \left[-\sum_{i=1}^{i=p} |d^T g_i|, \sum_{i=1}^{i=p} |d^T g_i| \right] \quad (3.3)$$

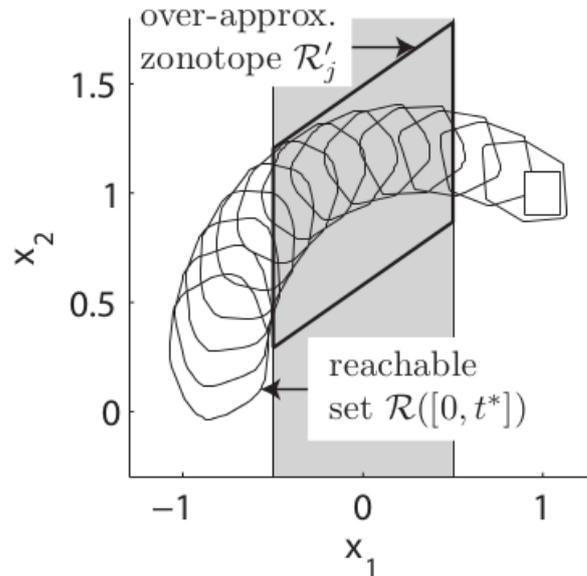


FIGURE 3.1 – Encadrement de l'ensemble qui intersecte la condition de garde par un zonotope (Althoff, 2010)

Malgré cette condition, l'utilisation des zonotopes n'est pas vraiment adaptée au cas des systèmes hybrides, les méthodes actuelles ne sont pas efficaces pour sur-approximer l'intersection³ avec une garde (le Guernic, 2005). En effet les zonotopes ne sont pas clos par intersection Combastel et al. (2008); Lalami and Combastel (2007).

Récemment dans ((Althoff and Krogh, 2012), (Althoff and Krogh, 2011), (Althoff et al., 2008)), les auteurs ont proposé des techniques de conversion d'ensemble efficaces pour résoudre le problème du franchissement des gardes dans le cas des systèmes hybrides linéaires. Ils considèrent que l'ensemble atteignable est représenté par des zonotopes et la condition de garde par un polytope⁴. Dès lors où il y a intersection entre le zonotope et le polytope (premier instant où la condition de garde est satisfaite), ils convertissent le zonotope en un polytope⁵, et ensuite récupèrent l'intersection de deux polytopes qui est un polytope (voir figure 3.2). Cet ensemble "intersection" calculé précédemment est encadré à nouveau dans un zonotope (voir figure 3.1)

3. L'intersection de deux zonotopes ou d'un zonotope avec un polytope n'est pas en général un zonotope .

4. Intersection finie de demi-espace.

5. Multi-Parametric Toolbox (MPT) de MatLab.

et ainsi de suite, jusqu'à ce que l'ensemble atteignable rencontre à nouveau une condition de garde. A noter que ces opérations de conversion et d'intersection sont réalisées grâce aux outils développés dans (Kvasnica et al., 2004).

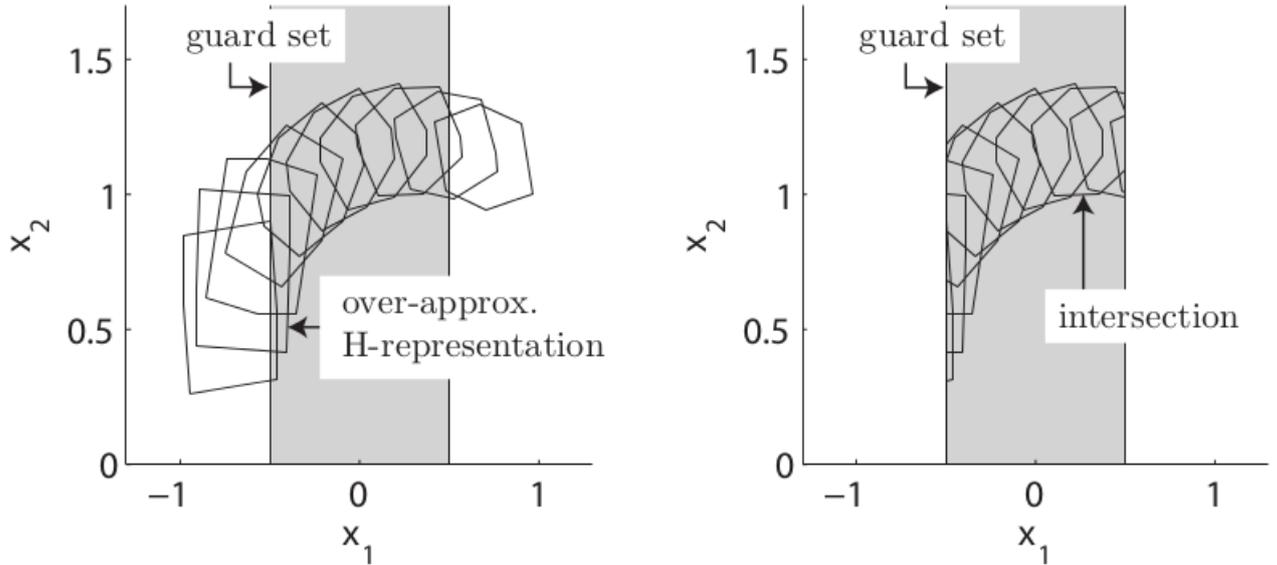


FIGURE 3.2 – Conversion zotopé → polytope et intersection polytope, polytope (Althoff, 2010)

3.4.1.2 ENSEMBLE ATTEIGNABLE REPRÉSENTÉ PAR DES POLYÈDRES

Dans (Girard, 2004) une autre formulation, basée sur les polyèdres est proposée. Soit $E_q = \{e \in \mathcal{E}, e = (q, q')\}$, le sous-ensemble des transitions discrètes de q vers q' . Pour tout $q \in Q$, on associe un ensemble de contraintes. Chaque contrainte correspond à une transition de l'ensemble E_q . La contrainte C_e , associée à la transition e peut s'écrire sous la forme d'une inégalité linéaire, $C_e = \{x \in \mathbb{R}^n, c_e^T x \geq d_e\}$ où c_e est un vecteur unitaire, et d_e un scalaire. Lorsque la variable discrète vaut q , la variable continue $x(t)$ doit satisfaire l'ensemble de contraintes :

$$\forall e \in E_q, c_e^T x(t) \geq d_e$$

Le domaine D_q (polyèdre, voir figure 3.3) est défini par

$$D_q = \bigcap_{e \in E_q} C_e = \{x \in \mathbb{R}^n, e \in E_q, c_e^T x \geq d_e\}$$

Quand $x(t)$ est sur le point de franchir l'hyperplan d'équation $c_e^T x = d_e$, alors la transition e s'effectue. Les gardes du système sont donc définies par la relation :

$$\forall e \in E_q, G_e = S_q \cap \{x \in \mathbb{R}^n, e \in E_q, c_e^T x \geq d_e\}$$

où S_q est l'ensemble des points sur la frontière de D_q . Très récemment, dans (Hagemann, 2014),

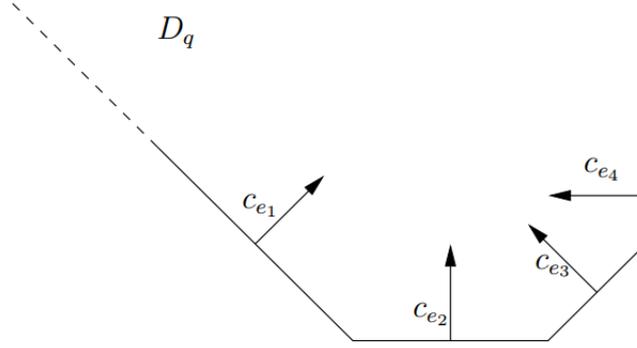


FIGURE 3.3 – Exemple d'un polyèdre D_q (Girard, 2004)

l'analyse d'atteignabilité des systèmes hybrides est effectuée en utilisant des projections orthogonales symboliques sur lesquelles différentes opérations géométriques, y compris des enveloppes convexes, des sommes de Minkowski, des applications linéaires, et des intersections, peuvent être réalisées efficacement et exactement. Dans (Frehse and Ray, 2012), il est proposé une approche pour le calcul de l'intersection du tube de trajectoires avec des gardes polyédrales avec une précision donnée. Le calcul de l'intersection du tubes de trajectoires avec la garde est ramené à un problème de minimisation convexe. Pour le résoudre, ils utilisent un algorithme similaire à celui de "sandwich" (Burkard et al., 1991), qui est un algorithme de minimisation de fonction convexe utilisé dans la littérature pour l'approximation.

3.4.1.3 ENSEMBLE ATTEIGNABLE REPRÉSENTÉ PAR DES FONCTIONS SUPPORTS

Soit $S \subseteq \mathbb{R}^n$ un ensemble compact convexe, la fonction support (Guernic and Girard, 2009b) de S est $\rho_S : \mathbb{R}^n \rightarrow \mathbb{R}$ définie par

$$\rho_S(l) = \max_{x \in S} l \cdot x \quad (3.4)$$

A travers cette définition, on peut (3.5) approcher l'ensemble S (voir figure 3.4). On peut remarquer que les vecteurs support $(l_1, l_2, l_3, \dots, l_r)$ ou direction de projection ne sont pas uniques.

$$S \subseteq \bigcap_{l \in \mathbb{R}^n} \{x \in \mathbb{R}^n : l \cdot x \leq \rho_S(l)\} \quad (3.5)$$

Les vecteurs supports $(l_1, l_2, l_3, \dots, l_r)$ sont choisis de façon arbitraire. Étant donné un vecteur, $\rho_S()$ est la solution d'un problème d'optimisation linéaire, ce qui revient à maximiser la fonction linéaire $l \cdot x$ pour tout $x \in S$.

Soit S , un ensemble convexe représenté par sa fonction support ρ_S et la condition de garde donnée par un hyperplan $G = \{x \in \mathbb{R}^n : c^\top \cdot x = d\}$. Il y a intersection entre S et G (voir figure 3.5), si et seulement si $-\rho_S(-c) \leq d \leq \rho_S(c)$.

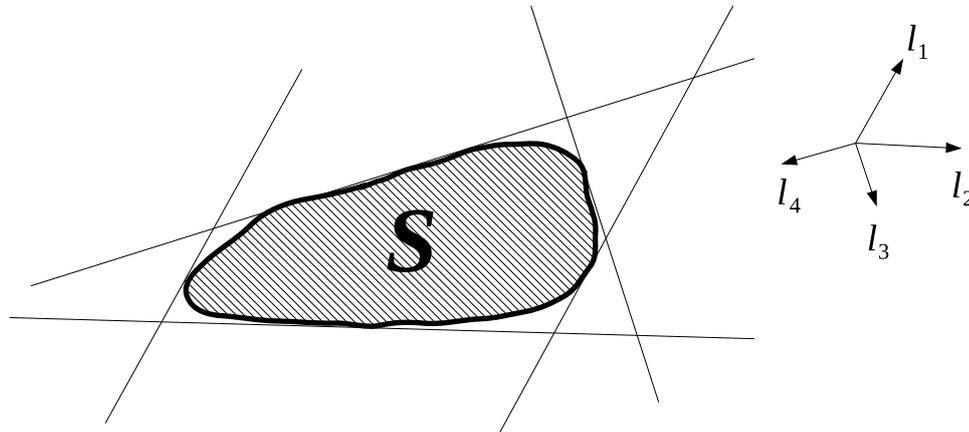


FIGURE 3.4 – Notion de fonction support

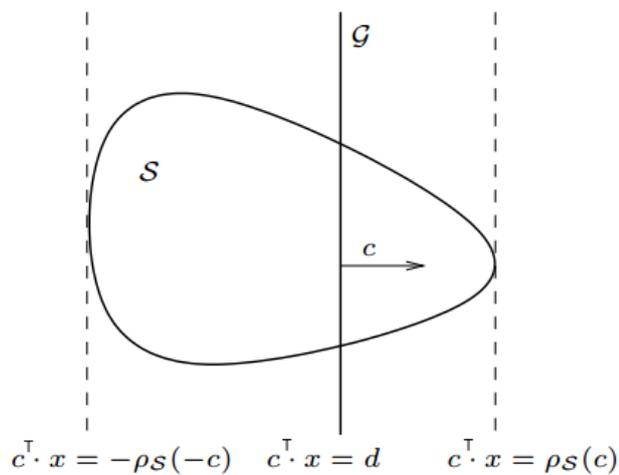


FIGURE 3.5 – Intersection ensemble (fonction support) et condition de garde (hyperplan) (Guernic and Girard, 2009b).

Pour le cas particulier des gardes définies par des hyperplans, dans (Guernic and Girard, 2009a), les auteurs ont montré comment transformer le problème de l'intersection de l'ensemble atteignable avec la garde en un problème de minimisation d'une fonction unimodale.

3.5 CAS OÙ LA GARDE EST NON LINÉAIRE

3.5.1 MÉTHODE PAR APPROXIMATION POLYNOMIALE DE LA GARDE

Dans (Shampine et al., 1987), une méthode permettant de tester si un événement a lieu dans un intervalle $[t_k, t_{k+1}]$ est proposée. Cette méthode consiste à construire une approximation polynomiale⁶ de la fonction de garde sur cet intervalle et ensuite à utiliser une séquence de Sturm⁷ pour déterminer le zéro d'une condition de garde linéaire. La localisation de l'événement est réalisée en utilisant la méthode de bisection conjointement avec une séquence de Sturm. L'inconvénient de cette méthode réside dans le fait qu'elle est assez coûteuse en temps de calcul. En approchant la condition de garde par un polynôme cubique et considérant toutes les configurations possibles du polynôme pour la détection d'événements, les événements sont localisés par la méthode *Regula-Falsi* et le schéma d'itération de Newton (Birta et al., 1985).

Dans (Park and Barton, 1996) est évoquée l'idée de combiner l'approximation polynomiale avec des méthodes de calcul par intervalle pour déterminer l'intervalle dans lequel les événements peuvent avoir lieu. Cette méthode est principalement basée sur un "test d'exclusion". En effet ce test d'exclusion est construit pour déterminer s'il est possible qu'un polynôme ait des racines dans un intervalle donné en utilisant le moins de calcul possible. Ce processus de détection se fait par le biais de la méthode Newton intervalle qui garantit l'existence d'un unique événement dans l'intervalle de temps. Ce test est basé sur l'opérateur de Krawczyk (Park and Barton, 1996)).

Toujours dans cette approche d'approximation polynomiale de la fonction de garde (ou fonction d'événements), (Esposito and Kumar, 2007; Esposito et al., 2001) proposent une transformation de cette fonction qui est initialement non linéaire, en une fonction linéaire en ajoutant une variable supplémentaire notée $z = g(x, t)$. D'où le nouveau système :

$$\{\dot{x}(t) = f(x); \quad g(x, t) \leq 0\} \rightarrow \left\{ \dot{x}(t) = f(x); \quad \dot{z}(t) = \frac{\partial g}{\partial x} \cdot f(x) + \frac{\partial g}{\partial t}; \quad z \leq 0 \right\}$$

Cet algorithme est une amélioration de celui proposé dans (Cellier, 1977), $\frac{\partial g}{\partial x} \cdot f(x)$ est une quantité très importante pour la détection d'événement, en effet elle traduit le taux de variation de la condition de garde le long du champ de vecteur. C'est en fait une dérivée de Lie (Röbenack, 2008; Tami et al., 2013).

Dans le même esprit, (Esposito and Kumar, 2007) ont proposé une technique robuste basée sur l'extrapolation des polynômes, qui serviront à détecter un changement discret se produisant dans le voisinage d'une singularité⁸ du modèle. Le déroulement de l'algorithme est le suivant, à la $k^{\text{ième}}$ itération on a :

6. Polynôme qui interpole la fonction de garde

7. Le nombre de racines réelles distinctes dans un intervalle $[a, b]$ d'un polynôme à coefficients réels, dont a et b ne sont pas des racines, est égal à la différence du nombre de changements de signe de la suite de Sturm aux bornes de cet intervalle.

8. Par singularité, on entend région de l'espace où le système est mal défini (discontinuité dans le système).

1. Construire un polynôme $P_k^i(h_k + 1)$ qui extrapole la fonction de garde $g^i(x(t_k + h_k, t_k + h_{k+1}))$ sur $h_{k+1} \in [0, \infty]$;
2. Déterminer si l'extrapolation polynomiale $P_k^i(h_k + 1)$ a une racine positive réelle h^* qui correspond au pas de simulation contenant la date de l'événement. Si tel est le cas on calcule la plus petite racine h^* . S'il n'existe pas de racine positive réelle, $h^* = \infty$;
3. Intégrer l'état x_k sur $h_{k+1} = \min(h^*, h^{err})$ ce qui produit x_{k+1} ;
4. Si un événement a lieu, $g^i(x_{k+1} + t_{k+1}) = 0$, mettre à jour le mode suivant ($i \rightarrow i'$)¹⁰, et l'ODE dans le nouveau mode, estimer $f^{i'}(x)$ et la fonction de garde devient $g^{i'}(x, t)$ et l'intégration recommence en utilisant x_{k+1} , où $t_{k+1} = t_k + h_{k+1}$.

Dans (Girard, 2002), une méthode est proposée pour la classe de systèmes hybrides linéaires par morceaux. Elle est basée sur une expression symbolique du tube de trajectoires. En plus de chercher à approximer l'instant auquel a eu lieu l'événement, on calcule aussi l'intervalle qui contient l'événement de manière garantie. Traditionnellement, un événement a eu lieu si la solution $X(t)$ du système hybride vérifie une condition $g(X(t)) = 0$ (g est la fonction de garde). Un algorithme "simple" pour détecter l'occurrence de cet événement est de calculer deux solutions successives $X(t_k)$, et $X(t_{k+1})$ et ainsi tester le signe de la fonction de garde dans cette intervalle. Malheureusement on peut avoir des cas où le signe de la fonction de garde change deux fois sur ce même intervalle $[t_k, t_{k+1}]$. Ainsi cette méthode n'est pas tout à fait adaptée pour des systèmes dont la dynamique continue est rapide.

3.5.2 MÉTHODES PAR INTERVALLES

Récemment, dans (Ramdani and Nedialkov, 2011), il a été proposé de combiner les méthodes de Taylor intervalles pour le calcul de l'expansion continue d'un ensemble avec les techniques de résolution de CSP pour détecter et localiser l'intersection du flux continu avec les frontières de l'invariant d'une part et avec les conditions de garde d'autre part. Il s'agit donc de déterminer entre deux instants consécutifs t_j et t_{j+1} l'ensemble des vecteurs z_j qui, partis de t_j , intersectent la garde à des instants t_e . La formulation la plus simple consiste à filtrer l'ensemble des vecteurs d'état initiaux z_j (tube de trajectoires) de façon à éliminer les vecteurs qui n'intersectent pas la garde pour tout $t \in [t_j, t_{j+1}]$ et de garder uniquement ceux qui intersectent la garde dans cet intervalle.

Dans (Ishii et al., 2009), le problème de détection d'événements pour un système hybride est formulé comme un HCS¹¹ constitué des contraintes sur les trajectoires et des contraintes de garde sur les états causant les changements discrets. Résoudre donc le HCS revient à calculer le domaine des boîtes faisables, ce qui veut dire encadrer les états avec des boîtes qui satisfont toutes les contraintes. La méthode développée pour résoudre le HCS est composée de deux approches :

-
9. Pas d'intégration estimé sur les considérations d'erreur de troncature
 10. L'indice "i" se réfère au mode courant
 11. Hybrid Constraint System

- la première basée sur l'intégration numérique garantie des ODEs non linéaire développée dans (Nedialkov et al., 1999),
- la deuxième basée sur la programmation par contraintes en utilisant les intervalles (ICP) (Henryck et al., 1997).

La technique ICP utilise la méthode de Newton intervalle pour une convergence quadratique de la réduction des boîtes, et elle garantit que les boîtes obtenues contiennent la solution. Cependant, la méthode n'est applicable qu'aux valeurs certaines et ne prend pas en compte des incertitudes paramétriques.

3.6 CONCLUSION

Dans la première partie de ce chapitre, nous sommes revenus sur les différentes méthodes permettant de résoudre de façon efficace le problème de valeur initiale (PVI) des équations différentielles ordinaires. Ceci revient à calculer l'ensemble atteignable par un système dynamique incertain non autonome décrit par une équation différentielle non linéaire.

Dans la deuxième partie, nous avons pu distinguer dans la littérature trois approches pour le passage des gardes dans l'étude des systèmes hybrides.

La première approche utilise le potentiel offert par les zonotopes, les fonctions support ou les polytopes. Ces méthodes ont montré un intérêt réel pour l'analyse d'atteignabilité des systèmes hybrides linéaires. Elles sont de plus capables de passer à l'échelle. Cependant, elles ne restent applicables qu'aux systèmes linéaires. Dans le cas non linéaire, il est possible de procéder à une linéarisation garantie, mais au prix de sur-approximations.

La seconde combine une approximation polynomiale de la condition de garde avec les algorithmes de recherche de zéro d'un polynôme. Cette approche est efficace mais est non garantie. De plus, elle ne tient pas compte de la présence d'incertitudes dans les états initiaux des vecteurs d'état et des paramètres du modèle.

La troisième approche utilise les méthodes de propagation de contraintes et l'intégration numérique d'équations différentielles non linéaires. Elle est directement applicable aux systèmes non linéaires. Par contre, ces méthodes ne prennent pas toutes en compte la présence d'incertitudes (sur le vecteur d'état et les paramètres). La méthode que nous proposerons dans le chapitre suivant fait partie de ce type d'approche. Nous avons considéré des incertitudes paramétriques et des incertitudes sur les vecteurs d'état.

CHAPITRE 4

MÉTHODES ENSEMBLISTES RAPIDES POUR L'ATTEIGNABILITÉ HYBRIDE

Sommaire

4.1	Introduction	52
4.2	Méthodes et algorithmes pour le problème de passage des gardes	53
4.2.1	Garde non linéaire et résolution par CSP	53
4.2.2	Garde linéaire	56
4.3	Méthode ensembliste pour le saut discret	68
4.3.1	Fonction de reset linéaire	68
4.3.2	Fonction de reset non linéaire	69
4.4	Méthodes pour la fusion des trajectoires	70
4.4.1	Calcul des sommets des sous-tubes	71
4.4.2	Enveloppe d'un nuage de points par un zonotope	73
4.4.3	Construction d'une MSBP à partir du zonotope englobant	76
4.4.4	Analyse de complexité	78
4.5	Évaluation de la méthode d'atteignabilité hybride sur des exemples simples	80
4.5.1	Exemple 1 : Balle rebondissante	80
4.5.2	Exemple 2 : Système masse-ressort à commutation	84
4.5.3	Exemple 3 : Balle rebondissante sur une surface sinusoïdale	87
4.6	Évaluation de la méthode d'atteignabilité hybride sur des benchmarks	91
4.6.1	Benchmark 1 : Manœuvre anti-collision	92
4.6.2	Benchmark 2 : Intégrateur non-holonyme	94
4.6.3	Benchmark 3 : Contrôle de la glycémie de patients diabétiques	95
4.7	Conclusion	100

4.1 INTRODUCTION

Nous regroupons dans ce chapitre nos contributions méthodologiques concernant l'atteignabilité hybride. Nous nous basons sur l'algorithme d'atteignabilité hybride proposé dans (Ramdani and Nedialkov, 2011) sur lequel nous avons procédé à des modifications et extensions techniques permettant d'améliorer ses performances. Nous avons en particulier combiné notre propre implémentation de l'algorithme de calcul du tube de trajectoires d'un système dynamique continu incertain par méthode de Taylor intervalle et l'algorithme de Lohner (Lohner, 1987b) pour contrôler l'effet d'enveloppement. Le chapitre est structuré comme décrit ci-dessous.

La première partie est divisée en deux sous-parties. La première sous-partie présente la nouvelle méthode proposée pour le passage ensembliste de gardes non linéaires et linéaires, c'est-à-dire la résolution du problème de la détection et localisation (temporelle) de l'intersection du tube de trajectoires continues avec des gardes ensemblistes. Nous proposons une méthode rapide basée sur la résolution d'un problème de satisfaction de contraintes. Elle utilise la bisection dans la seule direction du temps et fait collaborer plusieurs contracteurs intervalle pour réduire le domaine des vecteurs d'état localisés sur la garde durant la tranche de temps étudiée. La deuxième sous-partie traite des méthodes de complexité polynomiale pour la résolution du passage des gardes ensemblistes, dans le cas où la garde est linéaire. Ces méthodes sont basées sur des opérations géométriques d'intersection entre zonotope et hyperplan/demi-espace. L'avantage de ces méthodes réside dans le fait qu'elles sont réduites à des calculs d'algèbre linéaire simples (principalement, évaluation du noyau, décomposition en valeurs singulières (SVD)) qui assurent une complexité polynomiale et des temps de calcul réduits favorisant le passage à l'échelle.

La deuxième partie est consacrée à une méthode pour résoudre le problème du saut ensembliste, ce qui revient à calculer l'image d'un ensemble par une fonction qui peut être non linéaire. A ce titre, nous avons pu élaborer une expression analytique de la fonction de reset en fonction d'une représentation zonotopique (domaine de solution de notre intégrateur d'équations différentielles). Cela nous a permis de réduire la sur-approximation introduite lors du saut. En effet, en utilisant la représentation sous forme de boîte dans le repère original, nous introduisons des sur-approximations qui peuvent être néfastes aux performances de notre algorithme d'atteignabilité.

La troisième partie concerne la fusion des trajectoires qui est nécessaire comme une conséquence directe du problème de passage de gardes ensemblistes. En effet lors de la localisation de l'événement, notre algorithme procède par bisection (même si uniquement dans la direction de la variable temporelle), générant ainsi un nombre important de boîtes qui constituent chacune le point de départ d'un nouveau tube de trajectoires dans le nouveau mode. Nous avons donc proposé une méthode de fusion des boîtes utilisant les zonotopes, car notre intégrateur d'équations différentielles représente la solution sous forme de MSB \mathbb{R} ce qui est un zonotope particulier comme nous l'avons déjà relevé.

Les méthodes précédentes (le passage ensembliste des gardes, le saut ensembliste, et la fusion des trajectoires) utilisées conjointement nous permettent de calculer l'ensemble atteignable par un système dynamique hybride dont la dynamique continue est décrite par un système d'équations différentielles non linéaires et incertaines.

L'algorithme global est évalué sur des exemples simples ainsi que sur trois benchmarks en fin de chapitre.

4.2 MÉTHODES ET ALGORITHMES POUR LE PROBLÈME DE PASSAGE DES GARDES

Les algorithmes développés dans cette partie sont regroupés en fin de section.

4.2.1 GARDE NON LINÉAIRE ET RÉOLUTION PAR CSP

Nous relaxons de manière garantie le problème de passage ensembliste des gardes proposé dans (Ramdani and Nedialkov, 2011). Le passage ensembliste de gardes revient à résoudre un CSP. Le passage ensembliste de garde en présence d'incertitudes est réalisé pendant un continuum d'instant de passage. Dès lors, l'encadrement de la solution spatio-temporelle, c'est-à-dire l'encadrement simultané des instants de passage et des vecteurs d'états correspondant à la solution (l'intersection tube-garde) souffre du problème de sur-approximation et donc nécessite de recourir à des pavages obtenus par bisection. Notre stratégie consiste donc à procéder à la bisection dans la seule direction du temps (temps d'intégration qui correspond aussi au temps réel du système hybride) et d'ajouter la collaboration du contracteur HC4_Revise de IBEX (Chabert, 2013), pour réduire le domaine des vecteurs d'état localisés sur la garde durant la tranche de temps étudiée. Nous réalisons la bisection uniquement dans la direction de la variable temporelle pour améliorer la précision de la détection de l'instant de passage de garde, mais nous acceptons de dégrader la finesse de l'encadrement de l'intersection tube-garde, tout en restant conservatif. En bisectant dans une seule direction, nous pouvons ainsi garder une complexité polynomiale¹ pour l'algorithme. Finalement, le contracteur est utilisé sur la tranche du tube de trajectoires correspondant à un intervalle de temps que nous pouvons rendre aussi fine que possible. Nous savons à présent comment conceptualiser l'intersection géométrique du tube de trajectoire continue avec l'ensemble garde². La question première est de détecter si le tube de trajectoires intersecte l'ensemble de garde, puis de calculer quand et où l'intersection a lieu, en d'autres termes nous avons besoin de calculer les instants t_e et les vecteurs d'état solution $z(t_e)$ tels que $\gamma_e(z(t_e)) = 0$ est satisfait. Comme le tube de trajectoires a une dimension non-nulle, il existe un continuum d'instant où l'intersection a lieu. Donc, nous avons besoin de caractériser l'ensemble de toutes ces solutions, c'est-à-dire,

1. pour des dimensions de travail modérées.

2. La même méthode de calcul est appliquée pour le calcul de l'invariant.

$$\mathcal{T}^* \times \mathcal{Z}^* = \{t_e \times z(t_e) \text{ tel que } (t_e \in [t_j, t_{j+1}]) \wedge (\gamma_e(z(t_e)) = 0) \wedge (\dot{z}(t) = f_q(z)) \wedge (z(t_j) \in [z]_j)\} \quad (4.1)$$

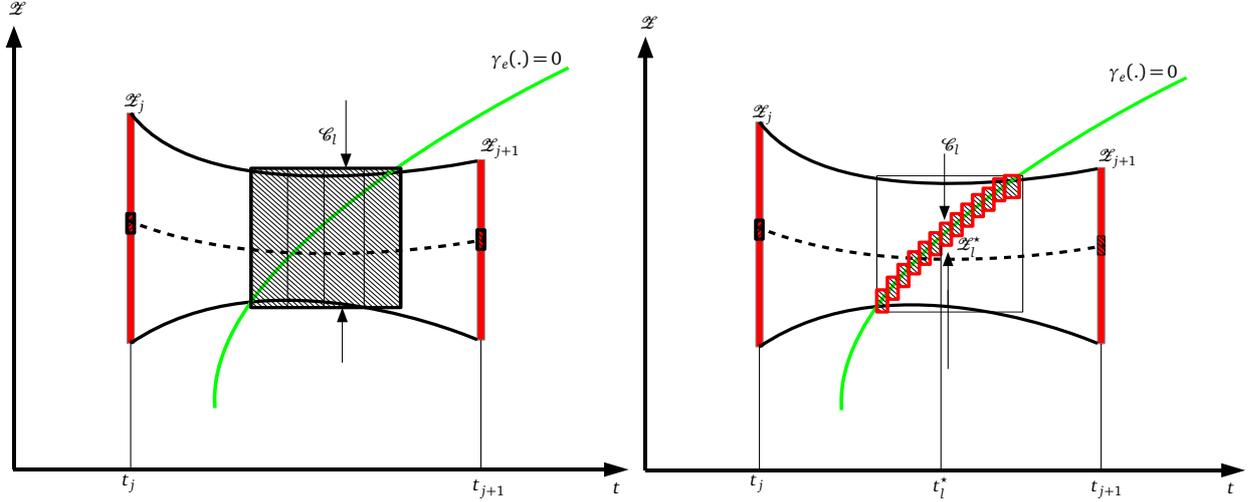


FIGURE 4.1 – Illustration de l'algorithme de calcul d'intersection

Supposons qu'il existe un événement pour $t_e \in [t_j, t_{j+1}]$, ainsi la méthode décrite dans la suite est capable de détecter l'existence d'un tel événement. Le calcul de l'ensemble solution (4.1) est maintenant un problème analytique. L'algorithme $\varphi^{QR}(t)$ donne pour tout t dans $[t_j, t_{j+1}]$ une solution analytique pour le tube de trajectoires sur l'intervalle de temps $[t_j, t_{j+1}]$, donc la méthode décrite dans la section 2.2.2 s'applique directement. Pour une caractérisation fine de (4.1), nous partitionnons l'espace de recherche.

Notons par $[t^*]_\ell = [\underline{t}^*, \bar{t}^*]_\ell \subseteq [t_j, t_{j+1}]$ un sous-intervalle sur lequel $\gamma_e(z(t_e)) = 0$ est satisfaite, et \mathcal{Z}_ℓ^* l'ensemble des vecteurs d'état pour lesquels il existe t dans $[\underline{t}^*, \bar{t}^*]_\ell$ qui satisfait $\gamma_e(z(t_e)) = 0$. L'ensemble solution (4.1) peut donc être sur-approximé par

$$\mathcal{T}^* \times \mathcal{Z}^* \subseteq \bigcup_{\ell=1}^L [\underline{t}^*, \bar{t}^*]_\ell \times \mathcal{Z}_\ell^* \quad (4.2)$$

où L est le nombre de sous-boîtes solutions.

Enfin, notre objectif est de contracter le tube de trajectoires sur de petits intervalles de temps pour calculer les plus petites boîtes solution englobant l'intersection entre le tube et la garde, comme illustré sur la figure 4.1. En permettant une certaine sur-approximation lors du calcul de t_e , nous disons qu'un événement a lieu sur le sous-intervalle $[\underline{t}^*, \bar{t}^*]_\ell$ si $w([\underline{t}^*, \bar{t}^*]_\ell)$ est plus petit qu'un seuil donné ϵ_T , comme suggéré dans (Ramdani and Nediakov, 2011). Maintenant,

la question demeure sur la manière de calculer une sur-approximation assez fine de \mathcal{X}_ℓ^* . Nous résolvons cette question par un problème de satisfaction de contraintes comme proposé dans (Maïga et al., 2014, 2013). La résolution est répétée pour $\ell = 1, \dots, L$.

Rappelons que l'enveloppe de la solution de (2.4), qui est un PVI EDO, peut être calculée pour un intervalle de temps petit $[t^*]_\ell = [\underline{t}^*, \bar{t}^*]_\ell$ sous la forme composée (4.3), que l'on note $[\chi]_\ell$, obtenu par une fonction d'inclusion pour l'algorithme $\varphi^{QR}(t)$, que nous obtenons en utilisant un intervalle $[t]$ en entrée de l'algorithme $\varphi^{QR}(t)$. Nous retenons la notation $[\chi^*]_\ell$ pour la forme composée caractérisant la sur-approximation fine de \mathcal{X}_ℓ^*

Nous avons

$$[\chi]_\ell = \{[z]_\ell, \hat{z}_\ell, [v]_\ell, [r]_\ell, A_\ell^*\}. \quad (4.3)$$

Supposons que $w([\underline{t}^*, \bar{t}^*]) \leq \epsilon_T$. Par conséquent

$$(\exists z \in [z]_\ell \text{ s.t. } \gamma_e(z) = 0) \Rightarrow (\exists v \times r \in [v]_\ell \times [r]_\ell \text{ s.t. } \gamma_e(v + A_\ell^* r) = 0), \quad (4.4)$$

Donc le calcul de l'intersection sur l'intervalle de temps $[t^*]$ se réduit à la résolution d'un CSP

$$\mathcal{E}_\ell := (\mathcal{C}_\ell, [v]_\ell \times [r]_\ell). \quad (4.5)$$

$$\text{où } \mathcal{C}_\ell := (\gamma_e(v + A_\ell^* r) = 0). \quad (4.6)$$

En utilisant un contracteur, nous pouvons obtenir une sur-approximation fine de \mathcal{X}^* sous la forme composée $[\chi^*]$, c'est-à-dire.

$$[v^*]_\ell \times [r^*]_\ell = \text{Contractor}(\mathcal{C}_\ell, [v]_\ell \times [r]_\ell). \quad (4.7)$$

Ici, nous utilisons un contracteur forward-backward implémenté dans la méthode HC4_Revise de la toolbox IBEX (Chabert, 2013). Comme l'ensemble condition de garde (2.6) est naturellement défini dans l'espace des z et le tube de trajectoires est défini comme une MSBP, nous avons besoin de relier la solution de (2.6) caractérisée naturellement dans l'espace des z à une MSBP dans l'espace $v \times r$ comme l'implique (4.4). Nous introduisons inévitablement des sur-approximations en travaillant dans l'espace des z plutôt que dans l'espace dans lequel la MSBP est définie. Pour freiner cette sur-approximation, l'idée est de combiner les solutions de deux contracteurs comme couramment fait lors de la construction de solveurs de CSP (voir (Jaulin et al., 2001), pp.90). Par conséquent, la contrainte (4.6) est réécrite en utilisant les contraintes redondantes comme suit

$$\mathcal{C}_\ell^R := (\gamma_e(v + A_\ell^* r) = 0) \wedge (\gamma_e(z) = 0) \wedge (z = v + A_\ell^* r), \quad (4.8)$$

dont la solution est obtenue par

$$[v^*]_\ell \times [r^*]_\ell \times [z^*]_\ell = \text{Contractor}(\mathcal{C}_\ell^R, [v]_\ell \times [r]_\ell \times [z]_\ell). \quad (4.9)$$

Notre intégrateur d'équations différentielles calcule la solution sur une grille temporelle, autrement dit la solution est calculée à chaque instant t ponctuel. Aussi, l'instant où l'événement a lieu n'est pas connu de façon précise, par contre nous sommes capable d'encadrer dans un intervalle $t_e = [\underline{t}^*, \bar{t}^*]$ cet instant. Ainsi, au niveau de la réinitialisation dans le nouveau mode, nous utilisons la relaxation garantie proposée par (Ramdani and Nedialkov, 2011).

Proposition 1 (Ramdani and Nedialkov, 2011) *Si l'ensemble solution pour le CSP (4.9) est non vide, nous considérons que l'événement $e = q \rightarrow q'$ a lieu à $t_e = \underline{t}^*$ et que $[\chi](t_e^-) = [\chi^*]_\ell$.*

4.2.2 GARDE LINÉAIRE

La première méthode, basée sur les CSP, est générale, c'est-à-dire qu'elle est applicable aussi bien aux gardes linéaires, qu'aux gardes non linéaires. Les méthodes que nous allons décrire dans cette section sont restreintes aux gardes linéaires (hyperplan, demi-espace, polytope). Ces méthodes restent applicables aux gardes non-linéaires, à condition de procéder à une linéarisation garantie de la garde au moment où le tube intersecte la garde.

4.2.2.1 INTERSECTION ZONOTOPE/HYPERPLAN

Cette méthode polynomiale pour la résolution du passage des gardes ensembliste est basée sur des opérations géométriques d'intersection entre zonotope et hyperplan. Nous utilisons l'approche initialement définie dans (Combastel et al., 2008) pour calculer l'intersection de deux zonotopes. Comparée à la première approche présentée dans la section 4.2.1, cette méthode est réduite à des calculs d'algèbre linéaire simples qui lui procurent une complexité polynomiale et des temps de calculs réduits qui favorisent le passage à l'échelle.

Proposition 2 (Combastel et al., 2008) *Considérons un zonotope $Z = c \oplus RB^p$ et un hyperplan $\mathcal{H}(\eta, d) = \{x | \eta^\top x = d\}$. Définissons*

$$\begin{aligned} \sigma &= R^\top \eta, \\ d' &= d - \eta^\top c, \\ c_s &= \sigma d' / \|\sigma\|_2^2, \\ V_0 &= N(\sigma), \text{ c'est-à-dire le noyau }^3 \text{ de } \sigma, \\ R_s &= V_0 V_0^\top, \\ c_\cap &= c + R c_s, \text{ et} \\ R_\cap &= R R_s, \end{aligned}$$

donc le zonotope englobant l'intersection entre le zonotope Z et l'hyperplan $\mathcal{H}(\eta, d)$ est donné par

$$Z \cap \mathcal{H} \subseteq Z_\cap = c_\cap \oplus R_\cap B^p. \quad (4.10)$$

3. Soit une matrice B $m \times n$, $N(B) = \{x \in \mathbb{R}^n; Bx = 0\}$ est le noyau de B ou le Kernel de B .

Preuve 1 Soit $x \in Z$, donc $s \in \mathbf{B}^p$ existe tel que $x = c + Rs$. Le vecteur ponctuel x appartient aussi à l'hyperplan $\mathcal{H}(\eta, d)$ si $\eta^\top x = d$, donc si $\eta^\top c + \eta^\top Rs = d$. Introduisons $\sigma = R^\top \eta$ et $d' = d - \eta^\top c$. Nous pouvons déduire une condition nécessaire et suffisante

$$x \in Z \cap \mathcal{H} \Leftrightarrow x \in \{c \oplus Rs \mid s \in \mathbf{B}^p \wedge \sigma^\top s = d'\} \quad (4.11)$$

Englober $Z \cap \mathcal{H}(\eta, d)$ revient à trouver tous les $s \in \mathbf{B}^p$ tels que $\sigma^\top s = d'$. Cette équation peut être résolue en trouvant la décomposition en valeurs singulières (SVD) du vecteur σ^\top , qui prend une forme particulière

$$\sigma^\top = U.S.V = [U_1] \cdot [S_1] \cdot \begin{bmatrix} V_1^\top \\ V_0^\top \end{bmatrix} \quad (4.12)$$

où $U_1 = 1$, $S_1 = \|\sigma\|_2$, $V_1 = \frac{\sigma}{\|\sigma\|_2}$, $V_0 = N(\sigma)$, et où $\|\sigma\|_2$ est la norme euclidienne de σ ($\|\sigma\|_2^2 = \sigma^\top \sigma$) et $N(\sigma)$ est une base orthonormale du noyau de σ . Introduisons $c_s = \sigma d' / \|\sigma\|_2^2$ et $R_s = V_0 V_0^\top$, une sur-approximation $[s]$ de l'ensemble de tous les $s \in \mathbf{B}^p$ qui satisfont $\sigma^\top s = d'$ est donnée par

$$[s] = c_s \oplus R_s \mathbf{B}^p \quad (4.13)$$

En combinant (4.11) et (4.13), il vient que :

$$x \in Z \cap \mathcal{H} \Rightarrow x \in c + R(c_s \oplus R_s \mathbf{B}^p) \Leftrightarrow x \in (c + Rc_s) \oplus RR_s \mathbf{B}^p. \quad (4.14)$$

Introduisons $c_\cap = c + Rc_s$, et $R_\cap = RR_s$, nous avons

$$x \in Z \cap \mathcal{H} \Rightarrow x \in Z_\cap = c_\cap \oplus R_\cap \mathbf{B}^p. \quad (4.15)$$

Corollaire 1 Par construction, le zonotope Z_\cap englobant l'intersection comme définie dans (4.11), est inclus dans l'hyperplan \mathcal{H} .

Preuve 2 Soit $x \in c_\cap \oplus R_\cap \mathbf{B}^p$ donc $\exists s \in \mathbf{B}^p$ tel que $x = c_\cap + R_\cap s$, alors $\eta^\top x = \eta^\top (c + R\sigma d' / \|\sigma\|_2^2 + RV_0 V_0^\top s) = \eta^\top c + \|\sigma\|_2^2 d' / \|\sigma\|_2^2 + \sigma^\top V_0 V_0^\top s$, car $\sigma^\top = \eta^\top R$. Comme $\sigma^\top V_0 = 0$ par construction de V_0 , et comme $d' = d - \eta^\top c$, $\eta^\top x = d$ ce qui complète la démonstration.

Finalement, la proposition 2 et l'équation (4.10) donnent un schéma de calcul de la sur-approximation de $Z \cap \mathcal{H}$ résumé dans l'algorithme 5. Cette approximation peut présenter un certain conservatisme, mais nous avons constaté qu'il est raisonnable pour nos applications. En se référant à l'équation (4.3), nous considérons le tube de trajectoires \mathcal{Z}_ℓ calculé sous le sous-intervalle $[t^*]_\ell = [\underline{t}^*, \bar{t}^*]_\ell$ en utilisant une fonction d'inclusion pour l'algorithme $\varphi^{QR}(t)$ et donné sous la forme composée (2.26). En retirant l'indice l pour des raisons de simplification, nous avons

$$\mathcal{Z} = \{v + A^* r \mid v \in [v], r \in [r]\} \subseteq A^*[r] \oplus [v]. \quad (4.16)$$

Algorithme 5 : Algorithme Intersection_Z_Hyperplan

input : c, R, η, d
output : c_\cap, R_\cap
1 $\sigma = R^\top \eta$;
2 $d' = d - \eta^\top c$;
3 $V_0 = N(\sigma)$;
4 $c_s = \sigma d' / \|\sigma\|_2^2$;
5 $R_s = V_0 V_0^\top$;
6 $c_\cap = c + R c_s$;
7 $R_\cap = R R_s$

En outre, il est intéressant de remarquer que chaque domaine \mathcal{Z} est un zonotope particulier. En effet, en introduisant le vecteur

$$c = A^* \text{mid}([r]) + \text{mid}([v]) \in \mathbb{R}^n, \quad (4.17)$$

et la matrice

$$R = \begin{bmatrix} A^* dr([r]) \\ dr([v]) \end{bmatrix} \in \mathbb{R}^{n \times 2n} \quad (4.18)$$

où $dr(\cdot)$ est une notation courte pour $\text{diag}(\text{rad}(\cdot))$, nous pouvons écrire le tube de trajectoires comme le zonotope suivant

$$\mathcal{Z} = c \oplus R B^{2n} \quad (4.19)$$

Supposons que la condition de garde est caractérisée par un hyperplan \mathcal{H} , c'est-à-dire que l'équation (2.6) devient

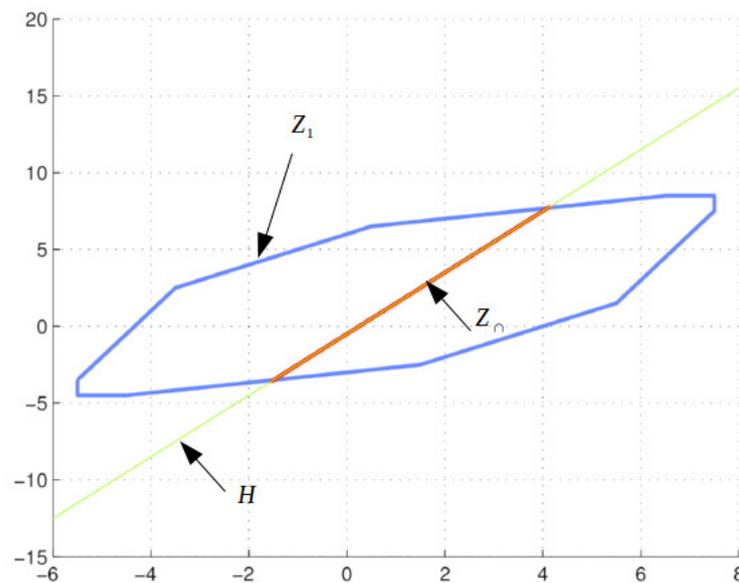
$$\gamma_e(z(t)) = 0 \Leftrightarrow \eta^\top z(t) = d \quad (4.20)$$

Ainsi, la MSBP $\mathcal{Z}^* = \mathcal{Z}_\cap$ qui sur-approxime $\mathcal{Z} \cap \mathcal{H}$ peut être directement obtenue en utilisant la proposition 2 avec $p = 2n$.

Exemple 4 Considérons le zonotope $Z_1 = c_1 \oplus R_1 B^p$ et un hyperplan $\mathcal{H}_1 = \{x | \eta_1^\top x = d_1\}$ avec

$$c_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad R_1 = \begin{bmatrix} 1 & 2 & 3 & 0.5 & 0 \\ 3 & 2 & 1 & 0 & -0.5 \end{bmatrix}, \quad \eta_1 = \begin{bmatrix} -2 \\ 1 \end{bmatrix}, \quad d_1 = -0.5;$$

Dans la Figure 4.2, la ligne rouge montre le zonotope obtenu Z_\cap englobant l'intersection $Z_1 \cap \mathcal{H}$.

FIGURE 4.2 – Intersection zonotope/hyperplan $Z_1 \cap \mathcal{H} \subseteq Z_0$

EXEMPLE 1 : COMPARAISON ENTRE L'APPROCHE DE RÉOLUTION PAR CSP ET L'APPROCHE ZONOTOPE/HYPERPLAN. Considérons le système dynamique hybride (brusselator hybride) avec deux modes $q = 1, 2$ et une transition $e = 1 \rightarrow 2$:

$$\left\{ \begin{array}{l} \text{flow}(1): \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 1 - (b_1 + 1)x_1 + a_1 x_1^2 x_2 \\ b_1 x_1 - a_1 x_1^2 x_2 \end{pmatrix} \\ \text{inv}(1): -4x_1 + x_2 + 2 < 0 \\ \text{flow}(2): \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 1 - (b_2 + 1)x_1 + a_2 x_1^2 x_2 \\ b_2 x_1 - a_2 x_1^2 x_2 \end{pmatrix} \\ \text{inv}(2): -4x_1 + x_2 + 2 > 0 \\ \text{guard}(1): -4x_1 + x_2 + 2 = 0 \\ \text{reset}(1): \rho_1(x_1, x_2) = L \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} \end{array} \right. \quad (4.21)$$

Avec $L = I_2^4$, $l_1 = l_2 = -0.5$, $a_1 = 1.5$, $a_2 = 3.5$, $b_1 = 1$, $b_2 = 3.5$ et $x_0 \in [2, 2.15] \times [0.1, 0.15]$. Nous considérons pour cette simulation, les réglages pour les algorithmes Transition-Hybride (algorithme 11 et 12) et φ^{QR} (voir **algorithme 3**) suivant : un pas d'intégration constant $h = 0.05$, l'intervalle de temps est bissecté jusqu'à un seuil $\varepsilon_T = 0.005$. Quand nous comparons les résultats obtenus dans les figures 4.3, nous voyons que la méthode utilisant l'intersection d'un zonotope avec un hyperplan fournit des temps de calcul moindre (Temps CPU=0.16s) par rapport à la méthode de résolution par CSP (Temps CPU= 0.20s). En outre, lorsque nous

nous focalisons sur la sur-approximation de l'intersection du tube et de la garde qui est représentée par des boîtes noires, nous voyons que l'approche par zonotope (volume des boîtes noires=0.0690590) fournit des intersections assez fines par rapport à l'approche par CSP (volume des boîtes noires=0.0968192) (voir aussi les figures 4.3-a,4.3-b).

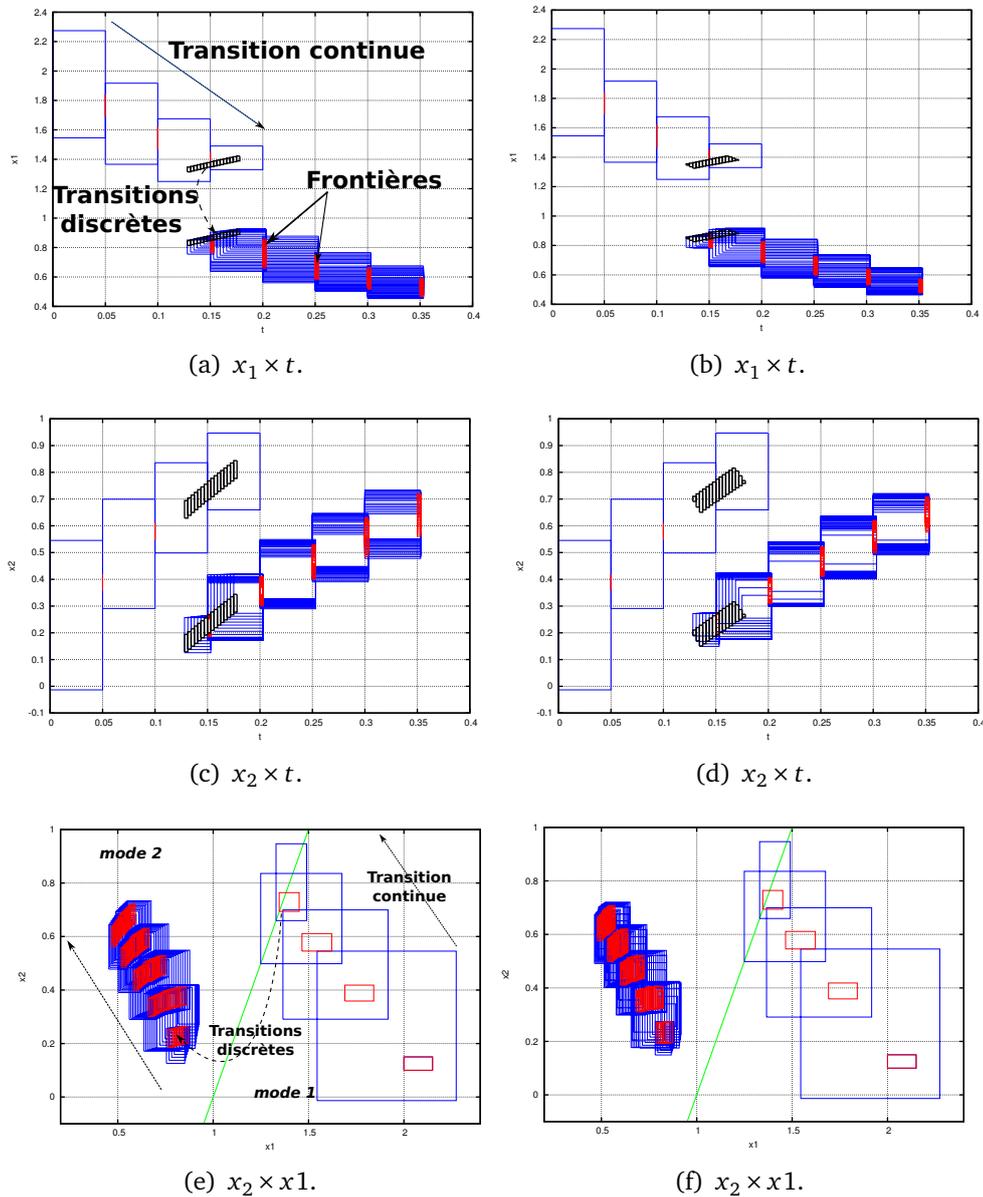


FIGURE 4.3 – Ensemble atteignable calculé pour (4.21). Les figures (a-c-e) illustrent la méthode de résolution par intersection zonotope/hyperplan, CPU times=0.16s ; les figures (b-d-f) illustrent la méthode de résolution par CSP, CPU times=0.20s.

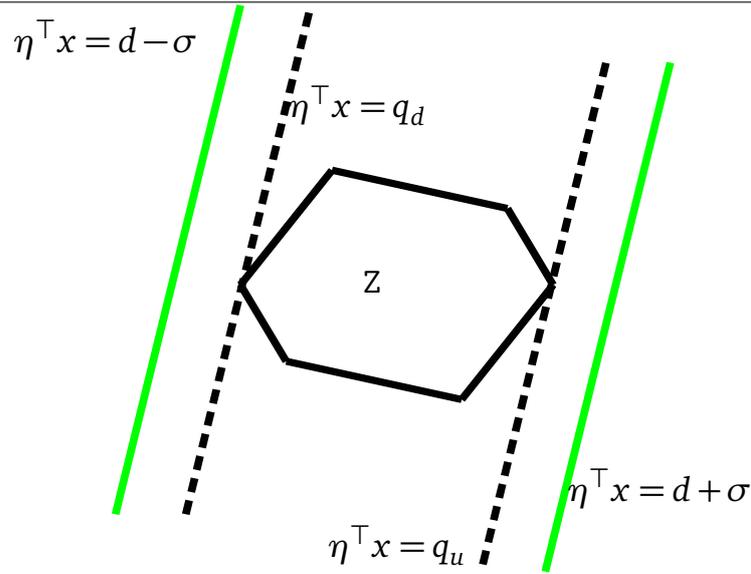


FIGURE 4.4 – Le "support" strip d'un zonotope

4.2.2.2 INTERSECTION ZONOTOPE/DEMI-ESPACE

Cette méthode de résolution du passage des gardes ensemblistes est basée purement sur la manipulation géométrique des ensembles convexes notamment les polytopes. Les polytopes sont l'intersection d'un nombre finis de demi-espaces $\mathcal{P} = \bigcap_i \mathcal{H}_i^{\leq} = \mathcal{H}_1^{\leq} \cap \mathcal{H}_2^{\leq} \dots \cap \dots \mathcal{H}_r^{\leq}$.

Nous commençons d'abord par décrire une méthode permettant de calculer l'intersection d'un zonotope et d'un strip⁵ avec une complexité polynomiale. Puis, nous en déduisons la procédure pour calculer l'intersection d'un zonotope et d'un demi-espace. Et une méthode générale sera proposée en fin de section permettant de calculer l'intersection d'un zonotope et d'un polytope.

DÉTECTION DE L'INTERSECTION ENTRE UN ZONOTOPE ET UN STRIP

Proposition 3 (Vicino and Zappa, 1996) *Étant donné un zonotope Z et un strip $\mathcal{S} = |\eta^\top x - d| \leq \sigma$, le "support" strip⁶ du zonotope (voir la figure 4.4) est défini par*

$$\mathcal{S}_Z = \{x \in \mathbb{R}^n \mid q_d \leq \eta^\top x \leq q_u\}, \quad (4.22)$$

où q_u et q_d sont définies comme

$$q_u = \max_{x \in Z} \eta^\top x \quad (4.23)$$

$$q_d = \min_{x \in Z} \eta^\top x \quad (4.24)$$

5. une bande de contraintes

6. bande support

qui sont calculées par

$$q_u = \eta^\top c + \|R^\top \eta\|_1 \quad (4.25)$$

$$q_d = \eta^\top c - \|R^\top \eta\|_1 \quad (4.26)$$

où $\|\cdot\|$ est la norme 1 d'un vecteur.

Théorème 3 (Alamo et al., 2005) Un zonotope $Z = c \oplus RB^p \in \mathbb{R}^n$ intersecte un strip $\mathcal{S} = \{x \in \mathbb{R}^n : |\eta^\top x - d| \leq \sigma\}$ si :

$$(q_d \leq d + \sigma) \wedge (q_u \geq d - \sigma) \quad (4.27)$$

INTERSECTION ZONOTOPE/STRIP

Définition 1 (Alamo et al., 2005) Étant donné un zonotope $Z = c \oplus RB^p \subset \mathbb{R}^n$ et un strip $\mathcal{S} = \{x \in \mathbb{R}^n : |\eta^\top x - d| \leq \sigma\}$ et le vecteur $\lambda \in \mathbb{R}^n$, définissons

$$\begin{aligned} \hat{c}(\lambda) &= c + \lambda(d - \eta c) \\ \hat{R}(\lambda) &= [(I - \lambda \eta)R \quad \sigma \lambda] \end{aligned}$$

$$\text{Donc } Z \cap \mathcal{S} \subseteq \hat{Z}(\lambda) = \hat{c}(\lambda) \oplus \hat{R}(\lambda)B^{p+1}$$

La sur-approximation de l'intersection ci-dessus peut être conservative. Pour obtenir une sur-approximation raisonnable, le vecteur λ doit être choisi selon un critère pour minimiser la taille du zonotope (Tabatabaeipour and Stoustrup, 2013).

- Dans (Alamo et al., 2005), deux critères sont utilisés pour minimiser la taille du zonotope :
- La minimisation de la longueur des segments du zonotope qui offre un calcul rapide du vecteur optimal λ , mais les résultats peuvent être conservatifs.
 - La minimisation du volume du zonotope qui offre un meilleur résultat en résolvant un problème d'optimisation convexe. Elle conduit parfois à des situations où le zonotope est très étroit, c'est-à-dire que l'incertitude dans une direction peut rester extrêmement large, mais dans le même temps le volume du zonotope tend vers zéro (Le et al., 2013).

Dans ce mémoire, nous utiliserons la première approche c'est-à-dire celle qui minimise la longueur des segments du zonotope, ce qui revient à minimiser la norme de *Frobenius* de $\hat{R}(\lambda)$. Le vecteur λ qui minimise la norme de *Frobenius* (Alamo et al., 2005) de $\hat{R}(\lambda)$ est donné par :

$$\lambda^* = \frac{RR^\top \eta^\top}{\eta RR^\top \eta^\top + \sigma^2} \quad (4.28)$$

Pour le calcul de l'intersection d'un zonotope Z et d'un demi-espace $\mathcal{H}^{\leq} = \{x | \eta^{\top} x \leq \gamma\}$, nous avons utilisé la méthode proposée dans (Tabatabaeipour and Stoustrup, 2013). Cette méthode consiste dans un premier temps à construire le "support" strip du zonotope Z suivant la direction η^{\top} que l'on notera \mathcal{S}_Z .

$$\mathcal{S}_Z = \{x \in \mathbb{R}^n | q_d \leq \eta^{\top} x \leq q_u\} \quad (4.29)$$

Donc le "support" strip le plus étroit de l'intersection du zonotope Z et du demi-espace $\mathcal{H}^{\leq} = \{x | \eta^{\top} x \leq \gamma\}$ suivant la direction η^{\top} est

$$\mathcal{S}_{Z \cap \mathcal{H}^{\leq}} = \{x \in \mathbb{R}^n | q_d \leq \eta^{\top} x \leq \gamma\} \quad (4.30)$$

Soit $q_d = d - \sigma$ et $\gamma = d + \sigma$, donc nous avons un strip $\mathcal{S}_{Z \cap \mathcal{H}^{\leq}} = \{x \in \mathbb{R}^n : |\eta^{\top} x - d| \leq \sigma\}$ avec $\sigma = \frac{\gamma - q_d}{2}$ et $d = \frac{\gamma + q_d}{2}$.

Maintenant le problème de l'intersection d'un zonotope et d'un demi-espace revient à calculer l'intersection entre le zonotope Z et le strip $\mathcal{S}_{Z \cap \mathcal{H}^{\leq}}$. On peut donc calculer cette intersection en utilisant la méthode de minimisation des longueurs des segments évoquée précédemment.

Donc, le zonotope obtenu est approximé par $Z_{int} = \hat{c} \oplus \hat{R}B^{p+1}$ où : $\hat{c} = c + \lambda^*(d - \eta c)$, $\hat{R} = [(I - \lambda^* \eta)R \quad \sigma \lambda^*]$,

$$\text{avec } \lambda^* = \frac{RR^{\top} \eta^{\top}}{\eta RR^{\top} \eta^{\top} + \sigma^2}$$

Proposition 4 *Étant donné un zonotope $Z = c \oplus RB^p$ et un demi-espace $\mathcal{H}^{\leq} = \{x | \eta^{\top} x \leq \gamma\}$. Le zonotope englobant l'intersection entre le zonotope Z et un demi-espace \mathcal{H}^{\leq} est donné par*

$$Z \cap \mathcal{H}^{\leq} \subseteq Z_{\cap} = \hat{c} \oplus \hat{R}B^{p+1}. \quad (4.31)$$

avec

$$\begin{aligned} \hat{c} &= c + \lambda^*(d - \eta c), \\ \hat{R} &= [(I - \lambda^* \eta)R \quad \sigma \lambda^*], \\ \text{et } \lambda^* &= \frac{RR^{\top} \eta^{\top}}{\eta RR^{\top} \eta^{\top} + \sigma^2}, \sigma = \frac{\gamma - q_d}{2}, d = \frac{\gamma + q_d}{2}, q_d = \eta^{\top} c - \|R^{\top} \eta\|_1 \end{aligned}$$

En considérant que la condition de garde est caractérisée par un demi-espace $\mathcal{H}^{\leq} = \{x | \eta^{\top} x \leq \gamma\}$, c'est-à-dire l'équation (2.6) se simplifie à

$$\gamma_e(z(t)) = 0 \Leftrightarrow \eta^{\top} z(t) \leq \gamma \quad (4.32)$$

La MSBP $\mathcal{Z}^* = \mathcal{Z}_{\cap}$ qui sur-approxime $\mathcal{Z} \cap \mathcal{H}^{\leq}$ peut être directement obtenue en utilisant la proposition 5 avec $p = 2n$ et l'algorithme 6.

ILLUSTRATION SUR UN EXEMPLE SIMPLE : INTERSECTION ZONOTOPE/DEMI-ESPACE En considérons le même système dynamique hybride que l'exemple 7 avec les mêmes réglages sur les algorithmes Transition-Hybride (**algorithme 13**) et φ^{QR} (voir **algorithme 3**). Quand nous

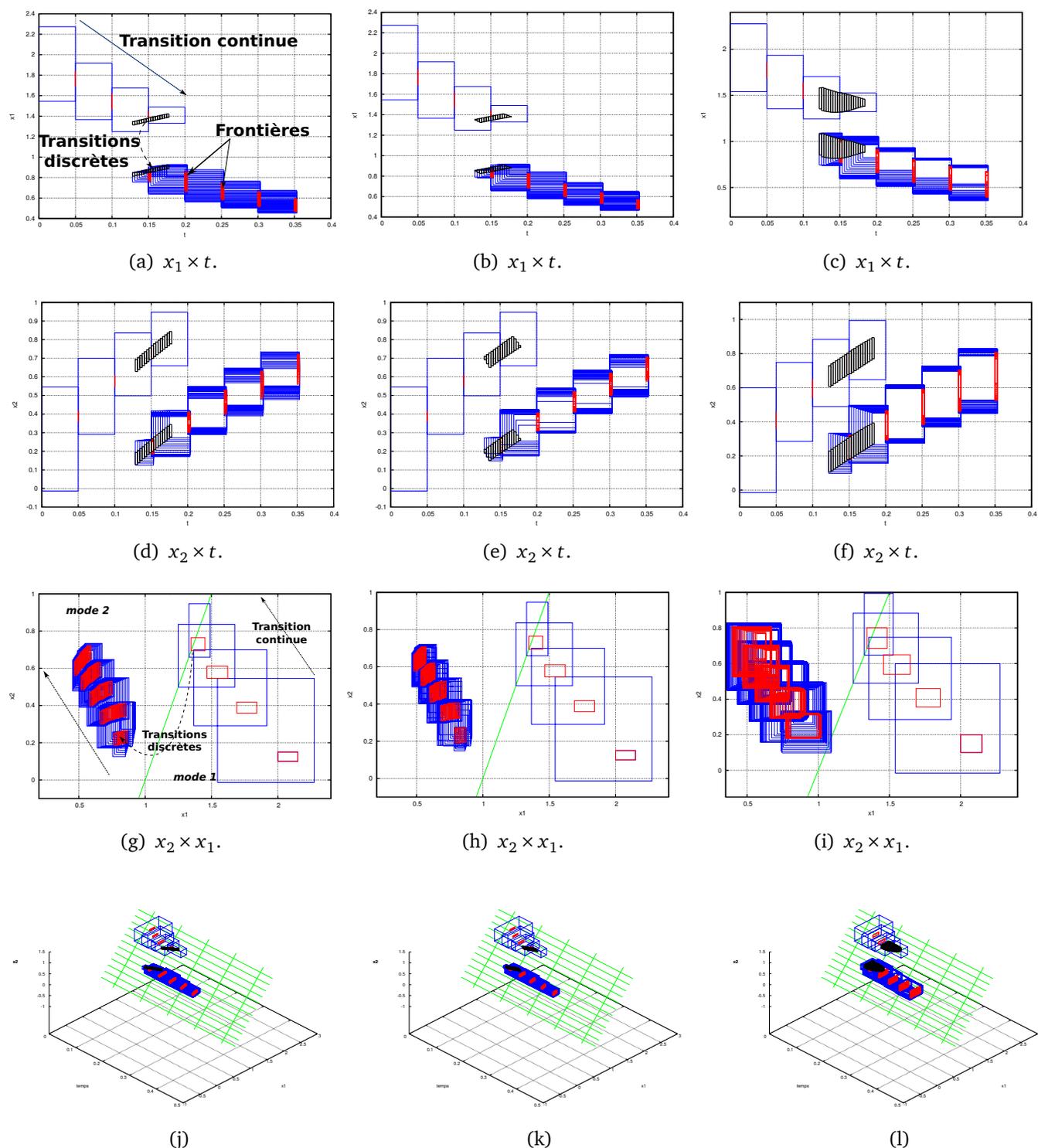


FIGURE 4.5 – Illustration sur l'exemple 4.21 des trois méthodes de passage ensembliste des gardes, les figures (a-d-g-j) illustrent la méthode de résolution par intersection zonotope/hyperplan, les figures (b-e-h-k) illustrent la méthode de résolution par CSP, les figures (c-f,i,l) illustrent la méthode de résolution par intersection zonotope/demi-espace, illustration 3D dans l'espace $x_1 \times x_2 \times t$ pour les trois méthodes

Algorithme 6 : Algorithme Int_Z_demi_espace

```

input :  $Z, \eta, \gamma$ 
output :  $Z_\cap \subseteq Z \cap \mathcal{H}^\leq$ 
1  $q_u = \eta^\top c + \|R^\top \eta\|_1;$ 
2  $q_d = \eta^\top c - \|R^\top \eta\|_1;$ 
3 if  $q_d \leq \gamma$  then
4   if  $q_u \leq \gamma$  then
5      $Z_\cap := Z$ 
6   else
7      $\sigma = \frac{\gamma - q_d}{2};$ 
8      $d = \frac{\gamma + q_d}{2};$ 
9      $\lambda^* = \frac{RR^\top \eta^\top}{\eta RR^\top \eta^\top + \sigma^2};$ 
10     $\hat{c} = c + \lambda^*(d - \eta c);$ 
11     $\hat{R} = [(I - \lambda^* \eta)R \quad \sigma \lambda^*];$ 
12     $Z_\cap = \hat{c} \oplus \hat{R} \mathbf{B}^{p+1};$ 
13  end if
14 else
15    $Z_\cap = \emptyset;$ 
16 end if

```

comparons les résultats obtenus dans les figures 4.5-c-f-i-l, nous voyons que la méthode utilisant l'intersection d'un zonotope avec un demi-espace fournit des temps de calcul presque similaires (Temps CPU=0.176s) par rapport à la méthode de résolution par CSP (Temps CPU= 0.20s) et par la méthode de résolution par intersection zonotope/hyperplan. En outre, lorsque nous nous focalisons sur la sur-approximation de l'intersection du tube et de la garde représentée par des boîtes noires, nous voyons aussi que l'approche par zonotope/demi-espace (volume des boîtes noires=0.9565156) est dix fois plus pessimiste en terme de volume que celui des deux méthodes (voir la figure 4.5-c).

Remarque 2 En comparant les trois méthodes de passage ensembliste sur l'exemple 4.21, nous remarquons que la méthode de résolution par zonotope/hyperplan est la meilleure, par contre elle n'est applicable qu'aux gardes linéaires. Le tableau 4.1 donne les différents temps de calcul (Temps CPU), le volume des boîtes intersectant la condition de garde (Vol), le nombre de boîtes solutions (NBSol) à la fin de l'horizon temporel, des différentes méthodes proposées dans ce mémoire pour le passage ensembliste des gardes.

Méthode 1 : Méthode de résolution par CSP ;

Méthode 2 : Méthode de résolution par intersection zonotope/hyperplan ;

Méthode 3 : Méthode de résolution par intersection zonotope/demi-espace ;

	Temps CPU(s)	Vol	NBSol
Méthode 1	0.16	0.0968192	17
Méthode 2	0.2	0.0690590	17
Méthode 3	0.176	0.956515	20

TABLE 4.1 – Comparatif des performances des trois méthodes de passage ensembliste des gardes

Algorithme 7 : Algorithm Intersection_Z_Polytope

```

input :  $Z, \mathcal{P} = \cap_i^r \mathcal{H}_i^{\leq}$ 
output :  $Z_{\cap} \subseteq Z \cap \mathcal{P}$ 
1  $Z_{\cap} = Z$ ;
2 for  $i = 1$  to  $r$  do
3   |  $Z_{\cap} = \text{Int\_Z\_demi\_espace}(Z_{\cap}, \mathcal{H}_i^{\leq})$ ;
4 end for

```

INTERSECTION ZONOTOPE/POLYTOPE Maintenant nous savons calculer l'intersection entre un zonotope et un demi-espace. Pour calculer l'intersection entre un zonotope Z et un polytope $P_0 = \cap_i^r \mathcal{H}_i^{\leq}$, il suffit donc de calculer séquentiellement l'intersection du zonotope Z avec le i ème demi-espace \mathcal{H}_i^{\leq} (voir l'**algorithme 7**). A noter que l'on pourrait utiliser cette méthode pour l'estimation d'état hybride, à condition que l'on dispose d'une méthode permettant de transformer un zonotope en polytope. Dans ce cas le problème du calcul de l'intersection pour l'estimation d'état hybride se ramènerait à calculer l'intersection entre deux polytopes. Nous verrons dans la section 5.2.1 que la résolution du problème d'estimation d'état hybride se ramène à un moment donné de l'algorithme à calculer l'intersection entre deux zonotopes.

ILLUSTRATION NUMÉRIQUE DE LA MÉTHODE D'INTERSECTION ZONOTOPE/POLYTOPE. Considérons le système dynamique hybride (brusselator hybride) avec deux modes $q = 1, 2$ et une transition $e = 1 \rightarrow 2$ de l'exemple 7. Nous considérons pour cette simulation, les réglages pour les algorithmes Transition-Hybride (**algorithme 14**) et φ^{QR} (voir **algorithme 3**) suivant : un pas d'intégration constant $h = 0.05$, l'intervalle de temps est bissecté jusqu'à un seuil $\varepsilon_T = 0.005$, avec les conditions initiales $x_0 \in [2.1, 2.3] \times [0.1, 0.12]$. La garde est un polytope défini par l'intersection de trois demi-espace (voir figure 4.6-c) $\mathcal{H}_1^{\leq} = \{x | \eta_1^{\top} x \leq \gamma_1\}$, $\mathcal{H}_2^{\leq} = \{x | \eta_2^{\top} x \leq \gamma_2\}$, et $\mathcal{H}_3^{\leq} = \{x | \eta_3^{\top} x \leq \gamma_3\}$. Avec $\eta_1 = [1 \quad -2]^{\top}$, $\gamma_1 = 2$, $\eta_2 = [1 \quad 0.4]^{\top}$, $\gamma_2 = -1.4$, $\eta_3 = [1 \quad -0.4]^{\top}$, $\gamma_3 = -0.2$.

Les résultats de cette simulation sont illustrés sur les figures 4.6 comme on peut le voir, cette méthode introduit une sur-approximation importante de l'intersection entre le tube de trajectoires et la garde, ceci s'explique en partie par le fait que cette méthode découle de celle qui consiste à calculer l'intersection entre un zonotope et un demi-plan que nous avons traitée.

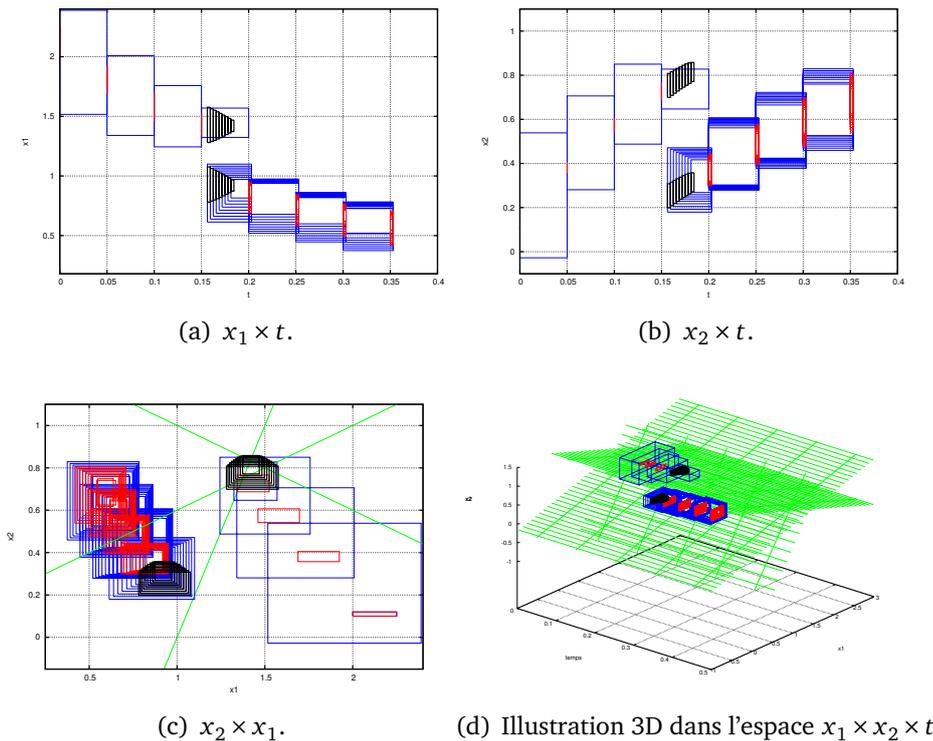


FIGURE 4.6 – Illustration de la méthode de passage ensembliste d’une garde de type polytope, Temps CPU=0.108s

Cette méthode séquentielle a donc tendance à accumuler le pessimisme. Nous avons vu que cette méthode (méthode 3) introduit des sur-approximations assez conservatrices par rapport aux autres méthodes de passage ensembliste des gardes (méthode 1 et 2). Donc a priori, la méthode de passage ensembliste de garde de type polytopique ne donne pas de résultats satisfaisants.

Comme alternative à l’approche proposée précédemment, nous pourrions utiliser l’approche proposée dans (Le, 2012), pour calculer directement l’intersection d’un zonotope et d’un polytope formé de l’intersection de n strips.

Proposition 5 *Étant donné un zonotope $Z = c \oplus RB^p \in \mathbb{R}^n$ et un polytope*

$$P_0 = \left\{ x \in \mathbb{R}^n : |\eta^\top x - d| < \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_n \end{bmatrix} \mid d \in \mathbb{R}^n, \sigma_i \in \mathbb{R}^+, i = 1, \dots, m \right\}$$

et une matrice $\Lambda \in \mathbb{R}^{n \times m}$. Le zonotope englobant l’intersection entre le zonotope Z et le polytope P_0 est donné par

$$Z \cap P_0 \subseteq Z_\cap = \hat{c}(\Lambda) \oplus \hat{R}(\Lambda) \mathbf{B}^{p+m}. \quad (4.33)$$

avec

$$\begin{aligned} \hat{c}(\Lambda) &= c + \Lambda(d - \eta c), \\ \hat{R}(\Lambda) &= [(I - \Lambda\eta)R \quad \Lambda\Sigma], \\ \Sigma = \text{diag}(\sigma_1, \dots, \sigma_m) &= \begin{bmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \sigma_m \end{bmatrix} \in \mathbb{R}^{m \times m} \end{aligned}$$

La matrice Λ peut être calculée en utilisant soit en minimisant de la longueur des segments, le volume, ou le P -radius (Le, 2012; Le et al., 2013) du zonotope que nous verrons dans la section 4.4.

Nous pouvons remarquer que les algorithmes de passage ensembliste de gardes proposés, caractérisent la solution à chaque tranche temporelle, finissent par générer un ensemble de tubes de trajectoires dans le nouveau mode. Ces trajectoires se chevauchent et pourraient naturellement être regroupées. Cette opération de «recollage» des boîtes doit se faire de manière précautionneuse afin de ne pas introduire de sur-approximation. C'est le but de la section 4.4, mais auparavant nous allons proposer une méthode permettant de mieux caractériser l'image d'un zonotope (MSBP) par une fonction linéaire ou non-linéaire.

4.3 MÉTHODE ENSEMBLISTE POUR LE SAUT DISCRET

Le calcul de l'image d'un ensemble par une fonction qui peut être non linéaire est traité dans cette section. Nous allons élaborer une expression analytique de la fonction de reset en fonction de la représentation zonotopique (MSBP). Cela nous permettra de réduire la sur-approximation introduite lors du saut car en utilisant la représentation dans le repère original, celui des boîtes, nous introduisons des sur-approximations qui peuvent être néfastes aux performances de notre algorithme d'atteignabilité.

Considérons une boîte solution sous la forme d'une MSBP \mathcal{X}_ℓ^* , $\ell \in 1, \dots, L$, parmi celles définies dans (4.2). Après le reset, la dynamique continue a pour condition initiale $\mathcal{X}_\ell^{*'} = \rho_e(\mathcal{X}_\ell^*)$. Comme notre outil d'atteignabilité continue fonctionne avec des ensembles caractérisés comme des MSBP, nous avons donc besoin de caractériser $\mathcal{X}_\ell^{*'}$ par une MSBP. L'image d'un zonotope par une fonction linéaire est un zonotope qui peut être obtenu par un produit matriciel standard. Par contre l'image d'un zonotope par une fonction non linéaire n'est pas, en général, un zonotope. Cependant, en utilisant les méthodes décrites dans (Alamo et al., 2005; Kühn, 1998; Raimondo et al., 2011) et rappelés dans la section 2.3.1.3, on peut calculer le plus étroit zonotope contenant $\mathcal{X}_\ell^{*'}$.

4.3.1 FONCTION DE RESET LINÉAIRE

Proposition 6 *Le et al. (2013) Étant donné une boîte solution sous la forme d'une MSBP $\mathcal{X}^* =$*

$A[r] \oplus [v]$, l'image $\mathcal{Z}^{*'} = \rho(\mathcal{Z}^*)$ par une fonction linéaire⁷ $\rho(\mathcal{Z}) = L.\mathcal{Z}$ (L est une matrice) peut être calculée par un produit matriciel standard. En effet :

$$\mathcal{Z}^{*'} \subseteq A'[r]' \oplus [v]' \quad (4.34)$$

où

$$A' = \tilde{K}(:, 1 : n) \in \mathbb{R}^{n \times n}, \quad (4.35)$$

$$[r]' = \mathbf{B}^n \subseteq \mathbb{R}^n, \quad (4.36)$$

$$[v]' = \rho_e(c) + \square(\tilde{K}(:, n+1 : 2n)\mathbf{B}^{2n}) \subseteq \mathbb{R}^n, \quad (4.37)$$

où $\square(\cdot)$ désigne l'enveloppe convexe d'un ensemble. La matrice \tilde{K} est définie par

$$\tilde{K} = \text{sort}(K), \quad K = [L.\mathcal{Z}] \in \mathbb{R}^{n \times 2n}, \quad (4.38)$$

où $\text{sort}(\cdot)$ est l'opérateur qui trie les colonnes de la matrice suivant leur norme, le vecteur c est défini dans (2.35), avec $m = 2n$. Ici $\tilde{K}(:, 1 : n)$ désigne les n premiers vecteurs colonnes de la matrice \tilde{K} , $\tilde{K}(:, n+1 : 2n)$ les $2n$ derniers vecteurs colonnes de la matrice \tilde{K} et $\rho_e(c) = L.c$

4.3.2 FONCTION DE RESET NON LINÉAIRE

Théorème 4 *Étant donné une boîte solution sous la forme d'une MSBP $\mathcal{Z}^* = A[r] + [v]$, l'image $\mathcal{Z}^{*'} = \rho(\mathcal{Z}^*)$ par une fonction non linéaire $\rho(\mathcal{Z}) = f(\mathcal{Z})$ (f est une application non linéaire) est contenue dans une MSBP calculée comme suit*

$$\mathcal{Z}^{*'} \subseteq A'[r]' \oplus [v]' \quad (4.39)$$

où

$$A' = \tilde{J}(:, 1 : n) \in \mathbb{R}^{n \times n}, \quad (4.40)$$

$$[r]' = \mathbf{B}^n \subseteq \mathbb{R}^n, \quad (4.41)$$

$$[v]' = \rho_e(c) + \square(\tilde{J}(:, n+1 : 3n)\mathbf{B}^{2n}) \subseteq \mathbb{R}^n, \quad (4.42)$$

où $\square(\cdot)$ désigne l'enveloppe convexe d'un ensemble. La matrice \tilde{J} est définie par

$$\tilde{J} = \text{sort}(J), \quad J = [\text{mid}([\nabla \rho_e]R) | G] \in \mathbb{R}^{n \times 3n}, \quad (4.43)$$

où l'opérateur $\text{sort}(\cdot)$ trie les colonnes de la matrice suivant leur norme, le vecteur c est défini dans (2.35), la matrice $R \in \mathbb{R}^{n \times 2n}$ définie par (2.36) avec $m = 2n$. Ici $\tilde{J}(:, 1 : n)$ désigne les n premiers vecteurs colonnes de la matrice \tilde{J} , $\tilde{J}(:, n+1 : 3n)$ les $2n$ derniers vecteurs colonnes de la matrice \tilde{J} , et $\rho_e(c) = f(c)$, $\nabla \rho_e = \nabla f_e$.

7. les zonotopes sont clos par transformation linéaire

Preuve 3 A partir de (4.39), nous avons $\mathcal{Z}^* = A[r] + [v] = c \oplus RB^{2n}$. En utilisant le théorème 2 de la section 2.3.1.3 (page 29), nous avons

$$\rho_e(\mathcal{Z}^*) = \rho_e(c) \oplus \diamond([\nabla \rho_e(\mathcal{Z}^*)R]B^{2n}),$$

alors d'après le théorème 1 de la section 2.3.1.3 (page 29),

$$\rho_e(\mathcal{Z}^*) = \rho_e(c) \oplus JB^{3n},$$

où J est défini par (4.43). Après avoir trié les vecteurs colonnes de J suivant leur norme, on obtient \tilde{J} . Il devient simple de trouver (4.39)-(4.43).

Remarque 3 On notera que dans l'algorithme 11 (Hybrid-Transition) le test d'inclusion à la ligne 23 consiste à vérifier que la condition de garde est satisfaite pour le vecteur d'état $z \in [\tilde{z}]$, où $[\tilde{z}]$ est le tube de trajectoires calculé sur l'intervalle de temps $[\underline{t}^*, \bar{t}^*]$. Une conséquence intéressante est que le tube de trajectoires n'a pas besoin de franchir complètement la condition de garde sur le dernier intervalle de temps afin d'activer le saut discret en utilisant la fonction de reset ρ .

4.4 MÉTHODES POUR LA FUSION DES TRAJECTOIRES

La détermination de l'enveloppe minimale d'un ensemble de points en dimension n (car les domaines de solution sont représentés par l'ensemble de leurs sommets) est traité dans cette section. Il existe de nombreux travaux sur la recherche d'enveloppe convexe d'un ensemble de points utilisant les parallélotopes en dimensions deux ou trois (Barequet and Har-Peled, 2001; O'Rourke, 1985; Vivien and Wicker, 2004). En dimension n , des ellipsoïdes englobants sont utilisées, mais il n'existe pas de travaux qui calculent un parallélotope avec un volume minimal (Althoff, 2010).

Cependant, des méthodes utilisant des zonotopes englobants, optimaux en terme de somme totale de la longueur des générateurs, est présentée dans (Stursberg and Krogh, 2003b). Le calcul d'une boîte orientée (ORH⁸) en dimensions n est calculé sur la base des orientations définies par une décomposition en valeurs singulières des échantillons des matrices de covariance des ensembles d'états atteignables.

Dans ce mémoire, nous utiliserons une approche basée sur une contraction itérative de l'ensemble des points pour construire un domaine orienté englobant sous une forme de générateurs (c'est-à-dire les zonotopes qui ne sont ni des hyperrectangles, ni des parallélotopes). Notre contribution dans cette section est de fournir une méthode pour fusionner les domaines de solutions de l'ensemble atteignable à un instant donné. Comme nous l'avons indiqué dans la section précédente, la bisection génère lors de la phase de détection et la localisation de l'événement,

8. Oriented Hyperrectangular Hull

plusieurs boîtes solutions, et après le reset ensembliste, elles sont les états initiaux de la transition continue dans le nouveau mode hybride (Maïga et al., 2014). Même si nos algorithmes de passage ensembliste des gardes ont de bonnes performances quand à la durée de franchissement des gardes, le calcul de tout ces tubes augmente indésirablement les temps de calcul.

Notre solution pour résoudre efficacement ce problème consiste donc à fusionner ces tubes en un seul tube englobant leur union. Notre algorithme de transition continue, c'est-à-dire l'algorithme $\varphi^{QR}(\cdot)$ (**algorithme 3**, page 32) utilise une MSBP, une forme compatible avec la représentation interne de notre intégrateur d'équations différentielles. Par conséquent, nous avons besoin de calculer l'union de ces tubes sous la forme d'une MSBP.

La méthode que nous proposons pour la fusion de l'ensemble des tubes se déroule en trois étapes :

- Premièrement, nous calculons les sommets du domaine convexe contenant l'union des MSBP décrivant les tubes ;
- Puis, nous calculons une MSBP paramétrée englobant ces sommets ;
- Finalement, nous réglons la MSBP obtenue dans le but d'optimiser sa taille.

Ces étapes sont décrites dans la suite.

4.4.1 CALCUL DES SOMMETS DES SOUS-TUBES

Nous supposons que la transition hybride est complètement achevée et que le domaine initial pour la transition continue dans le nouveau mode est caractérisée par l'union de MSBP suivante

$$\bigcup_{\ell=1}^L \mathcal{X}_\ell^{*'} \quad (4.44)$$

Chacun des l domaines de solution MSBP $\mathcal{X}_\ell^{*'}, l = 1$ à L , obtenu via (4.34) ou (4.39), est caractérisé comme :

$$\mathcal{X}_\ell^{*'} = \{A_\ell r + v \mid v \in [v]_\ell, r \in [r]_\ell\} = A_\ell [r]_\ell \oplus [v]_\ell = c_\ell \oplus Z(R_\ell), \quad (4.45)$$

où $Z(R_\ell)$ est un zonotope centré généré par la matrice $R_\ell \in \mathbb{R}^{n \times 2n}$, où c_ℓ et R_ℓ sont définis à partir de $[v]_\ell$ et $[r]_\ell$ comme dans (2.35)-(2.36).

Définition 2 Soit $R \in \mathbb{R}^{n \times p}$ une matrice et soit $Z(R) = RB^p \subset \mathbb{R}^n$ un zonotope centré généré par R . Nous dénotons par $Z^\pm(R)$, l'ensemble des vecteurs de points obtenus comme une image linéaire par R de 2^p sommets d'un cube unitaire B^p :

$$Z^\pm(R) = \{R\sigma, \sigma \in \{-1, 1\}^p\}. \quad (4.46)$$

Propriété 1 (Propriétés de contenance) Soit $\mathcal{C}(\cdot)$ (respectivement $\mathcal{V}(\cdot)$) l'enveloppe convexe (resp. les sommets) de n'importe quel polytope. Pour tout $R \in \mathbb{R}^{n \times p}$, nous avons

$$\mathcal{V}(Z(R)) \subseteq Z^\pm(R), \quad (4.47)$$

$$\mathcal{C}(Z^\pm(R)) = Z(R), \quad (4.48)$$

$$p = 2n \Rightarrow \exists R_g \in \mathbb{R}^{n \times p},$$

$$2\text{card}(\mathcal{V}(Z(R))) \leq \text{card}(\mathcal{V}(Z^\pm(R))) \leq 2\text{card}(\mathcal{V}(Z(R_g))) \quad (4.49)$$

$Z^\pm(R)$ contient tous les sommets de $Z(R)$, $Z(R)$ est l'enveloppe convexe de $Z^\pm(R)$, et $Z^\pm(R)$ contient seulement 2 fois plus de points que le vrai nombre de sommets d'une MSBP (c'est-à-dire $Z(R_g)$ ⁹ avec $p = 2n$). Le cas général donne des égalités au lieu des inégalités dans (4.49).

Preuve 4 Les propriétés (4.47)-(4.48) découlent directement de la définition 2 et la convexité des domaines zonotopiques. (4.49) résulte du nombre maximum de sommets d'un p -zonotope dans \mathbb{R}^n , $2 \sum_{i=0}^{n-1} \binom{p-1}{i}$, égale à $2^{(2n-1)}$ lorsqu'elle est appliquée à une MSBP ($p = 2n$). Par comparaison, le nombre potentiel des sommets d'un MSBP $\text{card}(Z^\pm(R))$ dans \mathbb{R}^n est $2^p = 2^{2n}$, est ainsi deux fois plus que le nombre réel de sommets d'une MSBP générique.

La propriété suivante formalise l'approche sur laquelle repose notre méthode pour la fusion de l'ensemble des tubes.

Théorème 5 Soit \mathcal{M} un ensemble convexe (par exemple un polytope, un zonotope, une MSBP etc.).

Considérons une MSBP $\mathcal{Z}_\ell^{*'} = c_\ell \oplus Z(R_\ell)$ dans (4.45) et nous appelons (2.35)-(2.36).

Soit \mathcal{P} un ensemble de vecteurs obtenu comme suit :

$$\mathcal{P} = \bigcup_{\ell=1}^L (c_\ell + Z^\pm(R_\ell)) \quad (4.50)$$

Donc,

$$\mathcal{P} \subseteq \mathcal{M} \Rightarrow \left(\bigcup_{\ell=1}^L \mathcal{Z}_\ell^{*'} \right) \subseteq \mathcal{M} \quad (4.51)$$

En d'autres termes, tout ensemble convexe \mathcal{M} contient les vecteurs dans \mathcal{P} , contient aussi l'union des domaines des MSBP $\mathcal{Z}_\ell^{*'}$ pour chaque ℓ , $\ell = 1, \dots, L$.

9. R_g désigne la matrice génératrice d'une MSBP quelconque lorsque $p=2n$ générateurs

Preuve 5 La preuve du théorème 5 repose principalement sur la nature polytopique (donc convexe) du domaine d'une MSBP $\mathcal{Z}_\ell^{*l} = A_\ell[r_\ell] + [v_\ell] = c_\ell \oplus Z(R_\ell)$ où $c_\ell \in \mathbb{R}^n$ et $R_\ell \in \mathbb{R}^{n \times 2n}$. (4.50) ne fournit pas seulement un moyen constructif pour calculer les $m2^{2n}$ vecteurs dans \mathcal{P} , mais assure aussi à travers (4.47) que \mathcal{P} contient l'ensemble \mathcal{V} de tous les sommets de tous les domaines des MSBP \mathcal{Z}_ℓ^{*l} , $l = 1, \dots, L$. En plus, (4.48) assure que tout ensemble convexe \mathcal{M} satisfaisant $\mathcal{P} \subseteq \mathcal{M}$ satisfait aussi la propriété de contenance $\mathcal{Z}_\ell^{*l} \subseteq \mathcal{M}$ pour chaque $l = 1, \dots, L$.

Remarque 4 Comme $Z^\pm([R_1, R_2]) = Z^\pm(R_1) \oplus Z^\pm(R_2)$, l'ensemble des vecteurs \mathcal{P} définie dans (4.50) peut être aussi exprimé comme $\mathcal{P} = \bigcup_{l=1}^L (A_\ell[r]^\pm \oplus [v]^\pm)$ où $[r]^\pm$ et $[v]^\pm$ se réfèrent à l'ensemble des sommets des boîtes alignées avec les axes de dimension n , $[r]_\ell$ et $[v]_\ell$, respectivement. Pour une valeur donnée de l , la figure 4.7 illustre comment l'ensemble des points $c_\ell + Z^\pm(R_\ell)$ peut être construit à partir de $A_\ell[r]^\pm \oplus [v]^\pm$.

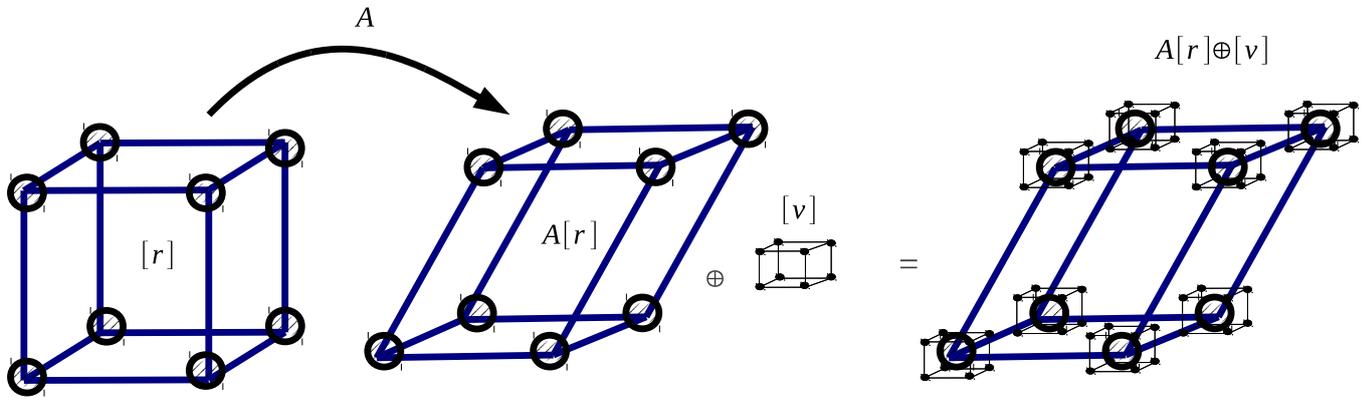


FIGURE 4.7 – Construction de l'ensemble des vecteurs $c_\ell + Z^\pm(R_\ell)$

4.4.2 ENVELOPPE D'UN NUAGE DE POINTS PAR UN ZONOTOPE

Cette sous-section décrit notre algorithme générique `cloud2zonotope` (**algorithme 10**) pour le calcul du centre $c \in \mathbb{R}^n$ et la matrice $R \in \mathbb{R}^{n \times p}$ donnant la forme du zonotope $c \oplus Z(R)$ englobant le nuage de points X de N vecteurs définis dans \mathbb{R}^n . Pour une meilleure lisibilité de l'algorithme, les vecteurs sont explicitement notés en utilisant la notation \vec{u} dans nos algorithmes. L'algorithme `cloud2zonotope` (**algorithme 10**), illustré aussi dans la figure 4.8, est principalement basé sur une contraction itérative du nuage X formé par N vecteurs et sur une construction itérative du zonotope englobant. Après avoir centré le nuage (ligne 5) au milieu de l'enveloppe convexe du nuage, chaque contraction (ligne 9) est réalisée selon le vecteur $\vec{g} \in \mathbb{R}^n$ qui définit l'un des segments de générateur du zonotope résultant (ligne 10). A chaque itération, \vec{g} est orienté le long de la direction principale \vec{u} du nuage courant (c'est-à-dire partiellement compressé) (ligne 7). En outre, l'amplitude de \vec{g} est une fraction de (une approximation de) la plus grande projection des points dans le nuage (centrée) le long de \vec{u} . La fraction utilisée est définie par le taux de compression, noté *ratio* ($ratio \in [0, 1]$), l'approximation de la plus grande

projection reposant sur un calcul très rapide basé sur l'enveloppe convexe du nuage courant qui correspond au produit scalaire $\vec{u} \cdot \vec{rad}$ impliquant seulement deux vecteurs de dimension n à la ligne 8.

Chaque itération de la boucle "while" dans `cloud2zonotope` effectue simultanément une compression du nuage, à la fois dans la direction et l'amplitude, comme définie par le générateur \vec{g} , et une expansion du zonotope sous construction, qui repose sur la somme de Minkowski avec le segment de droite $[-1, +1]\vec{g}$, à la ligne 10. Par ailleurs, la mise à jour du nuage $\underline{X} = \text{compress}(X, \vec{g})$ à partir de la compression de X dans la direction \vec{g} , décrit dans l'algorithme 9, est conçue de manière à satisfaire

$$X \subset (\underline{X} \oplus \vec{g}[-1, +1]), \quad (4.52)$$

où les matrices X et \underline{X} correspondent chacune à l'ensemble des N vecteurs. Notons que (4.52) est un point essentiel pour assurer que le nuage initial soit éventuellement englobé dans le zonotope obtenu par l'algorithme `cloud2zonotope`.

Le processus itératif "compression du nuage et expansion du zonotope" est répété jusqu'à ce que l'un des deux critères d'arrêt s_1 ou s_2 soit satisfait (ligne 11) : s_1 est un entier définissant le nombre maximum d'itérations autorisées, et $s_2 \in [0, 1]$ est un réel positif définissant une condition d'arrêt de la compression itérative comme la fraction (ratio) du nuage initial (\vec{w} assigné à la ligne 6). A la sortie de la boucle "while", l'enveloppe convexe du nuage résiduel est calculé (ligne 13) et la boîte alignée correspondante est additionnée (Minkowski sum) avec le zonotope précédemment obtenu à partir d'un ensemble initialement vide pendant les itérations de la boucle while. Le résultat de la ligne 14 se traduit par un zonotope $c \oplus Z(R)$ qui garantit contenir tous les N vecteurs définis par X .

La figure 4.8 illustre les étapes principales de l'algorithme 10 `cloud2zonotope` dans le cas d'un nuage de points à 2-dimensions et la MSBP obtenue. La figure 4.8.a montre que l'algorithme centre d'abord le nuage de points, ensuite contracte le nuage le long de la première direction \vec{u}_1 . Ici on peut voir le vecteur \vec{z}_1 et \vec{z}_2 projeté sur les axes \vec{u}_1 tout en considérant les bornes définies par *ratio*. Donc, contracter le nuage revient à garder uniquement la différence $\vec{z}_i^1 = \vec{z}_i - \vec{z}_i^g$. La figure 4.8.b montre le nuage après la première contraction le long de la direction \vec{u}_1 . Notre algorithme trouve la deuxième direction \vec{u}_2 de contraction du nuage résiduel. Dans le but d'obtenir une MSBP comme sortie de l'algorithme 10 `cloud2zonotope`, le nombre d'itérations de compression est égale à $n = 2$ (dimension de l'espace d'état pour cet exemple) ; La figure 4.8.c montre qu'après 2 itérations de compression, les vecteurs résiduels sont englobés dans une boîte alignée avec les axes. La figure 4.8.d montre le zonotope, ici MSBP, obtenu comme la somme de Minkowski de la boîte alignée et d'un parallélogramme défini par les segments générateurs $\vec{g}_1 = \|\vec{g}_1\|\vec{u}_1$ et $\vec{g}_2 = \|\vec{g}_2\|\vec{u}_2$.

Algorithme 8 : Algorithm cloud2zonotope

Input : $X := \{\vec{x}_i, i = 1 \dots N\}$, $ratio \in [0, 1]$, s_1, s_2
Output : $\vec{c} \in \mathbb{R}^n$, $R \in \mathbb{R}^{n \times p}$

- 1 Initialization : $\vec{c} = 0$, $R = \emptyset$, $iter = 0$, $i = 0$, $r\vec{a}d_0 = 0$, $\vec{w} = 0$;
- 2 **while** ($iter < s_1$) \vee ($\exists i r\vec{a}d_i > s_2 \vec{w}_i$) **do**
- 3 update iteration counter. $iter := iter + 1$;
- 4 compute cloud center vector and cloud radius vector. $\{\vec{m}id, r\vec{a}d\} := \text{minbox}(X)$;
- 5 center point-cloud around mid . $X := X - \vec{m}id$;
- 6 store initial radius. **if** $iter = 1$ **then** $\vec{w} := r\vec{a}d$;
- 7 find first principal direction (unit vector \vec{u}). $(U, S) := \text{svd}(X)$, $\vec{u} := U(:, 1)$;
- 8 choose generator segment. $\vec{g} := ratio |\vec{u} \cdot r\vec{a}d| \cdot \vec{u}$;
- 9 compress cloud along \vec{g} using $ratio$. $X := \text{compress}(X, \vec{g})$;
- 10 update zonotope generators. $\vec{c} := \vec{c} + \vec{m}id$, $R := [R, \vec{g}]$;
- 11 **end while**
- 12 enclose remaining cloud in a box. $\{\vec{m}id, \vec{r}ad\} := \text{minbox}(X)$;
- 13 update zonotope. $\vec{c} := \vec{c} + \vec{m}id$, $R := [R, \text{diag}(r\vec{a}d)]$;

Algorithme 9 : Algorithm compress

- 1 **Function** $X = \text{compress}(X, \vec{g})$
- 2 compute compression direction (unit vector). $\vec{u} = \vec{g} / \|\vec{g}\|$;
- 3 **for each** point-vector \vec{x}_i **in** X **do**
- 4 compute coordinates along \vec{u} . $d := \vec{x}_i \cdot \vec{u}$;
- 5 compression along \vec{u} . $d := \min(\|\vec{g}\|, \max(-\|\vec{g}\|, d))$;
- 6 update coordinates. $\vec{x}_i = \vec{x}_i - d\vec{u}$;
- 7 **end for**
- 8 **return**

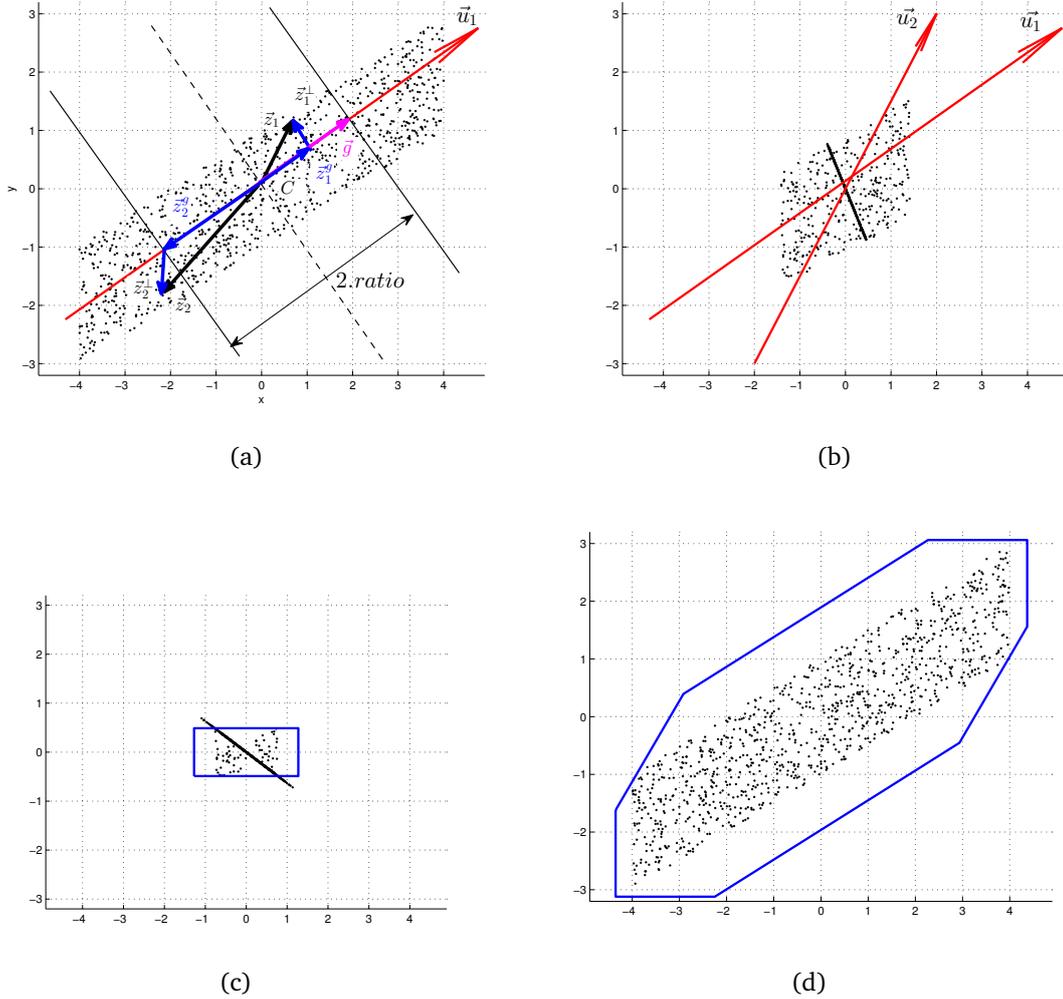


FIGURE 4.8 – Étapes principales pour construire le zonotope englobant réalisée par l’algorithme10 `cloud2zonotope` dans le cas d’un nuage de point en 2D en entrée et une MSBP en sortie , (a) centrage du nuage de points et compression le long de la première direction principale, (b) trouver la seconde direction de compression comme la première direction principale du nuage restant, (c) englober le reste du nuage dans une boîte alignée avec les axes, (d) calcul de la MSBP englobant le nuage de points.

4.4.3 CONSTRUCTION D’UNE MSBP À PARTIR DU ZONOTOPE ENGLOBANT

En utilisant l’algorithme `cloud2zonotope`, on peut régler les paramètres $ratio$ et $s_i, i = 1, 2$ introduit dans la section précédente dans le but d’obtenir un zonotope particulier dont le centre $c_{\mathcal{M}}$ et la matrice de forme du zonotope $R_{\mathcal{M}} \in \mathbb{R}^{n \times 2n}$ représentent la MSBP minimale, au sens d’un critère à définir, englobant le nuage de points \mathcal{P} défini dans (4.50).

En choisissant $s_1=n$ et $s_2=0$, l'algorithme `cloud2zonotope` fournit une MSBP car il y a exactement n itérations de compression, donc exactement n vecteurs générateurs. En plus, le nuage résiduel est englobé dans une boîte alignée avec les axes. La taille de la MSBP dépend uniquement du paramètre $ratio$ qui peut être réglé pour optimiser un critère fondé sur la taille. Celui-ci agit comme un réglage qui permet d'avoir un compromis entre l'importance relative du parallélotope et de la boîte alignée avec les axes (2.33). En effet en choisissant $ratio=0$, on obtient une boîte alignée avec les axes qui est une MSBP particulière, et pour $ratio=1$, on obtient un parallélotope (boîte orientée). Le choix optimal pour le paramètre $ratio$ est ainsi formalisé par

$$ratio^* = \operatorname{argmin}_{ratio \in [0,1]} \mu(Z(R)) \quad (4.53)$$

où $\mu(\cdot)$ est la taille du zonotope obtenue de trois manières : le volume, la longueur des segments, ou le P -radius.

Pour obtenir la solution $ratio^*$, on évaluera seulement la taille de la MSBP pour dix valeurs de $ratio$ prises sur une grille de valeurs entre 0 et 1, puis on prendra simplement la valeur qui minimise (4.53).

Pour résumer, l'union des ℓ ($\ell=1, \dots, L$) domaines de solution, MSBP, définis par (4.44) est étroitement enveloppée dans un seule MSBP avec une taille optimisée. Elle est obtenue avec l'algorithme `cloud2zonotope` paramétré avec $ratio=ratio^*$, $s_1=n$ et $s_2=0$.

LE CRITÈRE DU VOLUME

Le volume d'un zonotope (Schneider, 1994; Ziegler, 1995) $Z(R) \subset \mathbb{R}^n$, où $R = [r_1, \dots, r_p] \in \mathbb{R}^{n \times p}$, avec $p \geq n$, est donné :

$$Vol(Z(R)) = 2^n \left(\sum_{1 \leq i_1 \leq i_2 \leq \dots \leq i_n \leq p} |\det([r_{i_1}, \dots, r_{i_n}])| \right) \quad (4.54)$$

Les entiers i_1, \dots, i_n correspondent aux différentes manières de choisir n éléments parmi p . La somme (4.54) est ainsi composée de $\binom{p}{n}$ éléments.

LE CRITÈRE LONGUEUR DES SEGMENTS

La longueur des segments d'un zonotope $Z = c \oplus RB^p \in \mathbb{R}^n$ est donné par la somme des normes carrées des générateurs du zonotope Z . Le calcul de la somme des normes carrées des générateurs du zonotope Z revient donc au calcul de la norme de *Frobenius* de R (Lalami, 2008; Le, 2012).

$$S_m(Z(R)) = \sum_{i=1}^n \sum_{j=1}^p R_{ij}^2 = \|R\|_F^2 = \operatorname{Tr}(R^T R) \quad (4.55)$$

où Tr dénote la trace d'une matrice carrée.

LE CRITÈRE *P-radius*

Le *P-radius* (Le et al., 2013) d'un zonotope $Z = c \oplus RB^p \in \mathbb{R}^n$ est défini comme

$$P_{rad}(Z(R)) = \max_{x \in Z} \|x - c\|_P^2 = \max_{x \in Z} (x - c)^\top P (x - c) \quad (4.56)$$

où $P = P^\top \geq 0$ est une matrice symétrique définie positive.

4.4.4 ANALYSE DE COMPLEXITÉ

Notre méthode d'atteignabilité combine l'utilisation de la méthode de Taylor intervalle implémentant la factorisation QR de Lohner pour la transition continue du système hybride et une méthode originale pour le passage des gardes matérialisé par des transitions discrètes. La méthode de Taylor intervalle sous-jacente pour l'atteignabilité continue est de complexité polynomiale. En notant k l'ordre du développement en série de Taylor, généralement choisi entre 10 et 20, le calcul des coefficients de Taylor à chaque itération est $O(k^2)$.

La complexité de calcul de la méthode d'atteignabilité est donc liée principalement au passage des gardes, qui inclut deux étapes principales : l'intersection du tube de trajectoires de la garde, résolu sous forme d'un CSP, et la fusion des trajectoires, qui est résolue par l'**algorithme 10** (page 80). Il est admis que la résolution de CSP, que ce soit sur des domaines discrets ou continus, est en théorie NP-difficile. Cependant, de nombreux efforts ont été fournis pour développer des techniques de résolution dont le temps de calcul observé en pratique est meilleur que le pire des cas (exponentielle) (Tuy, 1995). Ces améliorations s'appuient sur des caractéristiques techniques qui ont été présentées dans ce mémoire : la relaxation garantie de la résolution par CSP de l'intersection du tube de trajectoires du flow et de la garde, la fusion des trajectoires qui permet de fusionner tous les tubes.

Dans la suite, nous discutons plus en détails de l'algorithme permettant de calculer l'union des MSBP. (Althoff et al., 2010) a proposé d'englober l'*union* de polytopes dans une boîte orientée, notre choix d'une MSBP est motivé par sa capacité à mieux représenter la forme de l'union d'un ensemble qu'une boîte orientée, tout en étant compatible avec notre algorithme d'intégration numérique utilisé pour gérer la dynamique continue non linéaire. L'**algorithme 10** permet de construire efficacement un zonotope en dimensions n (ainsi, en particulier, une MSBP) à partir d'un nuage de points (ici, les sommets potentiels). Le calcul de l'enveloppe MSBP implique donc la génération d'un nuage de points et n décompositions en valeurs singulières (SVD). La génération du nuage de points est $O(2^{2n})$, et chaque calcul de SVD est $O(Nn^2)$.

L'algorithme MSBP fusionne toutes les MSBP en une seule \mathcal{M} . Considérons un zonotope $Z = c \oplus RB^p \in \mathbb{R}^n$, où $R \in \mathbb{R}^{n \times p}$ et rappelons qu'une MSBP \mathcal{M} est un zonotope particulier avec $p = 2n$.

L'énumération des $P = 2^{2n}$ Sommets Potentiels (SP) d'une MSBP (**algorithme SP-MSBP**)¹⁰ dans \mathbb{R}^n peut être réalisée avec $fl_{SP-MSBP}(n, P) \approx nP$ additions, où $fl_{alg}(\cdot)$ est une estimation du nombre d'opérations à virgule flottante pour l'algorithme alg . A cet effet, il suffit d'effectuer une seule marche à travers les entrées successives d'un code binaire réfléchi (comme dans les cartes Karnaugh) représentant les sommets relatifs aux hypercubes de dimension $2n$ (c'est-à-dire $\mathbf{B}^{2n} = [-1, +1]^{2n}$). À chacune des étapes P de cette marche, puisque le signe est lié uniquement à l'un des $2n$ générateurs, seulement n additions sont requises pour calculer le futur sommet potentiel à partir du sommet courant. Le coût d'énumération des $N = mP = m2^{2n}$ sommets potentiels relatifs au m MSBP peut ainsi être évalué par $fl_{PV-Enum}(n, m, P) \approx mnP$.

Le coût de calcul de la SVD d'une matrice X ($X = USV^T$ c'est-à-dire $X^T = VS^T U^T$) de dimension $N \times n$ sans la nécessité d'un calcul explicite de $U \in \mathbb{R}^{N \times N}$ est $fl_{SVD}(n, N) \approx 4Nn^2 + 8n^3$, qui est linéaire en N (Golub and Loan, 1996). La complexité de l'algorithme `compression` peut être approximativement évalué par $fl_C(n, N) \approx 4Nn$ (c'est-à-dire $2Nn$ pour la ligne 4, $2Nn$ pour la ligne 6, et négligée pour la ligne 5)

La complexité d'une itération de la boucle `while` dans `cloud2zonotope` est principalement attribuable à l'appel de `compress` (ligne 9) et un appel à la SVD (ligne 7). Dénotons par q le nombre de compressions itérées durant l'exécution de `cloud2zonotope`, le coût de calcul lié à un tel appel est donc $fl_{C2Z}(n, N, q) \approx q(fl_{SVD}(n, N) + fl_C(n, N)) = q(4Nn^2 + 4Nn + 8n^3)$ avec $N = mP$. L'algorithme MSBP impose $q = n$ itérations lors de l'appel `cloud2zonotope`. Cependant, le coût de l'algorithme MSBP peut être évalué comme

$$fl_{MSBP}(n, m, P) \approx fl_{PV-Enum}(n, m, P) + 10fl_{C2Z}(n, mP, n) \approx mP(40n^3 + 40n^2 + n) + 80n^4$$

(le facteur 10 est pour l'optimisation basée sur une grille de taux de compression ratio). En pratique le coût de l'algorithme de fusion de trajectoires est donc linéaire en nombre m de MSBP à fusionner et est presque linéaire en le nombre maximal de sommets d'une unique MSBP (c'est-à-dire $P/2 = 2^{(2n-1)}$). L'énumération des sommets d'une MSBP dans \mathbb{R}^n peut être réalisée de deux manières différentes :

- (a) En passant par une représentation demi-espace pour énumérer les sommets MSBP (par exemple avec des algorithmes utilisés pour transformer les H-polytopes¹¹ en V-polytopes¹²), ce qui nécessiterait

$$fl_{H-MSBP}(n) \approx \frac{4}{3} \binom{2n}{n-1} (n-1)^3 n$$

opérations en virgule flottante sans tenir compte de la deuxième phase d'énumération de sommets.

- (b) Réalisation de l'enveloppe convexe des sommets potentiels d'une MSBP¹³.

10. Combinaison des algorithmes de génération des sommets (voir 4.50) et de l'algorithme de fusion de trajectoires (voir **algorithme 10**)

11. Un H-polytope est l'intersection d'un nombre fini de demi-espaces de \mathbb{R}^n

12. Un V-polytope est l'enveloppe convexe d'un nombre fini de points de \mathbb{R}^n

13. L'enveloppe convexe des $N = mP$ sommets potentiels de m MSBP pourrait également être envisagée pour caractériser directement l'union, mais le coût de calcul relatif devient très rapidement prohibitif quand n augmente.

Basé sur la complexité des algorithmes d'énumération des sommets (par exemple QuickHull (Barber et al., 1996)), on peut montrer qu'il y a un intérêt évident à utiliser les sommets potentiels plutôt que l'énumération de véritables sommets : le premier procédé induit seulement deux fois plus de points pour une seule MSBP générique et ce grand nombre de points est facilement équilibré par la simplicité de l'algorithme d'énumération (bien que la complexité reste bien sûr exponentielle dans les deux cas).

Globalement, notre algorithme d'atteignabilité hybride non linéaire est de complexité exponentielle.

Algorithme 10 : Algorithm MSBP

Input : \mathcal{L}
Output : \mathcal{L}

- 1 initialization : $\mathcal{P} = \emptyset$ i.e. empty set of points ;
- 2 **for** $j \leftarrow 1$ **to** $Length(\mathcal{L})$ **do**
- 3 pick up $\mathcal{X}_\ell^{*'}$ from \mathcal{L} (i.e. $q, A_\ell, [r]_\ell, [v]_\ell$);
- 4 $\mathcal{P} := \mathcal{P} \cup (A_\ell[r]_\ell^\pm \oplus [v]_\ell^\pm)$
- 5 **end for**
- 6 $X_{\mathcal{P}} :=$ matrix representation (n col.) of $\mathcal{P} \subset \mathbb{R}^n$;
- 7 **for** $ratio =: 0, 0.1, \dots, 0.9, 1$ **do**
- 8 $\{c_{\mathcal{M}}, R_{\mathcal{M}}\} := \text{cloud2zonotope}(X_{\mathcal{P}}, ratio, n, 0)$;
- 9 compute the size of zonotope $\mu(Z(R_{\mathcal{M}}))$ as in (4.54);
- 10 **end for**
- 11 $r^* := \text{argmin} \mu(Z(R_{\mathcal{M}}))$;
- 12 extract MSBP attributes of $\mathcal{M}^* := c_{\mathcal{M}^*} \oplus Z(R_{\mathcal{M}^*})$ obtained with r^* :
- 13 $A := R_{\mathcal{M}^*}(:, 1 : n)$;
- 14 $[r] := [-1, +1]^n$;
- 15 $[v] := c_{\mathcal{M}^*} + \text{diag}(R_{\mathcal{M}^*}(:, (n+1) : (2n)))[-1, +1]^n$;
- 16 store MSBP : $\mathcal{L} := (q, A, [r], [v])$ in list \mathcal{L} ;

4.5 ÉVALUATION DE LA MÉTHODE D'ATTEIGNABILITÉ HYBRIDE SUR DES EXEMPLES SIMPLES

4.5.1 EXEMPLE 1 : BALLE REBONDISSANTE

En modifiant le modèle de la balle rebondissante proposé initialement dans (Ishii et al., 2009), on obtient une balle qui rebondit sur un plan incliné avec une pente a (voir figure 4.9). Les états continus sont décrits par quatre variables (p_x, p_y, v_x, v_y) , où (p_x, p_y) , (v_x, v_y) représentent respectivement la position et la vitesse de la balle. Le pas de temps est choisi constant $h = 0.1$. L'intervalle de temps est bissecté jusqu'au seuil $\varepsilon_T = 0.005$. Les paramètres sont ré-

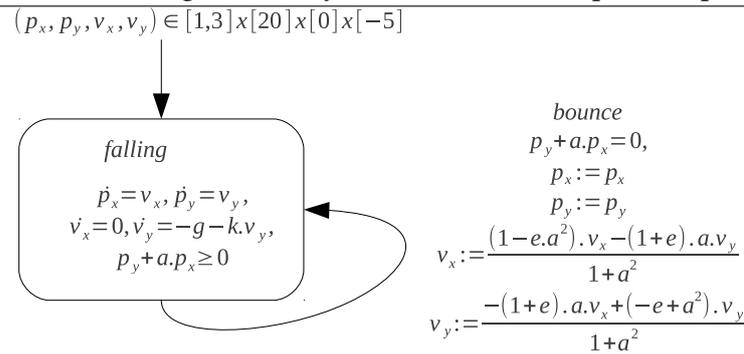


FIGURE 4.9 – Automate hybride de la balle rebondissante

glés à $g = 9.8, a = 0.25, e = 1.5$ et $k = 0.5$. Quand on utilise le contracteur sans la fusion des boîtes, les résultats obtenus sont donnés dans les figures (4.11(a),4.10(e)), tandis que les figures (4.11(b),4.10(f)) illustrent le cas où l'on utilise notre méthode de fusion des trajectoires.

Sur cet exemple, nous avons utilisé les trois critères pour trouver la valeur optimale de *ratio* (voir tableau 4.2), mais pour chacune de ces valeurs, on introduit une sur-approximation plus ou moins importante. La meilleure valeur de *ratio* pour cet exemple vaut 1 (voir figure 4.11(i)), c'est-à-dire les boîtes orientées sont plus favorisées que les boîtes alignées.

Cependant sur cet exemple nous trouvons des valeurs différentes de *ratio* pour les trois critères (0 pour le critère *P-radius*, 0.3 pour le critère minimisation de segments et le critère volume), la valeur optimale de *ratio* étant égale à 1.

En utilisant la méthode de fusion des trajectoires, nous arrivons à diviser les temps de calcul par 3 voir figure (4.11(a) et 4.11(b)).

 TABLE 4.2 – (*P-radius*, S_m , Volume) vs *ratio* pour la balle rebondissante

<i>ratio</i>	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
<i>P-radius</i>	4.26	5.19	6.15	5.87	7.37	8.01	8.59	9.83	9.76	10.16	11.07
S_m	4.26	2.10	1.56	1.33	1.58	2.03	2.56	3.25	3.98	4.86	5.95
volume $\times (10^{-3})$	62.96	24.48	13.05	4.35	14.04	5.52	0.29	26.88	37.78	43.63	50.99

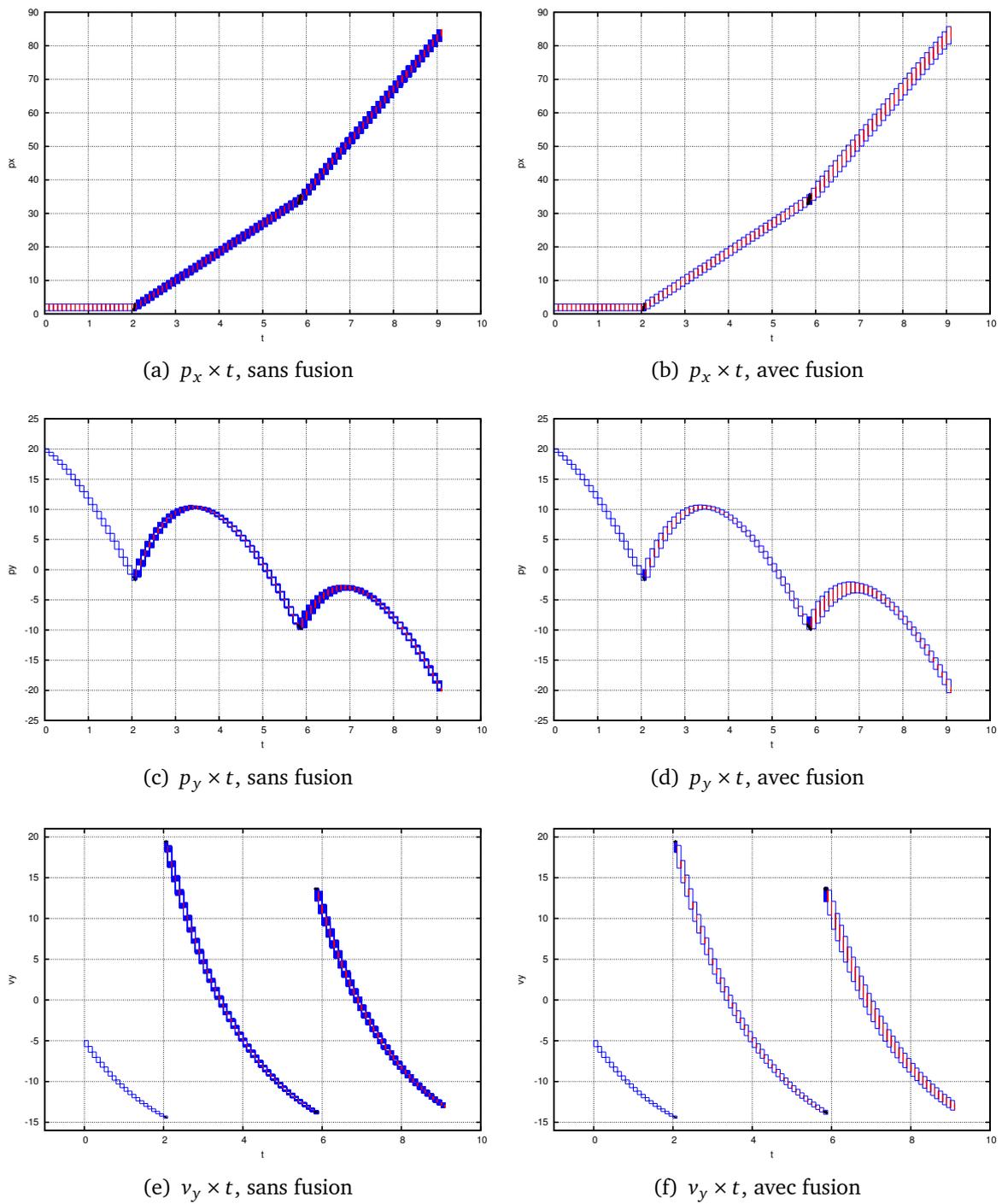
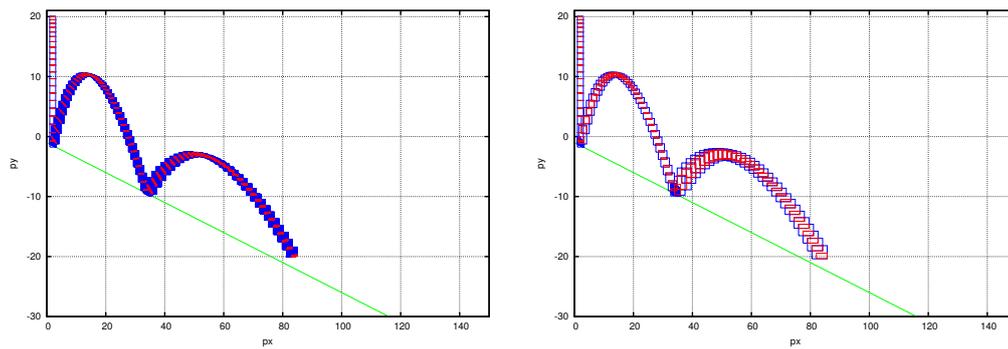
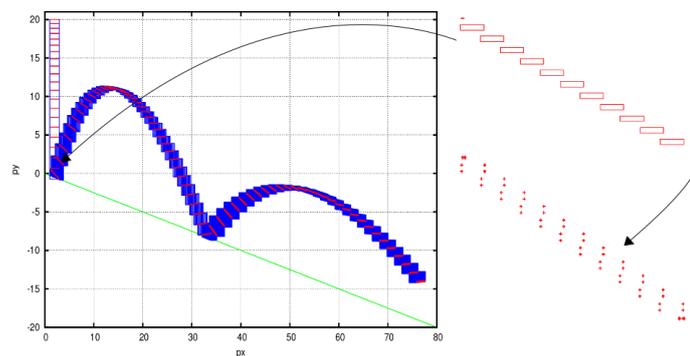


FIGURE 4.10 – L'ensemble atteignable obtenu pour la balle rebondissante. Historique des variables p_x, p_y, v_y en fonction du temps

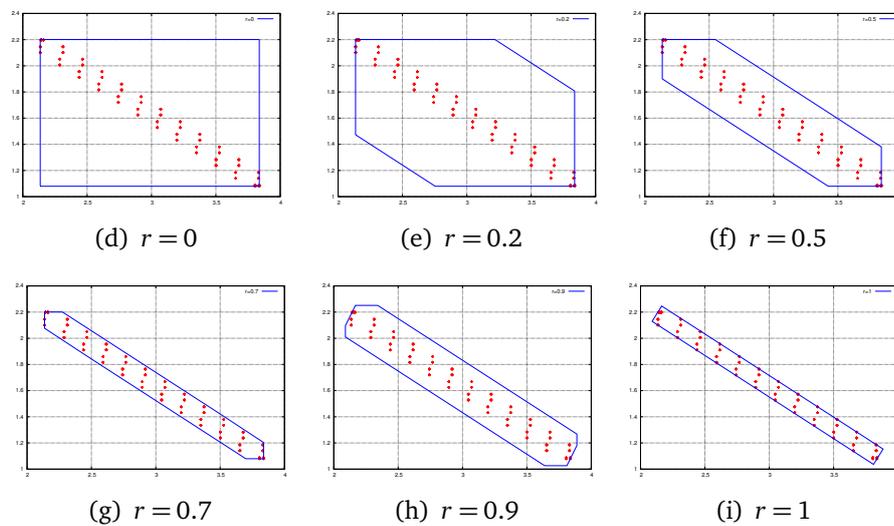


(a) CPU times = 0.98s

(b) CPU times = 0.34s



(c) Zoom sur les boîtes solution à $t = 2$



(d) $r = 0$

(e) $r = 0.2$

(f) $r = 0.5$

(g) $r = 0.7$

(h) $r = 0.9$

(i) $r = 1$

FIGURE 4.11 – (a-b) : L'ensemble atteignable obtenu pour la balle rebondissante. Projection dans l'espace $p_y \times p_x$, (a) : sans la fusion des trajectoires, et (b) : avec fusion des trajectoires, (c-d-e-f-g-h-i) : les zonotopes obtenus en 2D (projection dans dans l'espace $p_x \times p_y$ en fonction du ratio r à $t=2s$) pour la balle rebondissante

4.5.2 EXEMPLE 2 : SYSTÈME MASSE-RESSORT À COMMUTATION

Considérons le système dynamique à commutation avec deux modes et une transition $e = 1 \rightarrow 2$ obtenue en introduisant une commutation artificielle dans un système masse-ressort. La commutation est artificielle car la fonction de reset est l'identité. L'idée principale est d'étudier l'impact de notre méthode de passage de garde et notre algorithme de fusion de trajectoires en comparant les résultats avec le système continu original c'est-à-dire sans la commutation.

Le système de commutation est donné par

$$\left\{ \begin{array}{l} \text{flow}(1) : (\dot{x}_1, \dot{x}_2) = \left(x_2, \frac{-k}{m}x_1 - \frac{c}{m}x_2\right) \\ \text{inv}(1) : x_1 - x_2 < 0 \\ \text{flow}(2) : (\dot{x}_1, \dot{x}_2) = \left(x_2, \frac{-k}{m}x_1 - \frac{c}{m}x_2\right) \\ \text{inv}(2) : x_1 - x_2 > 0 \\ \text{guard}(1) := x_2 - x_1 = 0 \\ \text{reset}(1) : \rho_1(x_1, x_2) = L \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} \end{array} \right. \quad (4.57)$$

Avec $L = I_2^{14}$, $l_1 = l_2 = 0$, $k = 4$, $m = 2$, $c = 1.25$. Les états continus sont décrits par deux variables (x_1, x_2) , où x_1 , x_2 représentent respectivement la position et la vitesse de la masse. Les conditions initiales sont données par $x_1 \in [1, 1.1]$, $x_2 \in [-0.63, -0.61]$. Les algorithmes Transition-Hybride (**algorithme 11**, page 101) et φ^{QR} (voir **algorithme 3**, page 32) sont réglés comme suit : le pas de temps est choisi constant $h = 0.1$; et l'intervalle de temps est bisecté jusqu'à seuil $\varepsilon_T = 0.005$. Les figures 4.12 regroupent les ensembles atteignables obtenus

TABLE 4.3 – (P-radius, S_m , Volume) vs ratio pour le masse-ressort, première commutation

ratio	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
P-radius	1.910	1.910	1.910	1.910	1.915	1.929	1.926	1.929	1.936	1.946	1.962
$S_m \times (10^{-3})$	4.646	3.174	2.002	1.823	1.792	2.179	2.329	2.729	3.331	4.149	5.115
Volume $\times (10^{-3})$	8.574	3.417	1.104	1.961	3.122	4.974	5.511	2.445	3.239	4.745	6.577

TABLE 4.4 – (P-radius, S_m , Volume) vs ratio pour le masse-ressort, deuxième commutation

ratio	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
P-radius	0.710	0.710	0.710	0.710	0.710	0.710	0.710	0.710	0.713	0.725	0.737
$S_m \times (10^{-3})$	8.641	6.116	4.094	3.781	3.950	3.653	4.048	4.769	5.899	7.403	9.122
Volume $\times (10^{-3})$	16.186	7.444	1.295	3.290	6.273	8.838	9.979	5.088	6.383	9.413	13.053

pour un horizon temporel $[0, 5]$. En premier, l'ensemble atteignable pour le système continu original est montré (figure 4.12-a), puis l'ensemble atteignable pour la version à commutation avec la fusion des trajectoires (figure 4.12-b). Les ensembles atteignables obtenus avec notre

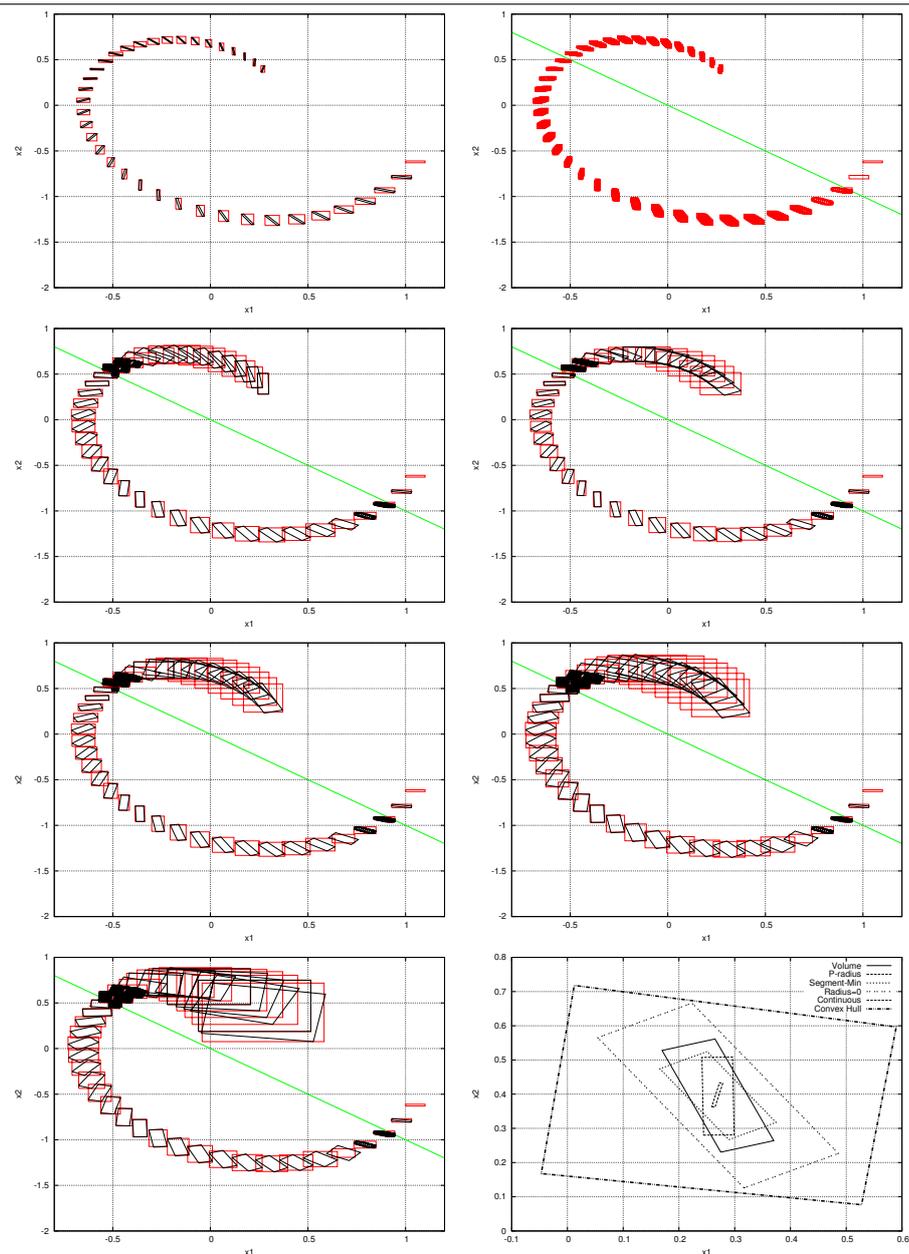


FIGURE 4.12 – L'ensemble atteignable du système masse-ressort. De gauche à droite, et de haut en bas, la version continue (CPU 0.028s), la version commutation sans la fusion de trajectoires (CPU 1.884s), la version commutation avec la fusion des trajectoires en utilisant le critère *P-radius* (CPU 0.196s), en utilisant le critère longueur des segments (CPU 0.164s), en utilisant le critère volume (CPU 0.2s), quand on utilise la fusion avec $\text{ratio}=0$ (CPU 0.308s), et la méthode naïve qui consiste à prendre l'enveloppe convexe des boîtes solutions. La dernière figure compare les MSBP obtenus au temps final pour les différentes expérimentations.

algorithme de fusion en utilisant les critères proposés dans la section 4.4.3 (page 76) : P -radius, avec P qui est prise comme une matrice identité, le volume et la longueur des segments. Les tableaux 4.3-4.4 montrent les critères obtenus pour la première et la deuxième commutation respectivement. Comme nous pouvons le remarquer, on n'a pas une monotonie des valeurs pour les différents critères. Ce qui explique amplement, pourquoi nous n'avons pas utilisé les méthodes d'optimisations (car on n'a pas un minimum global). Notre méthode qui consiste à considérer dix valeurs de *ratio* et prendre le minimum. Le paramètre de réglage optimal dépend bien évidemment du critère, bien qu'il ne semble pas changer de façon significative avec la commutation. La figure 4.12 en bas à droite montre clairement que le critère P -radius fournit une MSBP étroite, mais reste grande comparée à l'enveloppe calculé pour le système continu (c'est-à-dire sans commutation). Néanmoins, les MSBP obtenues avec notre méthode de fusion sont toujours nettement plus étroites que les enveloppes obtenues avec la méthode naïve de l'enveloppe convexe du tubes de trajectoires. Les figures 4.13 illustrent l'ensemble atteignable du système masse-ressort en 3D, sans et avec la fusion des trajectoires pour le critère P -radius.

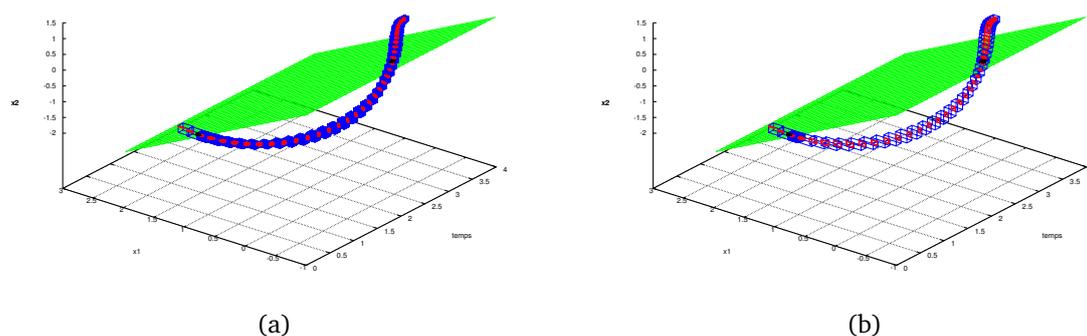


FIGURE 4.13 – L'ensemble atteignable du système masse-ressort en 3D ($x_2 \times x_1 \times t$), à gauche sans la fusion de trajectoires et à droite avec la fusion des trajectoires pour le critère P -radius.

4.5.3 EXEMPLE 3 : BALLE REBONDISSANTE SUR UNE SURFACE SINUSOÏDALE

Nous considérons le modèle incertain de la balle rebondissante sur une surface sinusoïdale, notre version est modifiée celle proposée par (Ishii et al., 2009). Elle peut être décrite par quatre variables (p_x, p_y, v_x, v_y) où (p_x, p_y) est la position de la balle en 2D et (v_x, v_y) la vitesse de la balle. La figure 4.14(a) représente l'automate hybride du système et la figure 4.14(b) illustre

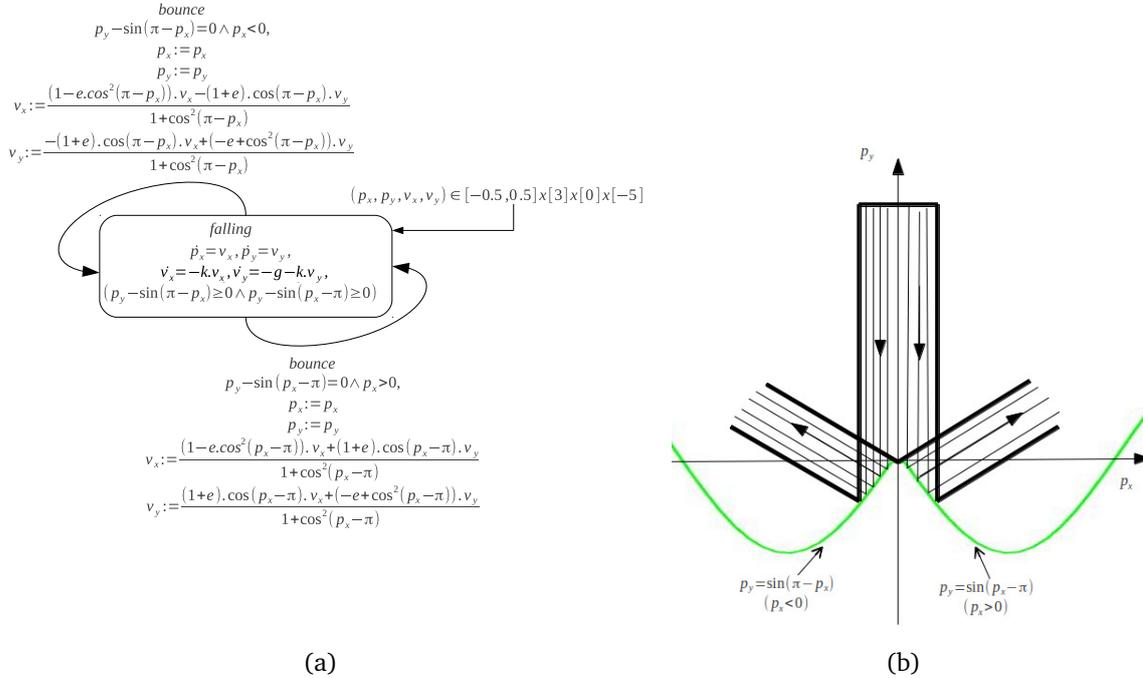


FIGURE 4.14 – A droite, automate hybride modélisant la balle rebondissante, et à gauche la balle rebondissante sur une surface non linéaire avec deux directions possible selon les positions initiales pour une hauteur donnée

la forme de la trajectoire de la balle avec des paramètres réglés comme $g \in [9.8, 9.85]$, $e = 3.5$ et $k \in [0.3, 0.4]$ (toutes les unités sont en S.I). Nous pouvons voir dans la figure 4.14(a), la dynamique de la balle rebondissante. Les algorithmes Hybrid-Transition (**algorithme 11**) et φ^{QR} (**algorithme 3**) ont été réglés comme suit : nous prenons un pas de temps constant $h = 0.1$ pour φ^{QR} et l'intervalle de temps est bissecté jusqu'à seuil $\varepsilon_T = 0.005$ dans Hybrid-Transition. Le temps final est 0.65s. Nous avons simulé notre méthode de passage de garde avec et sans fusion de trajectoires. Tous les résultats sont affichés dans les figures 4.16. Ces figures illustrent comment notre algorithme de passage de garde est capable de gérer deux tubes de trajectoires avec des conditions de garde non linéaires. De plus, nous arrivons à simuler le modèle avec un pas de temps assez grand, ce qui montre les bénéfices de la méthode de Taylor intervalle utilisée. Le tableau 4.2 regroupe les valeurs des critères pour le réglage du paramètre *ratio*. Dans cet exemple aussi, la valeur optimale dépend du choix du critère. Contrairement au premier cas d'étude, la MSBP la plus étroite est obtenue avec le critère longueur des segments. Les figures 4.18 illustrent l'ensemble atteignable pour la balle rebondissante en 3D, sans et avec la

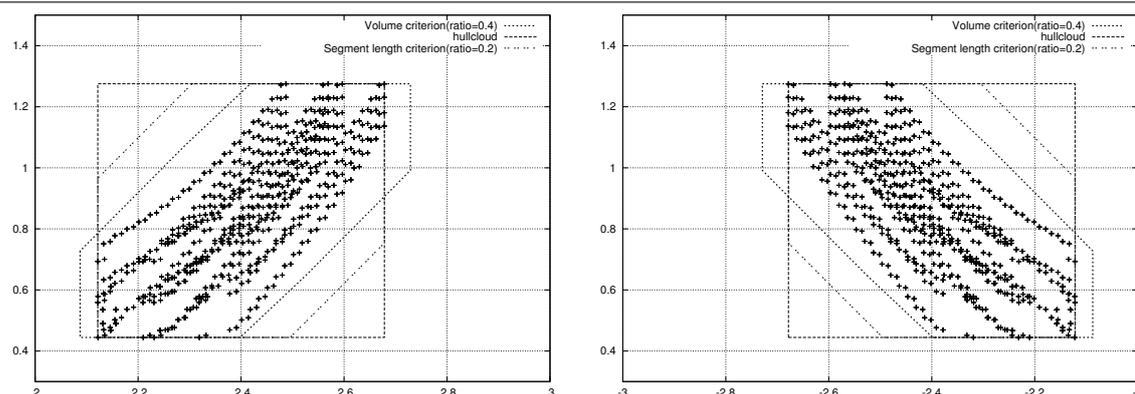


FIGURE 4.15 – Capture des sous-trajectoires droites et gauche après le premier rebond. Les points en noir représentent les sommets des MSBP (nuage de points) obtenues après le saut, l'unique MSBP est obtenu avec chacun des trois critères, ainsi que la valeur optimale pour *ratio*

fusion des trajectoires pour le critère longueur des segments.

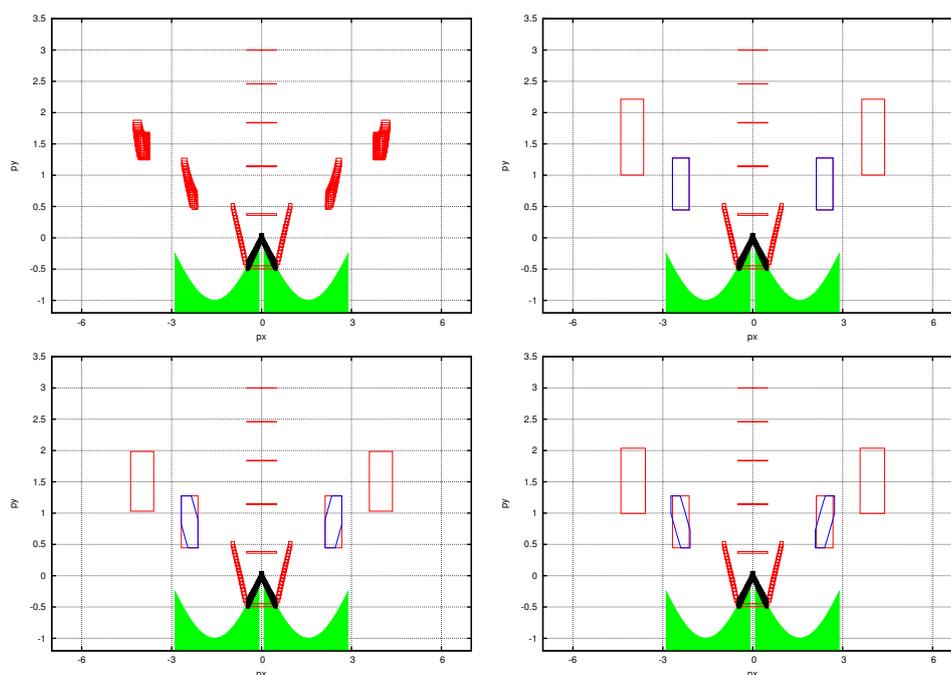


FIGURE 4.16 – La frontière, c'est-à-dire l'ensemble solution à chaque point de la grille temporelle dans l'espace $p_y \times p_x$ pour la balle rebondissante. De gauche à droite, l'ensemble solutions obtenu sans la fusion des trajectoires, avec la fusion des trajectoires en utilisant le critère P-radius, puis la longueur des segments, et le volume.

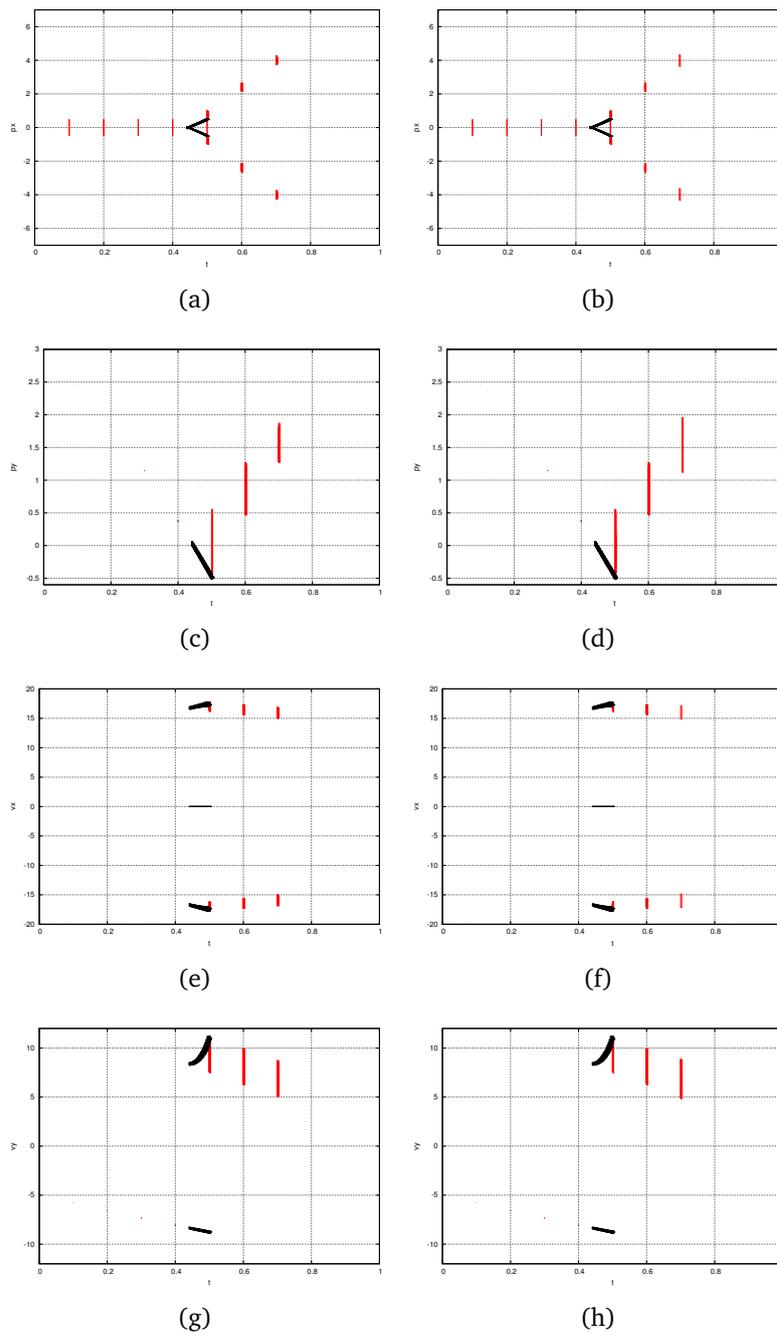


FIGURE 4.17 – L'ensemble atteignable (à gauche sans la fusion, et à droite avec la fusion avec le critère de longueur des segments) obtenu pour la balle rebondissante sur la surface sinusoïdale. Historique des variables p_x, p_y, v_x, v_y en fonction du temps

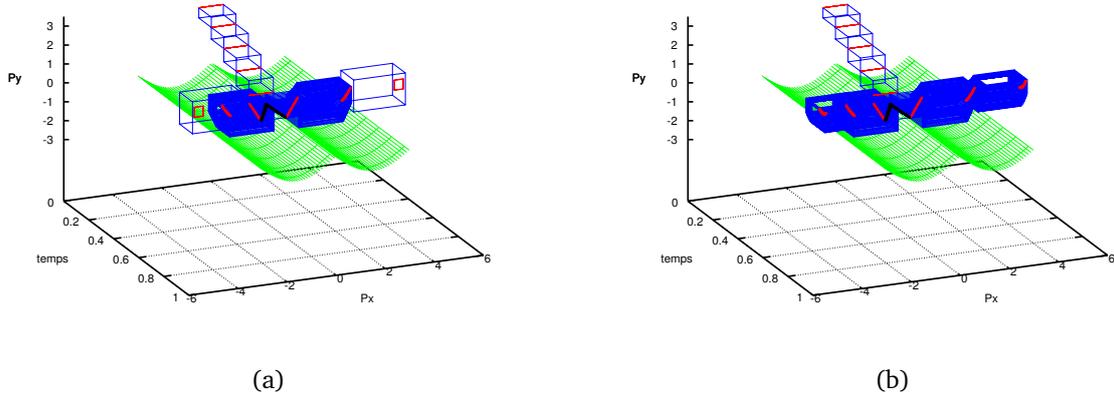


FIGURE 4.18 – L'ensemble atteignable de la balle rebondissante sur une surface sinusoïdale en 3D, (a) : à gauche sans la fusion de trajectoires et (b) : à droite avec la fusion des trajectoires pour le critère longueur des segments.

RÉSUMÉ DE LA MÉTHODE D'ATTEIGNABILITÉ HYBRIDE

Nous pouvons maintenant résumer les méthodes précédentes (passage ensembliste des gardes, saut ensembliste, et fusion des trajectoires) dans les algorithmes Transition-Hybride (11, 12, 13,14). Nous choisissons de n'expliquer qu'un seul de ces algorithmes (Transition-Hybride 11), car ils fonctionnent avec le même principe. La seule différence, entre ces algorithmes, réside dans la manière d'intersecter le tube de trajectoires et la condition de garde.

L'algorithme d'atteignabilité hybride traite à la fois des transitions continues et discrètes. Sa structure principale est similaire à celle suggérée dans (Henzinger et al., 2000a). Dans cette version, le pas d'intégration est fixé. L'algorithme gère une liste de triplets $(q, t, \chi(t))$ composée du mode, le temps et la forme composée (2.26) pour l'ensemble solution de l'état continu à l'instant t .

Les listes gérées sont les suivantes :

1. La liste de l'ensemble atteignable \mathcal{L}_j^R est l'ensemble atteint à l'instant t_j . Elle contient les enveloppes du tube de trajectoires continu de tous les modes, obtenu via l'**algorithme 3**, page 32, entre les instants t_0 et t_j (cette liste contient les solutions a priori).
2. La liste frontière \mathcal{L}_j^F est la frontière atteinte à l'instant t_j . Elle contient l'enveloppe du tube de trajectoires continu obtenu via l'**algorithme 3** à l'instant t_j , qui intersecte l'invariant. Notons que la forme composée χ est utilisée pour caractériser l'ensemble solutions.
3. La liste de saut courante \mathcal{L}_e est la liste courante servant à résoudre les transitions discrètes. Elle contient le mode q , un temps initial t_0 et l'ensemble solution de l'état continu à l'instant t_0 , mais elle contient aussi un temps final t_1 . Cette donnée supplémentaire est

nécessaire puisque nous analysons l'occurrence d'une transition discrète sur l'intervalle de temps $[t_0, t_1]$.

4. La liste frontière courante \mathcal{L} , qui est nécessaire pour procéder la transition continue avec le mode q' sur $[t_e, t_{j+1}]$, quand une transition discrète $e = q \rightarrow q'$ a lieu à $t_e \in [t_j, t_{j+1}]$.

L'**algorithme** Transition-Hybride procède à la transition continue sur le pas d'intégration $[t_j, t_{j+1}]$ (étapes 6 à 15) pour toutes les boîtes de la liste frontière courante. Il calcule l'enveloppe de toute la trajectoire à l'étape 6 et met à jour l'ensemble atteignable à l'étape 7. Il calcule l'ensemble solution à l'étape 8 et calcule à l'étape 9 l'intersection entre l'ensemble solution et l'invariant. Si non vide, l'ensemble solution filtré est stocké dans la liste frontière (étape 11). Si le nombre de boîtes solutions est supérieur à 2, les trajectoires sont fusionnées grâce à l'algorithme de fusion (**algorithme** 10).

L'**algorithme** Transition-Hybride procède ainsi à la transition discrète qui peut avoir lieu sur le pas d'intégration $[t_j, t_{j+1}]$. Pour toutes les boîtes dans la liste de saut courante, il vérifie si la transition discrète existe (étape 22). Si l'ensemble continu intersecte la garde (étape 23), alors la transition est faite si la taille de $[\underline{t}^*, \bar{t}^*]$ est plus petite qu'un seuil donné ϵ_T ¹⁵ (étape 25). Dans ce cas, l'enveloppe du tube de trajectoires sur $[\underline{t}^*, \bar{t}^*]$, à savoir χ est calculé (étape 26), ensuite nous utilisons un contracteur pour obtenir l'ensemble χ^* . Si non vide, alors l'algorithme procède à la transition discrète, la liste frontière courante est mise à jour (étape 29), et éventuellement une nouvelle transition continue se produira de \underline{t}^* à t_{j+1} (étape 4). Si la taille de $[\underline{t}^*, \bar{t}^*]$ est assez large, alors l'intervalle de temps est bissecté et la liste de sauts courante est mise à jour. Les temps initiaux et finaux sont mise à jour (étape 32 à 35)

4.6 ÉVALUATION DE LA MÉTHODE D'ATTEIGNABILITÉ HYBRIDE SUR DES BENCHMARKS

Dans cette partie, nous avons regroupé les résultats d'évaluation de la méthode rapide du calcul de l'ensemble atteignable par un système hybride non linéaire en présence d'incertitudes.

Nous utilisons des benchmarks proposés par Xin Chen¹⁶ :

Ensuite nous procédons à une comparaison de performances en comparant nos résultats avec ceux de Flow*¹⁷ sur les benchmarks disponibles sur le site¹⁸.

15. Le réglage de ce paramètre a un impact sur la qualité des résultats, en effet en choisissant ϵ_T très grand on bissecte moins donc les résultats sont moins précis. Par contre quand on choisit ϵ_T est petit, nous obtenons des résultats assez précis.

16. <http://systems.cs.colorado.edu/research/cyberphysical/taylormodels/>

17. <http://www-i2.informatik.rwth-aachen.de/i2/chen/>

18. <http://ths.rwth-aachen.de/research/hypro/benchmarks-of-continuous-and-hybrid-systems/>

4.6.1 BENCHMARK 1 : MANŒUVRE ANTI-COLLISION

Nous considérons le modèle du système anti-collision de deux avions pour une altitude fixée, proposée dans (Platzer and Clarke, 2007; Tomlin et al., 2003). Au début, les deux avions sont en vol rectiligne avec un cap relatif. Quand les avions se rapprochent et que la distance minimale entre les deux avions est inférieure à une valeur spécifique, le contrôleur apporte une modification instantanée de cap à 90 degré sur chacun d'eux. Ainsi les deux avions accompliront un vol en arc semi-circulaire de π (voir la figure 4.19). Ensuite, les deux avions refont un autre changement de cap instantané à 90 degré et reprennent leur cap initial. Un tel système peut être modélisé par un système hybride avec trois modes (voir la figure 4.19)

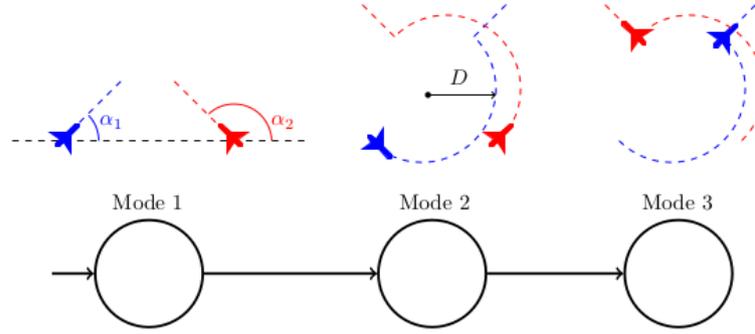


FIGURE 4.19 – Automate hybride du système de manœuvre anti-collision

Les variables d'états sont données par

- x_1, y_1 : les coordonnées du premier avion ;
- α_1 : le cap du premier avion ;
- x_2, y_2 : les coordonnées du deuxième avion ;
- α_2 : le cap du premier avion ;
- z : la minuterie pour le vol de l'arc semi-circulaire.

Les dynamiques de x_1, y_1, x_2, y_2 dans chaque mode sont

$$\dot{x}_1 = v_1 \cdot \cos \alpha_1, \quad \dot{y}_1 = v_1 \cdot \sin \alpha_1$$

$$\dot{x}_2 = v_2 \cdot \cos \alpha_2, \quad \dot{y}_2 = v_2 \cdot \sin \alpha_2$$

Les caps α_1, α_2 changent de valeurs uniquement dans le mode 2, ainsi que le minuteur z . Pour la transition du mode 1 au mode 2, la garde est $(x_1 - x_2)^2 + (y_1 - y_2)^2 \leq D^2$ et les fonctions de reset : $\alpha'_1 = \alpha_1 - \pi/2$, et $\alpha'_2 = \alpha_2 - \pi/2$. Pour la transition du mode 2 à mode 3, la garde est $z = \pi$, et les fonctions de reset sont identiques à celle de la transition $1 \rightarrow 2$.

Pour la simulation $v_1 = v_2 = 1$ et $D = 2$. La position initiale du premier avion est la région $(x_1(t_0) \times y_1(t_0)) \in ([-0.1, 0.1] \times [-0.1, 0.1])$. Et pour le second avion, la position initiale est la région $(x_2(t_0) \times y_2(t_0)) \in ([9.9, 10.1] \times [-0.1, 0.1])$. Le cap initial du premier avion est $\pi/4$ et celui du deuxième avion est $3\pi/4$. Les paramètres de réglages sont : 0.02 pour le pas de temps, et l'horizon temporel $T = 15$. Les résultats obtenus sont donnés dans les figures 4.20.

Pour ce benchmark, l'outil Flow* a besoin de transformer le système afin de le rendre polynomial en ajoutant 4 variables, donc on passe d'un système de dimension 8 à 12. Notre outil d'atteignabilité hybride n'a pas besoin de cette transformation, car il gère aussi bien les polynômes, que les fonctions trigonométriques. Ainsi dans Flow*, on introduit les quatre variables c_1, s_1, c_2, s_2 pour substituer respectivement les termes trigonométriques $\cos \alpha_1, \sin \alpha_1, \cos \alpha_2, \sin \alpha_2$.

Donc pour Flow*, la dynamique dans le mode 1 et 3 devient

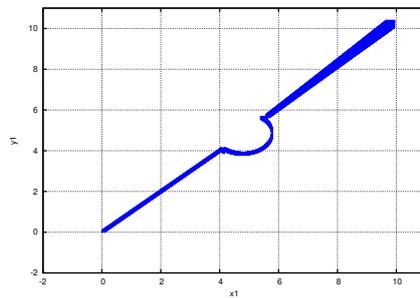
$$\dot{x}_1 = v_1 \cdot c_1, \quad \dot{y}_1 = v_1 \cdot s_1, \quad \dot{c}_1 = 0, \quad \dot{s}_1 = 0, \quad \dot{\alpha}_1 = 0, \quad \dot{z} = 0$$

$$\dot{x}_2 = v_2 \cdot c_2, \quad \dot{y}_2 = v_2 \cdot s_2, \quad \dot{c}_2 = 0, \quad \dot{s}_2 = 0, \quad \dot{\alpha}_2 = 0$$

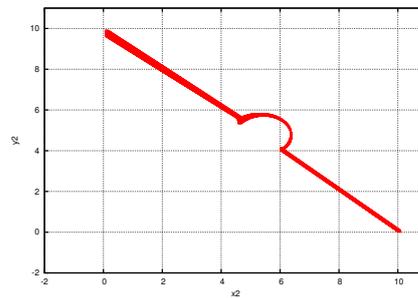
et la dynamique dans le mode 2 est

$$\dot{x}_1 = v_1 \cdot c_1, \quad \dot{y}_1 = v_1 \cdot s_1, \quad \dot{c}_1 = -s_1, \quad \dot{s}_1 = c_1, \quad \dot{\alpha}_1 = 1, \quad \dot{z} = 1$$

$$\dot{x}_2 = v_2 \cdot c_2, \quad \dot{y}_2 = v_2 \cdot s_2, \quad \dot{c}_2 = -s_2, \quad \dot{s}_2 = c_2, \quad \dot{\alpha}_2 = 1$$



(a) Trajectoire avion 1



(b) Trajectoire avion 2

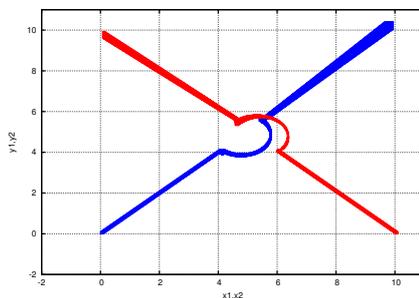
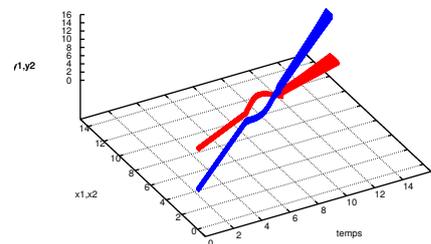

 (c) Plan de phase $y_2 - y_1 \times x_2 - x_1$

 (d) Plan de phase $y_2 - y_1 \times x_2 - x_1$

FIGURE 4.20 – Résultats système de manœuvre anti-collision

4.6.2 BENCHMARK 2 : INTÉGRATEUR NON-HOLONOME

Une version simplifiée de la commande hybride pour l'intégrateur non-holonome de Brockett (Brockett, 1982; Brockett and Dai, 1993) est décrite par l'EDO suivante.

$$\begin{cases} \dot{x} = u \\ \dot{y} = v \\ \dot{z} = x \cdot v - y \cdot u \end{cases}$$

où u et v sont des entrées de contrôle définies par :

$$u = \begin{cases} 1, & x^2 + y^2 \leq |z| \\ -x + \frac{2 \cdot y \cdot z}{x^2 + y^2}, & x^2 + y^2 > |z| \end{cases}$$

$$v = \begin{cases} 1, & x^2 + y^2 \leq |z| \\ -y - \frac{2 \cdot x \cdot z}{x^2 + y^2}, & x^2 + y^2 > |z| \end{cases}$$

Nous pouvons modéliser ce système par deux modes de fonctionnement. En effet si $x^2 + y^2 \leq |z|$, la dynamique dans le mode 1 qui est

$$\dot{x} = 1, \quad \dot{y} = 1, \quad \dot{z} = x - y \quad (4.58)$$

Si $x^2 + y^2 > |z|$ la dynamique dans le mode 2 est donnée par

$$\dot{x} = -x + \frac{2 \cdot y \cdot z}{x^2 + y^2}, \quad \dot{y} = -y - \frac{2 \cdot x \cdot z}{x^2 + y^2}, \quad \dot{z} = -2z \quad (4.59)$$

En considérant les conditions initiales $x = 0$, $y = 0$ et $z \in [14.9, 15.1]$ et l'horizon temporel $T = 7.5$, nous obtenons les résultats (regroupés dans les figures 4.21 et 4.22) .

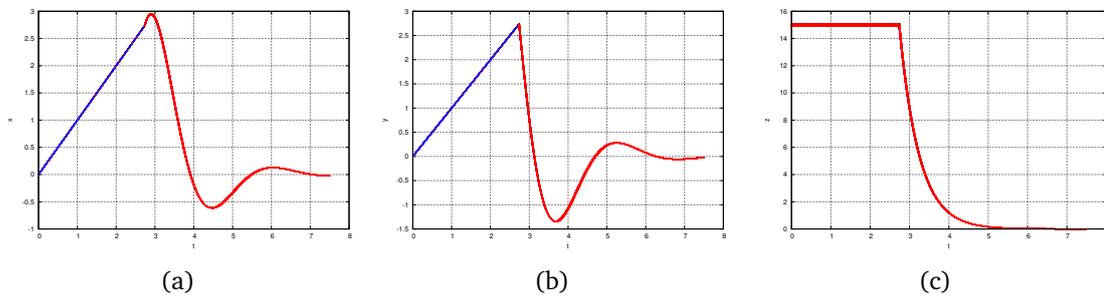


FIGURE 4.21 – L'ensemble atteignable obtenu pour l'intégrateur non-holonome. Historique des variables x, y, z en fonction du temps

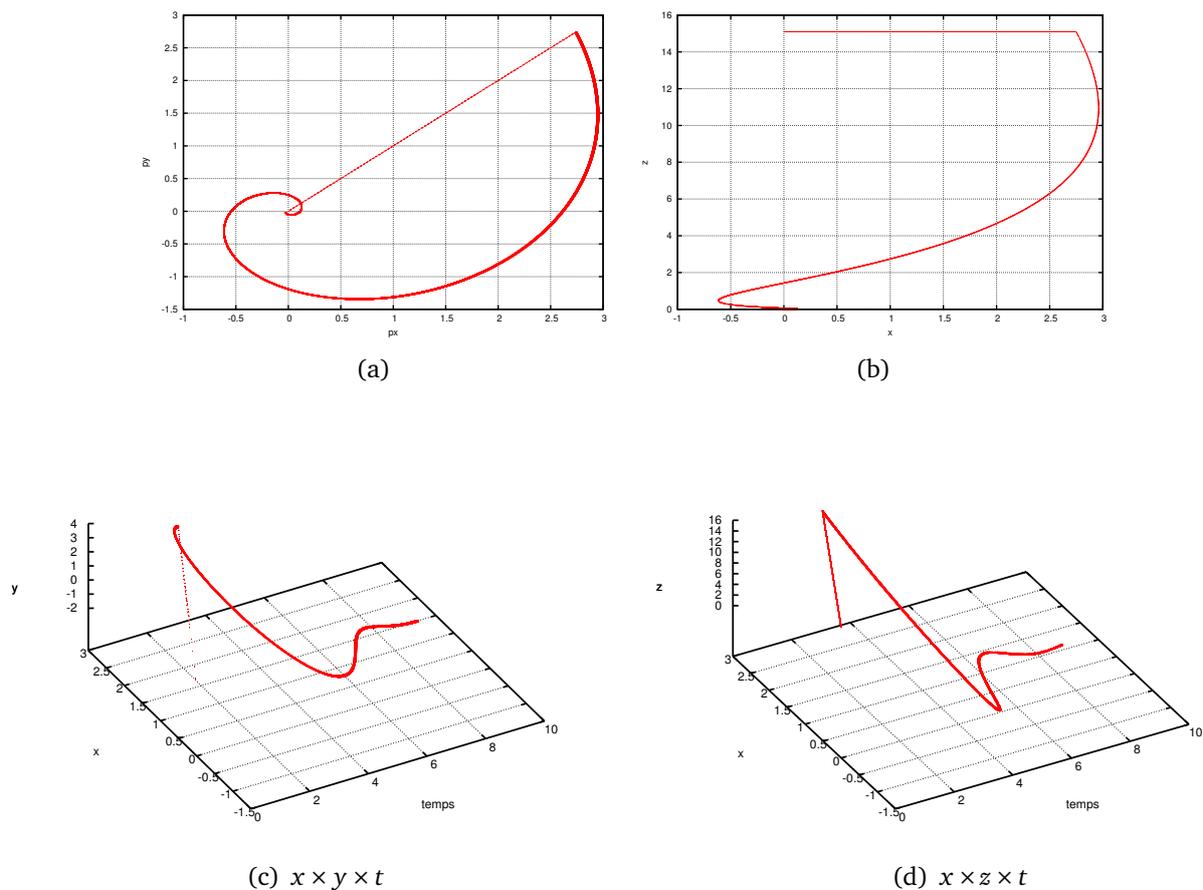


FIGURE 4.22 – Résultats intégrateur non-holonome

4.6.3 BENCHMARK 3 : CONTRÔLE DE LA GLYCÉMIE DE PATIENTS DIABÉTIQUES

Le modèle de contrôle de la glycémie dans le sang des patients diabétiques comporte trois variables d'états (G, X, I) , où G est la concentration de glucose, I la concentration d'insuline dans le plasma, et X la concentration d'insuline dans la chambre interstitielle. L'objectif de ce benchmark est de calculer l'ensemble atteignable (G, X, I) sur l'horizon temporel $t \in [0, 360]$.

Le modèle dynamique continu est comme suit

$$\begin{aligned}\frac{dG}{dt} &= -p_1 G - X(G + G_B) + g(t) \\ \frac{dX}{dt} &= -p_2 X + p_3 I \\ \frac{dI}{dt} &= -n(I + I_b) + \frac{1}{V_I} i(t)\end{aligned}$$

Les valeurs typiques des paramètres sont : $p_1 = 0.01, p_2 = 0.025, p_3 = 1.3 \cdot 10^{-5}, V_I = 12, n = 0.093, G_B = 4.5, I_b = 15$.

Les fonctions $i(t)$ et $g(t)$ déterminent les transitions entre les différents modes. En considérant $i(t) = i_1(t)$ proposée par (Furler et al., 1985), on a :

$$i_1(t) = \begin{cases} \frac{25}{3} & G(t) \leq 4 \\ \frac{25}{3}(G(t)-3) & G(t) \in [4, 8] \\ \frac{25}{3} & G(t) \geq 8 \end{cases} \quad (4.60)$$

Et en considérant $i(t) = i_2(t)$ proposée par (Fisher, 1991), on a :

$$i_2(t) = \begin{cases} 1 + \frac{2G(t)}{9} & G(t) < 6 \\ \frac{50}{3} & G(t) \geq 6 \end{cases} \quad (4.61)$$

Le flux de glucose après avoir pris un repas est modélisé par :

$$g(t) = \begin{cases} \frac{t}{60} & t \leq 30 \\ \frac{120-t}{180} & t \in [30, 120] \\ 0 & t \geq 120 \end{cases}$$

On considère les conditions initiales suivantes :

$$G(0) \in [-2, 2] \quad X(0) = 0 \quad I(0) \in [-0.1, 0.1]$$

BENCHMARK 3, CAS 1

Nous considérons ici le cas où $i(t) = i_1(t)$ (4.60). La dynamique discrète est décrite par l'automate hybride représenté sur la figure 4.23 avec neuf modes discrets obtenus en combinant chaque inégalité sur les variables t et $G(t)$.

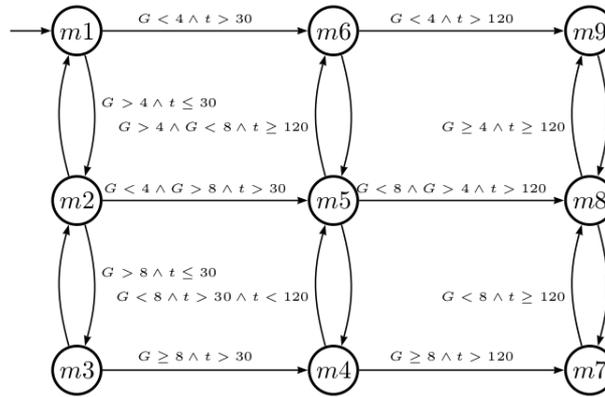


FIGURE 4.23 – Automate hybride-bench3-cas 1

Nous obtenons pour ce deuxième cas, l'ensemble atteignable dont la composante G est représentée sur la figure 4.25. Nous avons utilisé un pas de temps d'intégration constant $h = 0.2$ et les seuils de bisection $\varepsilon_T = 0.05$ dans l'algorithme 11.

BENCHMARK 3, CAS 2

Nous considérons ici le cas où $i(t) = i_2(t)$ (4.61). La dynamique discrète est décrite par l'automate hybride représenté sur la figure 4.24 avec six modes discrets obtenus en combinant chaque inégalité sur les variables t et $G(t)$.

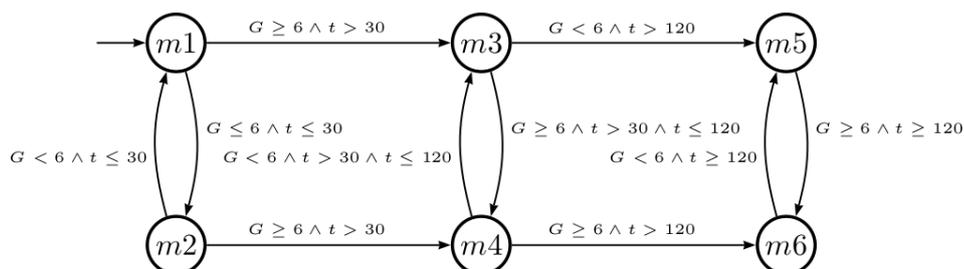
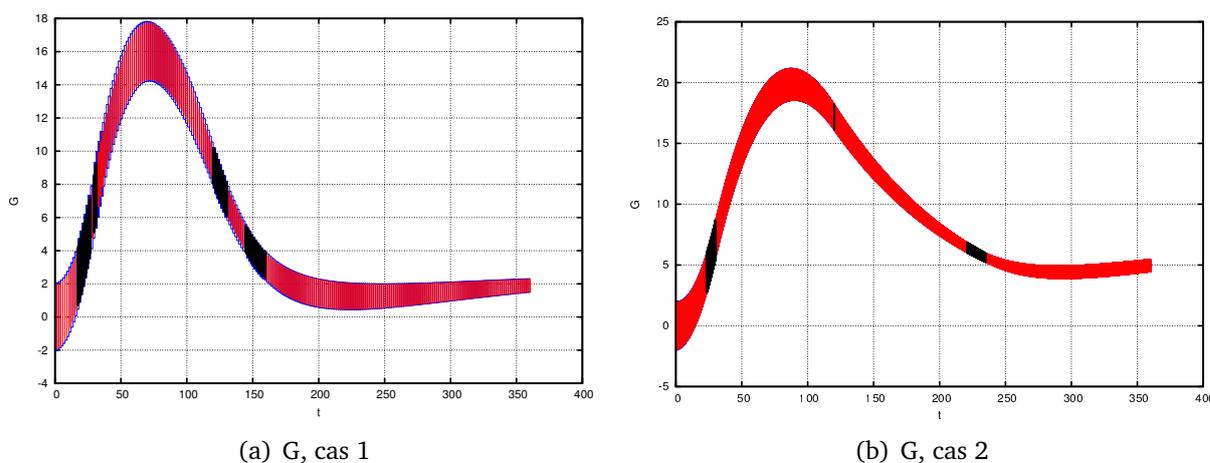


FIGURE 4.24 – Automate hybride-bench3-cas 2

Nous obtenons pour ce premier cas, l'ensemble atteignable dont la composante G est représentée sur la figure 4.25. Nous avons utilisé un pas de temps d'intégration constant $h = 0.2$ et les seuils de bisection $\varepsilon_T = 0.05$, dans l'**algorithme** 11, page 101.

FIGURE 4.25 – Composante G de l'ensemble atteignable pour les cas 1 et 2 du benchmark Glycémie.

COMMENTAIRES Notre solveur d'intégration numérique utilise un polynôme de Taylor par rapport au temps, ce polynôme est donc une fonction du temps et nous utilisons aussi la représentation MSBP pour contrôler l'effet d'enveloppement. Par contre, Flow* (Berz and Makino, 1998b) utilise ce que l'on appelle des modèles de Taylor intervalles qui est de développement en série de Taylor par rapport aux conditions initiales et du temps, donc c'est un polynôme du temps et des conditions initiales (voir tableau 4.6), Flow* donc arrive à mieux contrôler l'effet d'enveloppement mieux que notre outil. Aussi il arrive à travailler sur des pas d'intégration adaptatif. En

TABLE 4.5 – Tableau comparatif de notre outil d'atteignabilité hybride non linéaire avec Flow*

	NVAR	NT	DY_CO	G&I	FR	N_Mode	TT*(s)	TT (s)
Bench 1	7	2	NL	L	L	3	71.650	113.892
Bench 2	3	5	NL	NL	I	4	2545.36	2174.536
Bench 3_1	3	10	NL	L	I	6	157.71	102.95
Bench 3_2	3	18	NL	L	I	9	232.69	900.224

	Description
NVAR	Nombre de variables
NT	Nombre de transitions
DY_CO	Dynamique continue
G&I	Garde et invariant
FR	Fonction de reset
N_Mode	Nombre de modes
TT*	Temps total de calcul Flow*
TT	Temps total de calcul notre outil
NL	Polynôme non linéaire
L	Polynôme linéaire
I	Identité

conclusion, malgré notre intégrateur continue qui n'est pas optimisé, avec peut être une marge d'amélioration sur la fusion de trajectoires. On a des temps de calcul qui sont comparables, voir mieux que Flow* 2 instances sur 4 (voir tableau 4.5 et figure 4.6.3).

TABLE 4.6 – Comparaison entre les méthodes d'intégration numérique

	Flow*	Notre outil
Polynôme de Taylor	$p(X, t)$	$p(t)$
Pas d'intégration	adaptatif	fixe
Ordre du polynôme	adaptatif	fixe

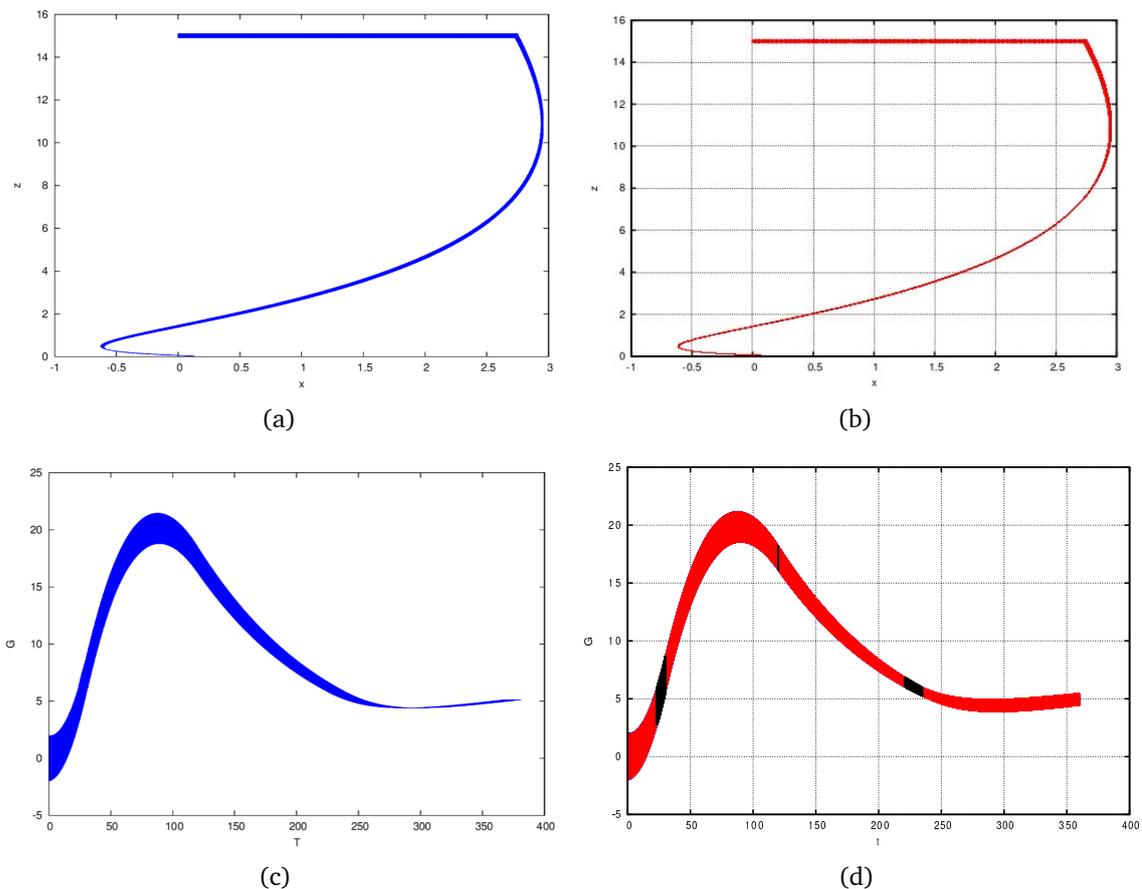


FIGURE 4.26 – Les figures (a-c) obtenus avec Flow* et les figures (b-d) obtenus avec notre outil

BIBLIOTHÈQUES

Pour les besoins de l'expérimentation de nos différentes méthodes développées dans ce mémoire. Nous utilisons Profil/Bias (Knüppel, 1994) pour l'arithmétique par intervalles, FABDAB++ (Bendtsen and Stauning, 1996) pour la différenciation automatique, AML++ Klima (2008-2010), Armadillo (Sanderson, 2013) et Eigen3 (Guennebaud et al., 2010) pour l'algèbre linéaire. Nous utilisons notre propre implémentation de la méthode de Lohner (**algorithme 3**) pour l'intégration ensembliste garantie des EDO pour les PVI. Finalement, nous utilisons les techniques de résolution des CSP implémentées dans IBEX (Chabert, 2013).

Toutes les expérimentations ont été testées sur une machine Ubuntu Linux *i5-2430M* 2.4GHz CPU avec 3.8GB de RAM. Nous avons aussi utilisé le logiciel Flow* (version *flowstar-1.2.2b*) sur notre machine afin de tester les benchmarks.

4.7 CONCLUSION

Nous avons proposé dans ce chapitre des méthodes permettant de faire le calcul d'ensemble atteignable hybride non linéaire.

Dans la première partie, nous avons proposé des méthodes pour le passage ensembliste de gardes non linéaires ou linéaires, c'est-à-dire la résolution du problème de la détection et localisation (temporelle) de l'intersection du tube de trajectoires continues avec des gardes ensemblistes. Nous avons considéré deux cas de figure, pour le premier cas, la condition de garde est linéaire. Ces méthodes sont basées sur des opérations géométriques d'intersection entre zonotope et hyperplan/demi-espace. L'avantage de ces méthodes réside dans le fait qu'elles sont réduites à des calculs d'algèbre linéaire simples (principalement, évaluation du noyau, SVD) qui assurent une complexité polynomiale et des temps de calcul réduits qui favorisent le passage à l'échelle.

Ensuite, pour le cas des gardes non linéaires, nous avons proposé une méthode rapide basée sur la résolution d'un problème de satisfaction de contraintes. Elle utilise la bisection dans la seule direction du temps et la collaboration de plusieurs contracteurs intervalles pour réduire le domaine des vecteurs d'état localisés sur la garde durant la tranche de temps étudiée.

Dans la deuxième partie, nous avons proposé une méthode pour le saut ensembliste. Ce problème revient à calculer l'image d'un ensemble par une fonction qui peut être non linéaire. A ce titre, nous avons pu élaborer une expression analytique de la fonction de reset en fonction d'une représentation zonotopique (domaine de solution de notre intégrateur). Cela nous a permis de réduire la sur-approximation introduite lors du saut.

Dans la troisième partie, nous avons proposé une méthode de fusion des trajectoires qui est une conséquence directe du problème de passage de gardes ensemblistes. Notre méthode est basée sur l'utilisation des zonotopes car le domaine de solution de notre intégrateur représente la solution sous forme de MSBP qui est un zonotope particulier. Après le saut, nous fusionnons toutes les trajectoires dans une MSBP qui est compatible avec notre approche d'atteignabilité continue. L'approche de fusion de trajectoires s'appuie sur un paramètre de réglage qui est choisi à la volée selon trois critères que nous avons définis. L'évaluation de la méthode sur les systèmes hybrides non linéaires a des performances prometteuses et en particulier, des temps de calcul relativement faibles.

Toutes ces méthodes utilisées conjointement, nous ont permis de calculer l'ensemble atteignable par un système dynamique hybride dont la dynamique continue est décrite par un système d'équations différentielles non linéaires et incertaines. Nous avons pu évaluer et tester les performances de notre outil avec l'outil Flow* de la littérature. Sur certaines instances, les temps de calcul obtenus avec notre outil sont nettement meilleurs.

Algorithm 11 : Algorithm Transition-Hybride (Garde non linéaire)

```

input  :  $\mathcal{L}_j^F, t_{j+1}, \{\varphi_q^{qr}(), \tilde{\varphi}_q()\}_{q \in \mathcal{Q}}, \{\gamma_e(), \rho_e()\}_{e \in \mathcal{E}}, \epsilon_T$ 
output :  $\mathcal{L}_{j+1}^F, \mathcal{L}_{j+1}^R$ 
1 Initialization : initialize running frontier list  $\mathcal{L} := \mathcal{L}_j^F$ ;
3 while  $\mathcal{L} \neq \emptyset$  do
4   pick up  $\mathcal{L}$  list element  $(q, t_0, [\chi_0])$ ;
5   /* Continuous transition */
6   compute continuous expansion over  $[t_0, t_{j+1}] \rightarrow [\tilde{z}_j]$ ;
7   update reached set list  $\mathcal{L}_{j+1}^R \leftarrow (q, t_0, t_{j+1}, [\tilde{z}_j])$ ;
8   compute new solution at time  $t_{j+1} \rightarrow [\chi_{j+1}]$ ;
9   solve CSP to compute  $[\chi'_{j+1}] := [\chi_{j+1}] \cap \text{inv}(q)$ ;
10  if  $[\chi_{j+1}]' \neq \emptyset$  then
11    | update frontier list  $\mathcal{L}_{j+1}^F \leftarrow (q, t_{j+1}, [\chi_{j+1}]')$ ;
12  end if
13  if  $\text{Length}(\mathcal{L}_{j+1}^F) > 2$  then
14    |  $\mathcal{L}_{j+1}^F := \text{MSBP}(\mathcal{L}_{j+1}^F)$ 
15  end if
16  /* Discrete transition */
17  forall the elements  $e \leftarrow \mathcal{E}$  do
18    initialize running jump list  $\mathcal{L}_e := \{(q, t_0, [\chi_0], t_1)\}; t_1 := t_{j+1}$ ;
19    while  $\mathcal{L}_e \neq \emptyset$  do
20      compute flow over  $[t_0, t_1] \rightarrow [\tilde{z}]$ ;
21      if  $e = (q, q')$  exists then
22        if  $(\gamma_e([\tilde{z}]) \ni 0)$  then
23           $\underline{t}^* := t_0; \bar{t}^* := t_1$ ;
24          if  $((\bar{t}^* - \underline{t}^*) \leq \epsilon_T)$  then
25            compute flow enclosure over  $[\underline{t}^*, \bar{t}^*] \rightarrow [\chi]$ ;
26             $[\chi]^* := \text{HC4\_Revise}([\chi])$ ;
27            if  $[\chi]^* \neq \emptyset$  then
28              | jump and update  $\mathcal{L} \leftarrow (q', \underline{t}^*, \rho_e([\chi]^*))$ ;
29            end if
30          else
31            compute solution set at  $\underline{t}^* \rightarrow [\chi]_1$ ;
32            compute solution set at  $\bar{t}^* \rightarrow [\chi]_2$ ;
33            update running jump list  $\mathcal{L}_e \leftarrow (q, \underline{t}^*, [\chi]_1, t_1)$ ;
34            update running jump list  $\mathcal{L}_e \leftarrow (q, \bar{t}^*, [\chi]_2, t_1)$ ;
35          end if
36        end if
37      end if
38    end while
39  end forall
40 end while
41 end while

```

Algorithme 12 : Transition-Hybride (Garde linéaire;zonotope-hyperplan)

```

input :  $\mathcal{L}_j^F, t_{j+1}, \{\varphi_q^{lqr}(), \tilde{\varphi}_q(), \nu_q()\}_{q \in \mathcal{Q}}, \{\gamma_e(), \rho_e()\}_{e \in \mathcal{E}}, \epsilon_T, \eta, d$ 
output :  $\mathcal{L}_{j+1}^F, \mathcal{L}_{j+1}^R$ 
1 Initialization : initialize running frontier list  $\mathcal{L} := \mathcal{L}_j^F$ ;
3 while  $\mathcal{L} \neq \emptyset$  do
4   pick up  $\mathcal{L}$  list element  $(q, t_0, [\chi_0])$ ;
5   /*Continuous transition                                     */
6     compute continuous expansion over  $[t_0, t_{j+1}] \rightarrow [\tilde{z}]_j$ ;
7     update reached set list  $\mathcal{L}_{j+1}^R \leftarrow (q, t_0, t_{j+1}, [\tilde{z}]_j)$ ;
8     compute new solution at time  $t_{j+1} \rightarrow [\chi_{j+1}]$ ;
9     solve CSP to compute  $[\chi'_{j+1}] := [\chi_{j+1}] \cap \text{inv}(q)$ ;
10    if  $[\chi_{j+1}]' \neq \emptyset$  then
11      | update frontier list  $\mathcal{L}_{j+1}^F \leftarrow (q, t_{j+1}, [\chi_{j+1}]')$ ;
12    end if
13    if  $\text{Length}(\mathcal{L}_{j+1}^F) > 2$  then
14      |  $\mathcal{L}_{j+1}^F := \text{MSBP}(\mathcal{L}_{j+1}^F)$ 
15    end if
16    /*Discrete transition                                     */
17    forall the elements  $e \leftarrow \mathcal{E}$  do
18      | initialize running jump list  $\mathcal{L}_e := \{(q, t_0, [\chi_0], t_{j+1})\}$ ;  $t_1 := t_{j+1}$ ;
19      | while  $\mathcal{L}_e \neq \emptyset$  do
20        | compute flow over  $[t_0, t_1] \rightarrow [\tilde{z}]$ ;
21        | if  $e = (q, q')$  exists then
22          | if  $(\gamma_e([\tilde{z}]) \ni 0)$  then
23            |  $\underline{t}^* = t_0$ ;  $\bar{t}^* = t_1$ ;
24            | if  $((\bar{t}^* - \underline{t}^*) \leq \epsilon_T)$  then
25              | compute flow enclosure over  $[\underline{t}^*, \bar{t}^*] \rightarrow [\tilde{\chi}]^*$ ;
26              |  $[c, R] \leftarrow \text{Box2zonotope}([\chi])$ ;
27              |  $(c_\cap, R_\cap) = \text{Intersection\_Z\_Hyperplan}(c, R, \eta, d)$ ;
28              | jump and update  $\mathcal{L} \leftarrow (q', \underline{t}^*, \rho_e(\mathcal{Z}_\cap))$ ;
29            | else
30              | compute solution set at  $\underline{t}^* \rightarrow [\chi]_1^*$ ;
31              | compute solution set at  $t_1 := (\underline{t}^* + \bar{t}^*)/2 \rightarrow [\chi]_2^*$ ; update running
32              | jump list  $\mathcal{L}_e \leftarrow (q, \underline{t}^*, [\chi]_1^*, t_1)$ ; update running jump list
33              |  $\mathcal{L}_e \leftarrow (q, t_1, [\chi]_2^*, \bar{t}^*)$ ;
34            | end if
35          | end if
36        | end while
37      | end forall
38    end while

```

Algorithme 13 : Transition-Hybride (Garde linéaire;zonotope-demi-espace)

```

input :  $\mathcal{L}_j^F, t_{j+1}, \{\varphi_q^{lqr}(), \tilde{\varphi}_q(), \nu_q()\}_{q \in \mathcal{Q}}, \{\gamma_e(), \rho_e()\}_{e \in \mathcal{E}}, \epsilon_T, \eta, d$ 
output :  $\mathcal{L}_{j+1}^F, \mathcal{L}_{j+1}^R$ 
1 Initialization : initialize running frontier list  $\mathcal{L} := \mathcal{L}_j^F$ ;
3 while  $\mathcal{L} \neq \emptyset$  do
4   pick up  $\mathcal{L}$  list element  $(q, t_0, [\chi_0])$ ;
5   /*Continuous transition                                     */
6     compute continuous expansion over  $[t_0, t_{j+1}] \rightarrow [\tilde{z}]_j$ ;
7     update reached set list  $\mathcal{L}_{j+1}^R \leftarrow (q, t_0, t_{j+1}, [\tilde{z}]_j)$ ;
8     compute new solution at time  $t_{j+1} \rightarrow [\chi_{j+1}]$ ;
9     solve CSP to compute  $[\chi'_{j+1}] := [\chi_{j+1}] \cap \text{inv}(q)$ ;
10    if  $[\chi_{j+1}]' \neq \emptyset$  then
11      | update frontier list  $\mathcal{L}_{j+1}^F \leftarrow (q, t_{j+1}, [\chi_{j+1}]')$ ;
12    end if
13    if  $\text{Length}(\mathcal{L}_{j+1}^F) > 2$  then
14      |  $\mathcal{L}_{j+1}^F := \text{MSBP}(\mathcal{L}_{j+1}^F)$ 
15    end if
16    /*Discrete transition                                     */
17    forall the elements  $e \leftarrow \mathcal{E}$  do
18      | initialize running jump list  $\mathcal{L}_e := \{(q, t_0, [\chi_0], t_{j+1})\}$ ;  $t_1 := t_{j+1}$ ;
19      | while  $\mathcal{L}_e \neq \emptyset$  do
20        | compute flow over  $[t_0, t_1] \rightarrow [\tilde{z}]$ ;
21        | if  $e = (q, q')$  exists then
22          | if  $(\gamma_e([\tilde{z}]) \ni 0)$  then
23            |  $\underline{t}^* = t_0$ ;  $\bar{t}^* = t_1$ ;
24            | if  $((\bar{t}^* - \underline{t}^*) \leq \epsilon_T)$  then
25              | compute flow enclosure over  $[\underline{t}^*, \bar{t}^*] \rightarrow [\tilde{\chi}]^*$ ;
26              |  $[c, R] \leftarrow \text{Box2zonotope}([\chi])$ ;
27              |  $(c_\cap, R_\cap) = \text{Intersection\_Z\_demi\_espace}(c, R, \eta, \gamma)$ ;
28              | jump and update  $\mathcal{L} \leftarrow (q', \underline{t}^*, \rho_e(\mathcal{Z}_\cap))$ ;
29            | else
30              | compute solution set at  $\underline{t}^* \rightarrow [\chi]_1^*$ ;
31              | compute solution set at  $t_1 := (\underline{t}^* + \bar{t}^*)/2 \rightarrow [\chi]_2^*$ ; update running
32              | jump list  $\mathcal{L}_e \leftarrow (q, \underline{t}^*, [\chi]_1^*, t_1)$ ; update running jump list
33              |  $\mathcal{L}_e \leftarrow (q, t_1, [\chi]_2^*, \bar{t}^*)$ ;
34            | end if
35          | end if
36        | end while
37      | end forall
38    end while

```

Algorithme 14 : Transition-Hybride (Garde linéaire;zonotope-polytope)

```

input :  $\mathcal{L}_j^F, t_{j+1}, \{\varphi_q^{lqr}(), \tilde{\varphi}_q(), \nu_q()\}_{q \in \mathcal{Q}}, \{\gamma_e(), \rho_e()\}_{e \in \mathcal{E}}, \epsilon_T, \eta, d$ 
output :  $\mathcal{L}_{j+1}^F, \mathcal{L}_{j+1}^R$ 
1 Initialization : initialize running frontier list  $\mathcal{L} := \mathcal{L}_j^F$ ;
3 while  $\mathcal{L} \neq \emptyset$  do
4   pick up  $\mathcal{L}$  list element  $(q, t_0, [\chi_0])$ ;
5   /*Continuous transition                                     */
6     compute continuous expansion over  $[t_0, t_{j+1}] \rightarrow [\tilde{z}]_j$ ;
7     update reached set list  $\mathcal{L}_{j+1}^R \leftarrow (q, t_0, t_{j+1}, [\tilde{z}]_j)$ ;
8     compute new solution at time  $t_{j+1} \rightarrow [\chi_{j+1}]$ ;
9     solve CSP to compute  $[\chi'_{j+1}] := [\chi_{j+1}] \cap \text{inv}(q)$ ;
10    if  $[\chi_{j+1}]' \neq \emptyset$  then
11      | update frontier list  $\mathcal{L}_{j+1}^F \leftarrow (q, t_{j+1}, [\chi_{j+1}]')$ ;
12    end if
13    if  $\text{Length}(\mathcal{L}_{j+1}^F) > 2$  then
14      |  $\mathcal{L}_{j+1}^F := \text{MSBP}(\mathcal{L}_{j+1}^F)$ 
15    end if
16    /*Discrete transition                                     */
17    forall the elements  $e \leftarrow \mathcal{E}$  do
18      | initialize running jump list  $\mathcal{L}_e := \{(q, t_0, [\chi_0], t_{j+1})\}$ ;  $t_1 := t_{j+1}$ ;
19      | while  $\mathcal{L}_e \neq \emptyset$  do
20        | compute flow over  $[t_0, t_1] \rightarrow [\tilde{z}]$ ;
21        | if  $e = (q, q')$  exists then
22          | if  $(\gamma_e([\tilde{z}]) \ni 0)$  then
23            |  $\underline{t}^* = t_0$ ;  $\bar{t}^* = t_1$ ;
24            | if  $((\bar{t}^* - \underline{t}^*) \leq \epsilon_T)$  then
25              | compute flow enclosure over  $[\underline{t}^*, \bar{t}^*] \rightarrow [\tilde{\chi}]^*$ ;
26              |  $[c, R] \leftarrow \text{Box2zonotope}([\chi])$ ;
27              |  $(c_\cap, R_\cap) = \text{Intersection\_Z\_Polytope}(c, R, \mathcal{P} = \cap_i^r \mathcal{H}_i^{\leq})$ ;
28              | jump and update  $\mathcal{L} \leftarrow (q', \underline{t}^*, \rho_e(\mathcal{Z}_\cap))$ ;
29            | else
30              | compute solution set at  $\underline{t}^* \rightarrow [\chi]_1^*$ ;
31              | compute solution set at  $t_1 := (\underline{t}^* + \bar{t}^*)/2 \rightarrow [\chi]_2^*$ ; update running
32              | jump list  $\mathcal{L}_e \leftarrow (q, \underline{t}^*, [\chi]_1^*, t_1)$ ; update running jump list
33              |  $\mathcal{L}_e \leftarrow (q, t_1, [\chi]_2^*, \bar{t}^*)$ ;
34            | end if
35          | end if
36        | end while
37      | end forall
38    end while

```

é

CHAPITRE 5

MÉTHODES ENSEMBLISTES POUR L'ESTIMATION DE PARAMÈTRES ET D'ÉTAT POUR LES SYSTÈMES HYBRIDES DANS LE CONTEXTE ERREUR BORNÉE

Sommaire

5.1 Motivations : détection de défauts et diagnostic	107
5.1.1 Approche prédiction/correction	109
5.1.2 Approche paramétrique	110
5.2 Observation ensembliste des SHIB	112
5.2.1 Approche prédiction-correction pour les SHIB	112
5.2.2 Exemple numérique 1 : système masse-ressort hybride	120
5.2.3 Exemple numérique 2 : système à deux réservoirs	124
5.2.4 Exemple de scénario de défaut	125
5.3 Méthode ensembliste pour l'estimation paramétrique des SHIB	126
5.3.1 Tests d'inclusion	127
5.3.2 SIVIA pour système hybride (SIVIA-H)	130
5.3.3 Exemple numérique : système masse-ressort hybride	132
5.3.4 Exemple de scénarios de défaut	134
5.4 Conclusion	136

5.1 MOTIVATIONS : DÉTECTION DE DÉFAUTS ET DIAGNOSTIC

Les méthodes de détection de défauts et de diagnostic des systèmes hybrides sont généralement des extensions de celles existantes pour les systèmes continus ou les systèmes à événements

discrets (Belard et al., 2011; Lafortune, 2015; Pencolé, 2003; Subias, 2006; Zaytoon and Lafortune, 2013). Parmi les extensions des approches issues des systèmes continus, nous pouvons citer l'estimation paramétrique (Isermann, 1984, 1993; Vento et al., 2015, 2012; Wahrburg and Adamy, 2012), l'estimation d'état (Isermann, 1997; Wang et al., 2007a; Zhao et al., 2005), les relations de redondance analytique ou l'espace de parité (Bayouhd et al., 2009; Mitchell and Tomlin, 2013; Shumsky and Zhirabok, 2012). D'autres approches sont issues des systèmes à événements discrets en utilisant par exemple le formalisme des réseaux de Petri (Mokhtari, 2007). Cependant, à notre connaissance, il n'existe pas à ce jour de travaux qui étudient la détection de défauts à base de modèles dans un contexte à erreurs bornées pour les systèmes dynamiques hybrides.

Dans une procédure de diagnostic par approche FDI (Fault Detection and Isolation), les tâches suivantes sont nécessaires :

- Détection de défaut : déterminer si un défaut est présent dans le système (décision binaire) ;
- Localisation (isolation) : déterminer le composant porteur du défaut ;
- Identification : déterminer la nature du défaut.

La méthode de diagnostic à base de modèles s'appuie sur la comparaison du comportement réel observé sur le système physique avec le comportement prédit à l'aide de modèles. La détection d'incohérences permet de conclure sur l'occurrence de défaut dans le système.

Un défaut est détecté si l'état hybride devient inconsistant avec le fonctionnement nominal, ce qui peut se traduire par un résidu (qui correspond à une différence entre le comportement prédit par le modèle de référence et le comportement observé du système (J. J. Gertler, 1997)) non nul, ou par l'estimation d'un mode discret correspondant à un mode de défaut. Dans ce dernier cas, on pourra procéder au diagnostic si l'automate hybride utilisé comporte des modes de défaut suffisamment détaillés (Mezyani, 2005).

En raison de leur nature très complexe, l'estimation d'état des systèmes hybrides est un problème important, un défi majeur et de challenge pour la communauté scientifique, avec un accent particulier sur les champs de la détection de défauts dans le milieu aéronautique (Bar-Shalom, 1987; Fisher and Maybeck, 2000; Maybeck and Hanlon, 1995; Mazor et al., 1998; Zhang and Jiang, 2001) et le diagnostic (Raïssi, 2012b; Seydou et al., 2012; Wang et al., 2007b, 2006).

L'estimation d'état des SDH est généralement divisée en deux tâches : l'estimation du mode (Bazart et al., 2013a) courant puis l'estimation de l'état continu connaissant le mode (Balluchi et al., 2002; Kajdan et al., 2007). De nombreuses contributions ne s'intéressent qu'au second point qui, malgré ce que l'on pourrait penser au premier abord, ne s'avère pas aussi simple qu'estimer l'état d'un système continu (Ichalal et al., 2010; Marx et al., 2010). En effet, du fait des particularités de la stabilité et de l'observabilité des SDH, il ne suffit pas à chaque sous-système d'être observable, ni d'assurer la stabilité asymptotique de l'erreur d'estimation pour chaque mode. Lorsque le mode est inconnu, son estimation est confiée à un observateur qui peut être un observateur destiné aux systèmes à événements discrets ou être basé sur un banc

d'observateurs continus. Dans ce dernier cas, chaque observateur du banc permet de générer des résidus qui sont nuls si le mode courant est le mode qui lui est associé.

D'autres méthodes utilisent l'estimation gaussienne multi-modèles (Wang et al., 2007b) et les approches basées sur le filtrage particulaire (Funiak and Williams, 2003), mais ces méthodes souffrent généralement d'une explosion combinatoire inévitable du nombre d'hypothèses à suivre. L'explosion est critique lorsque le modèle est dédié au diagnostic et comporte de plus un ensemble de modes de défaut. A ce problème s'ajoute le fait que les modes de défaut sont généralement d'un ordre de grandeur les moins probables, ce qui va à l'encontre du principe général de ces méthodes qui ne conservent que les hypothèses les plus probables. Ainsi, malgré de nombreuses solutions proposées pour contrer ces problèmes, force est de constater que l'approche stochastique n'est pas la panacée.

Dans (Eggers et al., 2012) le problème d'estimation ensembliste des systèmes hybrides est formulé comme un problème SAT mod ODE¹.

Nous allons proposer une méthode ensembliste basée sur un observateur de type prédiction-correction pour résoudre de façon robuste l'estimation ensembliste des systèmes hybrides.

Pour la détection de défauts, nous utiliserons deux approches. La première est basée sur un observateur de l'état hybride de type prédicteur-correcteur qui contient une étape de prédiction ensembliste basée sur un calcul d'atteignabilité hybride sur un horizon fini, puis une étape de correction qui vise à réduire les domaines atteignables en retirant les parties inconsistantes avec les données expérimentales et les bornes retenues pour les incertitudes. La deuxième approche est basée sur l'estimation de paramètres sur la base d'un modèle hybride.

5.1.1 APPROCHE PRÉDICTION/CORRECTION

L'estimation d'état se fait en deux étapes distinctes : la prédiction et la correction. La prédiction utilise d'une manière classique l'équation d'état (Jaulin et al., 2001; Kieffer et al., 1998; Raïssi et al., 2004a,b). La seconde étape, dite de correction, consiste à corriger l'état du système en utilisant l'équation de mesure, c'est-à-dire que l'on déduit l'état courant à partir de l'information fournie par la sortie mesurée. L'intersection des deux ensembles, prédit et déduit, donne l'état estimé par l'observateur d'état par prédiction/correction. De cette manière, l'étape de correction a l'avantage de limiter l'effet d'enveloppement inhérent aux méthodes de prédiction ensemblistes. La détection de défauts basée sur l'estimateur d'état par prédiction/correction consiste à vérifier la consistance entre l'ensemble des états prédits et l'ensemble des états consistants avec la mesure. Pour cela, l'intersection des deux ensembles est réalisée et un défaut est détecté si l'intersection est vide. Dans le contexte des SHIB, il s'agit de vérifier si le comportement prédit par le modèle (2.1–2.2–2.3) est compatible avec les mesures (2.7). Dans notre

1. Une formule SAT mod ODE est une combinaison booléenne sans quantificateur, de contraintes arithmétiques sur des réels, des entiers, des variables booléennes, des bornes sur ces variables, et des contraintes différentielles (mettant en jeu des équations différentielles)

cas, la démarche consiste à prédire, à l'instant $j + 1$, le domaine \mathcal{X}_{j+1}^F de l'état connaissant son domaine \mathcal{X}_j . La détection et la localisation sont alors effectuées en analysant l'intersection du domaine prédit \mathcal{X}_{j+1}^F avec le domaine \mathcal{X}_{j+1}^{inv} obtenu en utilisant les mesures. Ainsi, si

$$\mathcal{X}_{j+1}^F \cap \mathcal{X}_{j+1}^{inv} = \emptyset \quad (5.1)$$

les mesures ne sont pas compatibles avec le modèle nominal et un défaut est détecté.

La détection de défauts peut aussi se réaliser en utilisant un test "direct" (Guerra and Puig, 2008). Ce test repose sur l'estimation ensembliste de la sortie ((Blesa et al., 2011, 2012a,b)).

Le test direct est basé sur l'évaluation du résidu obtenu à partir de la différence entre les mesures y_j (2.7) et les sorties du modèle hybride \hat{y}_j (2.1–2.2–2.3) à chaque instant t_j :

$$r_j = y_j - \hat{y}_j \quad (5.2)$$

Idéalement, lorsque ni les erreurs de modélisation, ni le bruit sont présents, le résidu donné par l'équation (5.2), connu aussi comme une équation de parité, doit être différent de zéro dans un scénario de faute, et zéro autrement. Cependant, en raison des erreurs de modélisation et de mesure et ainsi que des incertitudes, on peut déduire un domaine admissible pour les résidus, et le test de détection repose sur la vérification de la condition suivante.

$$0 \notin \Gamma_j \quad (5.3)$$

où Γ_j est l'ensemble admissible pour le résidu r_j en considérant des incertitudes paramétriques et un bruit additif bornés

$$\Gamma_j = \{r_j | r_j = y_j + \epsilon_j - \hat{y}_j \text{ and } \epsilon_j \in \mathbb{E}_j\} \quad (5.4)$$

5.1.2 APPROCHE PARAMÉTRIQUE

Dans de nombreux domaines de l'ingénierie, les paramètres sont inconnus et doivent être estimés à partir de données expérimentales. Ces types de problèmes inverses (Aster et al., 2013) surviennent dans de nombreux domaines, notamment le diagnostic (Isermann, 1984, 1993), la pharmacocinétique, la biologie des systèmes, l'identification des systèmes, l'analyse de stabilité (Hiskens, 2001; Johnson and Tucker, 2008).

Dans la littérature, les problèmes d'estimation d'état et/ou des paramètres sont généralement résolus par des méthodes probabilistes, qui sont pertinentes quand une caractérisation explicite du bruit de mesure est disponible (Kay, 1993; Walter and Piet-Lahanier, 1990; Walter and

2. F pour Forward

Pronzato, 1994). Nous pouvons trouver dans (Hoffman et al., 2009) une approche probabiliste pour résoudre l'estimation des paramètres des systèmes hybrides. Dans certaines applications, le bruit ne peut pas être décrit par une loi de probabilité (on ne dispose pas d'assez de données par exemple), dans ce cas il est plus judicieux de considérer que ces perturbations sont bornées et de bornes connues.

L'estimation dans un contexte à erreurs bornées constitue alors une alternative intéressante aux méthodes statistiques. Le problème est donc de caractériser à chaque instant, d'une manière garantie, toutes les valeurs du vecteur d'état compatibles avec les mesures et avec les bornes d'erreurs supposées connues (Raïssi et al., 2004a). Le problème de l'estimation des paramètres peut formellement être considéré comme un cas particulier de l'estimation d'état (Bezzaoucha et al., 2013b; Braems et al., 2005; Raïssi et al., 2004a, 2006, 2009)). Nous avons trouvé dans la littérature de nombreuses contribution portant sur la reconstruction des paramètres (Bazart et al., 2013b; Bezzaoucha et al., 2013a; Granvilliers et al., 2004; Kieffer and Walter, 2011; Raïssi et al., 2004a, 2006; Tucker et al., 2007; Walter and Kieffer, 2007) pour les systèmes continus.

Pour détecter l'apparition de défaillances dans le système, il faut effectuer la comparaison entre les paramètres estimés et les paramètres nominaux théoriques (Isermann, 1984, 1993; Willsky, 1976). L'estimation paramétrique possède l'avantage d'apporter de l'information sur la taille des déviations par rapport au nominal. Elle réalise la localisation et l'identification simultanément. Toutefois, un des inconvénients majeurs de la méthode réside dans la nécessité d'avoir un système physique excité en permanence (Gucik-Derigny, 2011).

Dans le test inverse (Guerra and Puig, 2008), le processus de détection de défauts est réalisé par une estimation de la valeur des paramètres du modèle en utilisant des procédés d'identification de systèmes et en comparant ces valeurs estimées à des valeurs nominales (considérées connues). La différence entre ces valeurs forme un résidu (Isermann, 1993)

$$r_j = p_n - p \quad (5.5)$$

Ici nous utilisons l'ensemble \mathbb{S} donné par l'équation (2.10) (page 20) et notons par \mathbb{P}_N l'ensemble de valeurs de paramètres nominaux.

1. premièrement \mathbb{P}_N est estimé avec des données sans défaut,
2. ensuite en cours de fonctionnement, l'ensemble \mathbb{S} donné par (2.10) est estimé à partir des données (contenant éventuellement des défauts) sur un horizon donné,
3. l'intersection $\mathbb{S} \cap \mathbb{P}_N$ est calculé,
4. un défaut est détecté si

$$\mathbb{S} \cap \mathbb{P}_N = \emptyset,$$

ce qui signifie que les valeurs des paramètres nominaux ne peuvent pas expliquer les données réelles tout en tenant compte des bornes d'erreur ; les données réelles et le modèle sont inconsistants, par conséquent il y a un défaut.

Nous présentons ci-dessous les méthodes pour l'observation ensembliste et l'estimation des paramètres pour les systèmes hybrides à Incertitudes Bornées (**SHIB**). Celles-ci nous permettent de réaliser les opérations de détection de défaut.

5.2 OBSERVATION ENSEMBLISTE DES SHIB

5.2.1 APPROCHE PRÉDICTION-CORRECTION POUR LES SHIB

5.2.1.1 ÉTAPE DE PRÉDICTION

L'étape de prédiction consiste à déterminer le domaine χ_{j+1}^F incluant les vecteurs d'état et les modes atteignables, connaissant le domaine χ_j . Pour cela, nous utilisons notre algorithme d'atteignabilité hybride non-linéaire (voir **algorithme 11**).

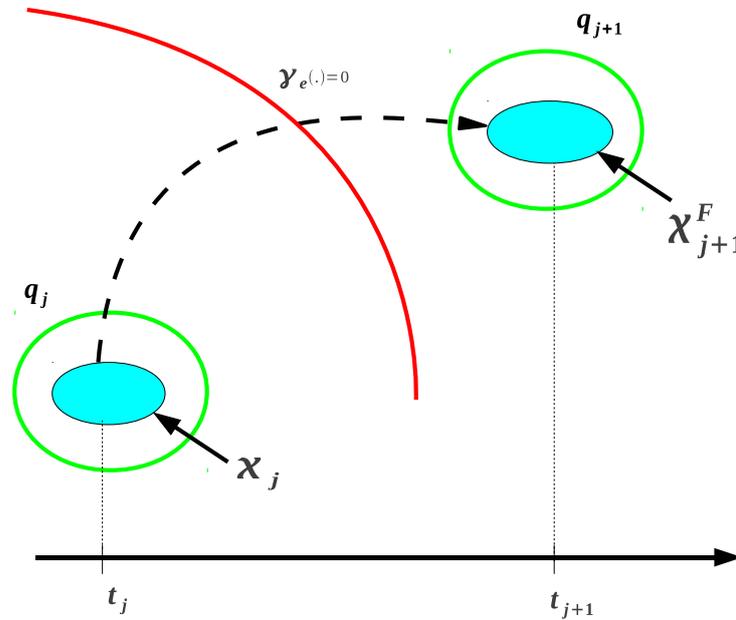


FIGURE 5.1 – Prédiction et atteignabilité hybride

5.2.1.2 ÉTAPE DE CORRECTION

A l'instant t_{j+1} , la mesure $y(t_{j+1})$ est disponible. L'étape de correction revient à déterminer le domaine χ_{j+1}^{inv} contenant l'ensemble des valeurs du vecteur d'état compatibles avec le domaine de la mesure à cet instant. Ce domaine est donné par :

$$\chi_{j+1}^{inv} \subset \{x \in \mathcal{D} \mid (g_{q_{j+1}}(x(t), p, t) \in \mathbb{Y}_{j+1}) = g_{q_{j+1}}^{-1}(\mathbb{Y}_{j+1}) \cap \mathcal{D} \quad (5.6)$$

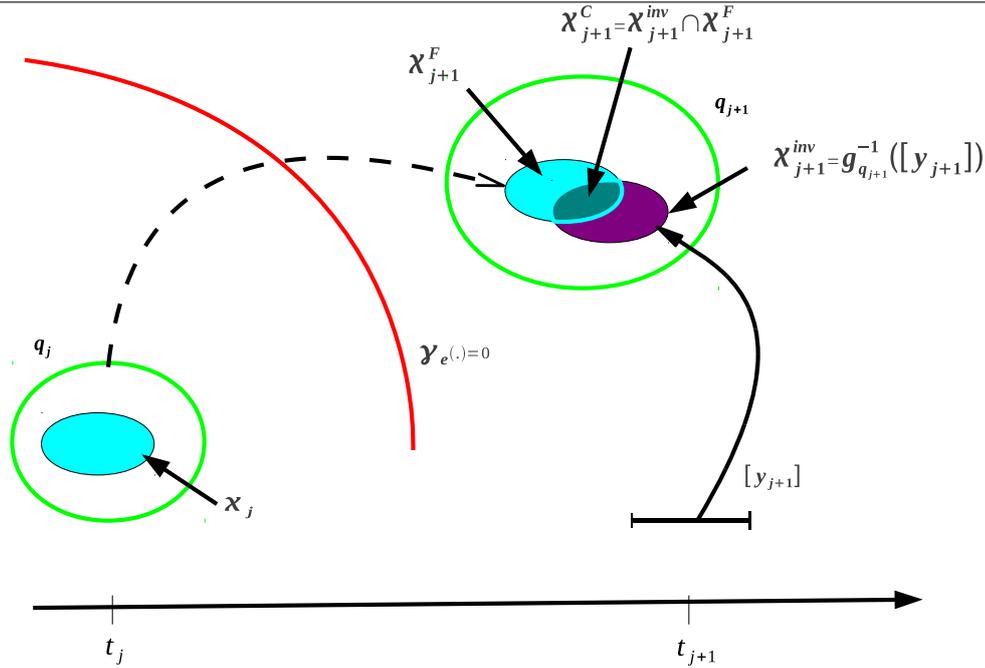


FIGURE 5.2 – Prédiction-correction hybride

L'estimation de l'état à l'instant t_{j+1} est donnée par l'intersection entre ces ensembles

$$\mathcal{X}_{j+1}^c = \mathcal{X}_{j+1}^F \cap \mathcal{X}_{j+1}^{inv} \quad (5.7)$$

avec $\mathbb{Y}_{j+1} = y_{j+1} + \mathbb{E}_{j+1}$, où \mathbb{E}_{j+1} est le domaine des incertitudes de mesure, supposé connu a priori. Si plusieurs modes sont atteignables, on applique à chacun les formules (5.6) et (5.7) avec l'évaluation de mesure correspondante.

La phase de correction est réalisée à l'aide de l'algorithme d'inversion ensembliste SIVIA (Set Inversion Via Interval Analysis) rappelé dans la Section 2.3. Cette approche permet de générer trois listes de pavés : admissibles, inadmissibles et indéterminés. L'intersection entre l'ensemble prédit et l'ensemble obtenu par inversion ensembliste sera réalisée par le biais des méthodes à base de zonotopes proposées dans (Combastel et al., 2008; Lalami and Combastel, 2007). Nous rappelons que notre ensemble atteignable hybride est représenté par une union de MSBP (zonotopes particuliers). Ainsi, en inversant les mesures, nous obtenons une union de boîtes qui sont aussi des zonotopes particuliers. L'intersection entre l'ensemble atteignable et l'ensemble obtenu par inversion des mesures revient à calculer une intersection entre zonotopes. Nous allons commencer par montrer comment on peut détecter la collision entre deux zonotopes, ensuite nous utiliserons une méthode de majoration de l'intersection de deux zonotopes (Combastel et al., 2008).

DÉTECTION DE COLLISION ENTRE DEUX ZONOTOPES Nous allons utiliser un algorithme sous-optimal de détection de collision qui présente l'avantage de s'appuyer sur un algorithme clas-

sique de l'algèbre linéaire.

Il s'agit de l'algorithme itératif Gilbert, Johnson et Keerthi (GJK) (notamment dans sa variante ISA-GJK) peut-être utilisé (Gilbert et al., 1988). Celui-ci calcule la distance³ entre les enveloppes convexes de deux ensembles de points X et Y . Il donne la valeur de cette distance dans le cas où les deux enveloppes convexes sont séparées, et une approximation de la distance d'interpénétration dans le cas contraire. Par contre cet algorithme présente l'inconvénient d'être lent en terme de convergence et de ne pas garantir le minimum global. En effet, la résolution est une méthode de recherche locale.

L'intersection entre deux zonotopes $Z_1(c_1 + \blacksquare(R_1))$ et $Z_2(c_2 + \blacksquare(R_2))$ est non vide s'il existe au moins un point dans l'espace abstrait⁴ de chaque zonotope (s_1 pour $\blacksquare(R_1)$ et s_2 pour $\blacksquare(R_2)$) qui vérifie la relation :

$$Z_1(c_1 + \blacksquare(R_1)) \cap Z_2(c_2 + \blacksquare(R_2)) \neq \emptyset \iff \begin{cases} \exists s_1 \in \mathbf{B}^{p_1} \\ \exists s_2 \in \mathbf{B}^{p_2} \end{cases}, \quad c_1 + R_1 s_1 = c_2 + R_2 s_2 \quad (5.8)$$

Le problème peut être reformulé sous la forme d'un test d'inclusion de 0 dans la somme de Minkowski de $(c_1 + \blacksquare(R_1))$ et $-(c_2 + \blacksquare(R_2))$:

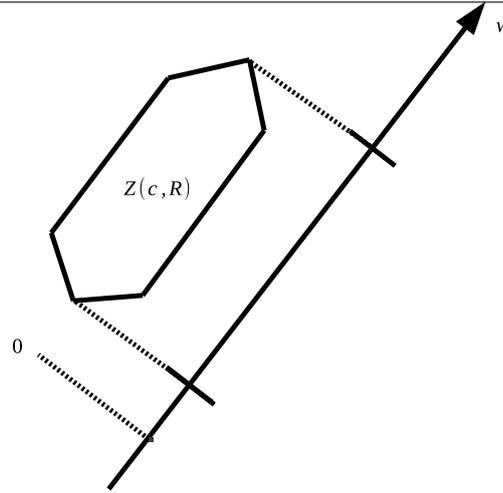
$$Z_1(c_1 + \blacksquare(R_1)) \cap Z_2(c_2 + \blacksquare(R_2)) = \emptyset \iff 0 \notin Z(c + \blacksquare(R)) \quad \text{avec } c = c_1 - c_2 \quad \text{et } R = [R_1 - R_2] \quad (5.9)$$

Même si l'on parle de détection de collision du fait de l'équivalence avec (5.8), le problème se ramène en fait à tester la non appartenance de l'origine à un zonotope $Z(c + \blacksquare(R))$. La solution consiste à chercher un vecteur de séparation w dont la direction vise à prouver la non appartenance de 0 au zonotope $Z(c + \blacksquare(R))$ (voir la figure 5.3).

En effet, un zonotope est un domaine convexe et la projection de ce domaine suivant une direction w donne un segment de droite. Ce segment de droite peut être caractérisé par un intervalle de valeur dans un repère mono-dimensionnel orienté le long de w . Il est alors possible de tester si la coordonnée de la projection de l'origine dans ce même repère appartient ou non à cet intervalle. En s'appuyant sur la convexité du zonotope $Z(c + \blacksquare(R))$, le test de non appartenance de l'origine à ce zonotope peut être reformulé analytiquement. Soit $R = [r_1 \dots r_p]$,

3. La distance euclidienne entre les deux enveloppes convexes (C_X, C_Y) est définie par : $d(C_X, C_Y) = \min\{|x - y|, x \in C_X, y \in C_Y\}$

4. L'espace \mathbb{R}^p des p grandeurs incertaines indépendantes correspondant aux composantes du vecteur s .


 FIGURE 5.3 – Projection du zonotope $Z(c + \blacksquare(R))$ et de l'origine sur l'axe orienté suivant w

dont les colonnes $r_1 \dots r_p$ sont les générateurs du zonotope.

$$\begin{aligned}
 0 \notin Z(c + \blacksquare(R)) &\Leftrightarrow \exists w, \quad 0 \notin w^\top c + w^\top R \mathbf{B}^p \\
 &\Leftrightarrow \exists w, \quad -w^\top c \notin w^\top R \mathbf{B}^p \\
 &\Leftrightarrow \exists w, \quad -w^\top c \notin w^\top (r_1 \mathbf{B} + \dots + w^\top r_p \mathbf{B}) \\
 &\Leftrightarrow \exists w, \quad -w^\top c \notin \left[-\sum_i |w^\top r_i|, \sum_i |w^\top r_i| \right] \\
 &\Leftrightarrow \exists w, \quad |w^\top c| \notin \left[0, \sum_i |w^\top r_i| \right] \\
 &\Leftrightarrow \exists w, \quad |w^\top c| > \|w^\top R\|_1
 \end{aligned} \tag{5.10}$$

où $\|*\|_1$ est la norme 1 du vecteur $*$.

Trouver un vecteur de séparation w peut donc se ramener à un problème de maximisation de critère :

$$J_1(w) = \frac{|w^\top c|}{\|w^\top R\|_1}, \quad w^* = \arg(\max_w (J_1(w))) \tag{5.11}$$

Si $|w^{*\top} c| > \|w^{*\top} R\|_1$ c'est à dire $J_1(w^*) > 1$ alors $0 \notin Z(c + \blacksquare(R))$. L'exemple de la figure 5.4 illustre les valeurs du critère $J_1(w)$ pour trois vecteurs w (w_1, w_2, w_3) dans des directions différentes. Seule la direction w_1 permet de prouver la séparation (c'est à dire la non appartenance de 0 au zonotope) parmi les trois cas illustrés sur la figure.

La résolution directe de (5.11) conduisant à des algorithmes itératifs peu appropriés à une

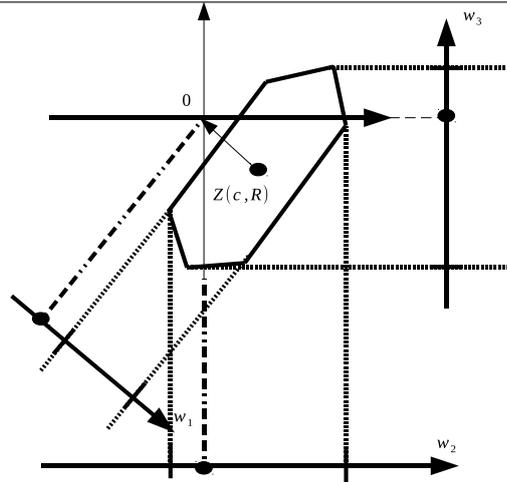


FIGURE 5.4 – Critère pour trouver un vecteur de séparation

implantation en ligne rapide des tests, une solution sous-optimale consiste à optimiser un critère basé sur la norme 2 pour trouver le vecteur de séparation candidat. L'idée reste de trouver un axe de projection le long duquel la projection du centre du zotoppe est reportée le plus loin possible par rapport à la projection du domaine caractérisé par R :

$$J_2(w) = \frac{\|w^\top c\|_2^2}{\|w^\top R\|_2^2} = \frac{w^\top c c^\top w}{w^\top R R^\top w} \quad (5.12)$$

Trouver un w qui maximise $J_2(w)$ se ramène à un calcul des valeurs propres généralisées⁵ de la paire $(c c^\top, R R^\top)$ et au calcul du vecteur associé à la plus grande valeur propre généralisée. Une fois le vecteur de séparation candidat w trouvé, le test de non appartenance de l'origine au zotoppe $Z(c, R)$ se fait en calculant $|w^\top c| > \|w^\top R\|$ (voir **algorithme 15**). Si le test $|w^\top c| > \|w^\top R\|$ est vrai (valeur 1), alors on est sûr que $0 \notin Z(c + \blacksquare(R))$, donc l'intersection est vide. Par contre si le test $|w^\top c| > \|w^\top R\|$ est faux (valeur 0), alors on ne peut pas affirmer avec certitude sur l'appartenance de 0 à $Z(c + \blacksquare(R))$, 0 peut être en effet dans $Z(c, R)$ ou près de $Z(c, R)$.

CALCUL D'INTERSECTION ENTRE DEUX ZONOTOPES Nous allons utiliser l'approche proposée dans (Combastel et al., 2008) pour la majoration par un zotoppe de l'intersection de deux zotopes. Soient deux zotopes $Z_1(c_1 + \blacksquare(R_1)), Z_2(c_2 + \blacksquare(R_2))$. D'après la définition d'un zotoppe, on a : $\forall x \in Z_1(c_1 + \blacksquare(R_1)) \cap Z_2(c_2 + \blacksquare(R_2)), \exists s_1 \in \mathbf{B}^{p_1}, \exists s_2 \in \mathbf{B}^{p_2} | x = c_1 + R_1 s_1 = c_2 + R_2 s_2$. La recherche d'un majorant de l'intersection entre $Z_1(c_1 + \blacksquare(R_1))$ et $Z_2(c_2 + \blacksquare(R_2))$ peut donc se faire en cherchant un majorant $[s]$ de l'ensemble des $s = [s_1, s_2] \in \mathbf{B}^p$ tels que $\Upsilon s = b$ avec $\Upsilon = [R_1 \quad -R_2]$, $b = (c_2 - c_1)$ et $p = p_1 + p_2$. Une manière d'obtenir $[s]$ repose sur le calcul d'une

5. Eigen3 : bibliothèque C++ en open source pour le calcul des valeurs propres, http://eigen.tuxfamily.org/index.php?title=Main_Page

Algorithme 15 : Détection d'intersection entre deux zonotopes

```

input  :  $Z_1(c_1 + \blacksquare(R_1)), Z_2(c_2 + \blacksquare(R_2))$ 
output :  $true, false$ 
1  $Z(c := c_2 - c_1, R := [R_1 \quad -R_2]);$ 
2  $[V, D] := eig(cc^\top, RR^\top);$ 
3  $D := diag(D);$ 
4  $i := index\_max(D);$ 
5  $w := V(:, i);$ 
6 if ( $|w^\top c| > \|w^\top R\|$ ) then
7   | return  $true;$ 
8 else
9   | return  $false;$ 
10 end if

```

décomposition en valeurs singulières de la matrice Υ ce qui conduit à l'**algorithme 16** (voir la référence (Combastel et al., 2008)).

La décomposition en valeurs singulières de Υ est donnée par $U \cdot S \cdot V^\top$ et telle que $U^\top U = I$, $V^\top V = I$. On a $U = U_1$, $S = [S_1, 0_{1 \times p_2}]$ et $V = \begin{bmatrix} V_1^\top \\ V_0^\top \end{bmatrix}$ où S_1 est une matrice diagonale dont les éléments diagonaux correspondent aux valeurs singulières non nulles de Υ . V_0 est une base du noyau de Υ .

Considérons le changement de variable $s = V \cdot \sigma = V_1 \cdot \sigma_1 + V_0 \cdot \sigma_0$. L'équation $\Upsilon s = b$ peut ainsi se réécrire sous la forme $U_1 S_1 V_1^\top \cdot (V_1 \cdot \sigma_1 + V_0 \cdot \sigma_0) = b$ d'où $\sigma_1 = S_1^{-1} \cdot U_1^\top b$. Comme $\sigma_0 = V_0^\top s$ et $s \in \mathbf{B}^p$, $\sigma_0 \in \blacksquare(V_0^\top)$. Une majoration du domaine $[s]$ des s solutions de $\Upsilon s = b$ sous la contrainte $s \in \mathbf{B}^p$ est donc : $[s] = c_s + \blacksquare(R_s)$ avec $c_s = V_1 S_1^{-1} U_1^\top b$ et $R_s = V_0 V_0^\top$. Soient c_{s_1} (resp. R_{s_1}), le vecteur (resp. la matrice) constitué (e) des p_1 premières lignes de c_s (resp. R_s). L'intersection de deux zonotopes $Z_1(c_1 + \blacksquare(R_1))$, $Z_2(c_2 + \blacksquare(R_2))$ est majorée par le zonotope $Z_{int}(c_{int} + \blacksquare(R_{int}))$ où $c_{int} = c_1 + R_1 c_{s_1}$ et $R_{int} = R_1 R_{s_1}$.

Le test d'inclusion entre deux zonotopes intervient à la ligne 17 de l'**algorithme 18**, la majoration de l'intersection entre ces deux zonotopes est réalisé à la ligne 18 de l'**algorithme 18**. Une fois le zonotope intersectant obtenu, l'**algorithme 17** permet de récupérer les composantes de la MSBP, à savoir $A, [r]$, et $[v]$.

Remarque 5 L'**algorithme 16** fournit un résultat identique en calculant l'ensemble $[s_2] = c_{s_2} + \blacksquare(R_{s_2})$ des s_2 déduit des $[s]$, au lieu de $[s_1]$. La matrice de projection P_2 (voir ligne 7 de l'**algorithme 16**) permettant d'extraire les p_2 dernières composantes de $[s]$ est alors de la forme $P_2 = [0 \quad I]$ et le majorant $c_{int} + \blacksquare(R_{int})$ de l'intersection s'obtient par : $c_2 + R_2 [s_2] = (c_2 + R_2 c_{s_2}) + \blacksquare(R_2 R_{s_2})$ avec $c_{s_2} = P_2 c_s$ et $R_{s_2} = P_2 R_s$.

Algorithme 16 : Majoration de l'intersection de deux zonotopes

```

input  :  $Z_1(c_1 + \blacksquare(R_1)), Z_2(c_2 + \blacksquare(R_2))$ 
output :  $Z_{int}(c_{int}, R_{int})$ 
1 /*  $p_1$ : nombre de segments pour  $Z_1$  */
2 /*  $p_2$ : nombre de segments pour  $Z_2$  */
3 /* Résoudre  $\Upsilon s = b$ , avec  $\Upsilon = [R_1 \quad -R_2]$ ,  $b = (c_2 - c_1)$ ,  $p = p_1 + p_2$  et  $s \in \mathbf{B}^p$  */
4  $U \cdot S \cdot V^\top := \text{svd}([R_1 \quad -R_2]) = [U_1] \cdot [S_1 \quad 0_{1 \times p_2}] \cdot \begin{bmatrix} V_1^\top \\ V_0^\top \end{bmatrix}$ ;
5  $c_s = V_1^\top S_1^{-1} U_1^\top b$ ;
6  $R_s = V_0 V_0^\top$ ;
7  $P_1 = [I_{p_1} \quad 0_{p_1 \times p_2}]$ ;
8  $c_{s_1} = P_1 c_s$ ;
9  $R_{s_1} = P_1 R_s$ ;
10  $c_{int} = c_1 + R_1 c_{s_1}$ ;
11  $R_{int} = R_1 R_{s_1}$ ;
12 return  $Z_{int}(c_{int}, R_{int})$ 

```

Algorithme 17 : Zonotope2ARV

```

Input  :  $Z(c, R)$ 
Output :  $A, [r], [v]$ 
1  $A := R(:, 1 : n)$ ;
2  $[r] := \mathbf{B}^n$ ;
3  $[v] := c + R(:, (n+1) : (2n)) \mathbf{B}^n$ ;

```

Nous pouvons maintenant résumer l'**algorithme 18** qui intègre les méthodes précédentes. L'algorithme démarre avec la liste \mathcal{L}_j^F , puis nous initialisons la liste courante \mathcal{L} avec \mathcal{L}_j^F . Tant que la liste \mathcal{L} non vide, pour chaque élément \mathcal{L}_i de la liste \mathcal{L} , nous calculons, grâce à notre algorithme d'atteignabilité non linéaire hybride (**algorithme 11**), l'ensemble \mathcal{L}_{j+1}^F : cette première étape est celle de la prédiction.

Pour la phase de correction, nous initialisons la liste courante \mathcal{L}_{j+1}^{run} avec la liste \mathcal{L}_{j+1}^F . Tant que la liste \mathcal{L}_{j+1}^{run} est non vide, pour chaque élément Z_{j+1}^F de la liste \mathcal{L}_{j+1}^{run} , nous inversons les mesures $[y_{j+1}]$ à l'aide de SIVIA. L'ensemble solution obtenu par SIVIA est contenu dans la liste \mathcal{L}^{inv} . Pour chaque élément Z_{j+1}^{inv} de \mathcal{L}^{inv} , nous vérifions l'intersection entre les zonotopes Z_{j+1}^F et Z_{j+1}^{inv} (voir l'**algorithme 15**). Si l'intersection est non vide, l'**algorithme 16** fournit l'intersection entre Z_{j+1}^F et Z_{j+1}^{inv} . Finalement la liste \mathcal{L}_{j+1}^{Fc} , c'est-à-dire l'ensemble atteignable hybride corrigé, est mis à jour avec la liste \mathcal{L}_{j+1}^{run} .

Algorithme 18 : Observation des systèmes hybrides (*Contracteur*[†])

```

input  :  $\mathcal{L}_j^F, \Delta_T = t_{j+1} - t_j, \text{hybrid\_reach}(), [y_{j+1}], n, \epsilon_d$ 
output :  $\mathcal{L}_{j+1}^{Fc}$ 
1   $\mathcal{L} := \mathcal{L}_j^F$ ;
3  while  $\mathcal{L} \neq \emptyset$  do
4      pick up  $\mathcal{L}$  list element  $(q, t_j, [\chi_j])$ ;
5      Initialization : initialize list  $\mathcal{L}_i \leftarrow (q, t_j, [\chi_j])$ ;
6      /* Forward hybrid reachable set over  $t_j \rightarrow t_{j+1}$  */
7       $\mathcal{L}_{j+1}^F \leftarrow \text{hybrid\_reach}(\mathcal{L}_i, \Delta_T)$ ;
8       $\mathcal{L}_{j+1}^{run} := \mathcal{L}_{j+1}^F$ ;
10     while  $\mathcal{L}_{j+1}^{run} \neq \emptyset$  do
11         pop  $Z_{j+1}^F$  from  $\mathcal{L}_{j+1}^{run}$  (i.e.  $A_{j+1}, [r_{j+1}], [v_{j+1}]$ );
12         /* Correction with data measure at  $t_{j+1}$  */
13          $\mathcal{L}_{inv} = g^{-1}([y_{j+1}])$ ;
14         while  $\mathcal{L}_{inv} \neq \emptyset$  do
15             pop  $Z_{j+1}^{inv}$  from  $\mathcal{L}_{inv}$ ;
16             if  $(Z_{j+1}^{inv} \cap Z_{j+1}^F) \neq \emptyset$  then
17                  $Z_{j+1}^c := Z_{j+1}^{inv} \cap Z_{j+1}^F$ ;
18                  $\mathcal{L}_{j+1}^{run} := (A_{j+1}^c, [r_{j+1}^c], [v_{j+1}^c])$ ;
19             end if
20         end while
21     end while
22 end while
23      $\mathcal{L}_{j+1}^{Fc} := \mathcal{L}_{j+1}^{run}$ ;
24 end while

```

Remarque 6 *L'arithmétique d'intervalles pourrait également être utilisée pour le calcul de l'intersection entre l'ensemble atteignable hybride et l'ensemble obtenu par inversion de mesure. Ces ensembles sont alors représentés par des boîtes et les opérations de détection et de calcul d'intersection sont implémenté directement dans l'outil d'arithmétique d'intervalles PROFIL/BIAS (Knüppel, 1994) que nous avons utilisé dans nos développements informatiques. Nous avons fait le choix de garder la structure MSBP (zonotope particulier) pour tirer profit de notre intégrateur d'équations différentielles qui représente la solution sous la forme MSBP. Nous réduisons ainsi la sur-approximation.*

Remarque 7 *Dans toutes les évaluations numériques de ce chapitre, nous avons choisi de ne pas utiliser notre méthode de fusion des trajectoires car les mesures permettent de contenir le tube de trajectoires et de bien filtrer l'ensemble atteignable.*

5.2.2 EXEMPLE NUMÉRIQUE 1 : SYSTÈME MASSE-RESSORT HYBRIDE

Considérons le système masse-ressort hybride (voir la figure 5.5-a) avec 4 modes de fonctionnement $q = 0, 1, 2, 3$ et quatre transitions, donné par :

$$\left\{ \begin{array}{l}
 \text{flow}(0) : \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ \frac{-k}{m}x_1 - \frac{c}{m}x_2 \end{pmatrix} \\
 \text{inv}(0) : \nu_0(x_1, x_2) = x_2 \geq -v_0 \\
 \text{guard}(0) : \gamma_1(x_1, x_2) = x_2 < -v_0 \\
 \text{reset}(0) : \rho_0(x_1, x_2) = L \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} \\
 \text{flow}(1) : \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ \frac{-k}{m}x_1 - \frac{c}{m}x_2 \end{pmatrix} \\
 \text{inv}(1) : \nu_1(x_1, x_2) = x_2 \leq -v_0 \wedge x_2 \geq v_0 \\
 \text{guard}(1) : \gamma_1(x_1, x_2) = x_2 > -v_0 \\
 \text{reset}(1) : \rho_1(x_1, x_2) = L \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} \\
 \text{flow}(2) : \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ \frac{-k}{m}x_1 - \frac{c}{m}x_2 \end{pmatrix} \\
 \text{inv}(2) : \nu_2(x_1, x_2) = x_2 \leq v_0 \\
 \text{guard}(2) : \gamma_1(x_1, x_2) = x_2 > v_0 \\
 \text{reset}(2) : \rho_2(x_1, x_2) = L \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} \\
 \text{flow}(3) : \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ \frac{-k}{m}x_1 - \frac{c}{m}x_2 \end{pmatrix} \\
 \text{inv}(3) : \nu_1(x_1, x_2) = x_2 \leq v_0 \\
 \text{guard}(3) : \gamma_1(x_1, x_2) = x_2 < v_0 \\
 \text{reset}(3) : \rho_3(x_1, x_2) = L \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} l_1 \\ l_2 \end{pmatrix}
 \end{array} \right. \quad (5.13)$$

Les états continus sont décrits par deux variables (x_1, x_2) , où x_1, x_2 représentent respectivement la position et la vitesse du système masse-ressort. On a pour le saut discret $L = I_2^6$, $l_1 = l_2 = 0$. Les coefficients sont réglés à : $k = 2.1, m = 1, c = 0.63$.

Soient $p_1 = \frac{k}{m}, p_2 = \frac{c}{m}$ les deux paramètres du système. Les conditions initiales sur les états continus sont $x_1 \in [1.1, 1.11] > 0, x_2 \in [-0.11, -0.1]$, le ressort est donc initialement allongé et les paramètres donnés par $p_1 = 2.1$ et $p_2 = 0.63$.

Le système masse-ressort hybride fonctionne de la manière suivante (voir la figure 5.5-b) : initialement, le système démarre dans le mode q_0 avec l'amortisseur déconnecté ($c_1 = 0$). Ainsi en appliquant une force \vec{F} sur la masse m quand la vitesse atteint un certain seuil $-v_0$ (voir la

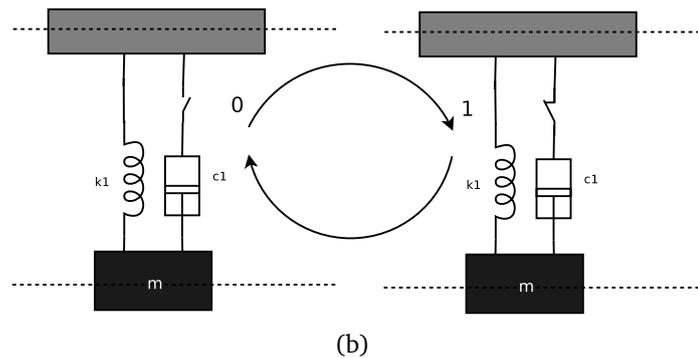
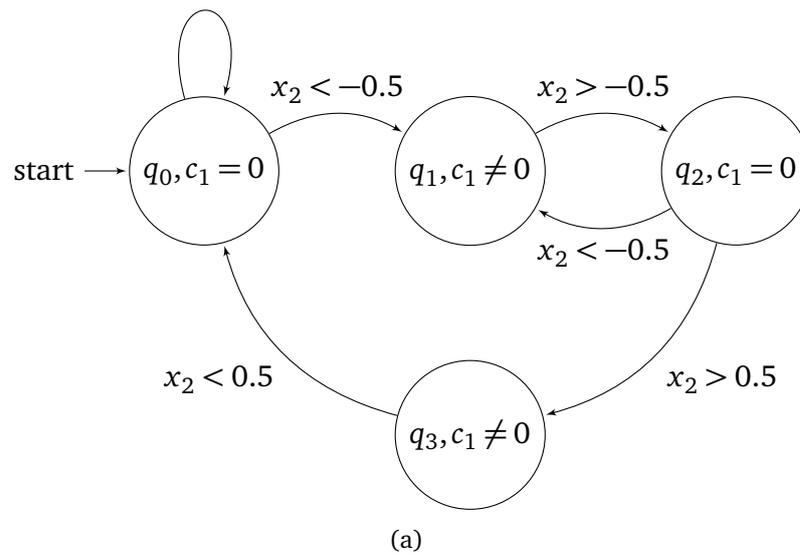


FIGURE 5.5 – Automate hybride du système masse-ressort et Schéma du fonctionnement du système masse-ressort hybride

figure 5.5-a), l'amortisseur $c_1 \neq 0$ est connecté afin de réduire la vitesse et le système bascule dans le mode q_2 . Et à partir du mode q_3 , quand la vitesse est en dessous du seuil v_0 , l'amortisseur est déconnecté et le système revient au mode q_0 et ainsi de suite.

Donc ce système fonctionne avec deux modes possibles 0 et 1 pour ($c_1 \neq 0$ et $c_1 = 0$) respectivement. Pour faciliter le codage informatique, nous avons considéré que la séquence ($q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow q_3 \rightarrow q_0$) correspond respectivement à la séquence ($0 \rightarrow 1 \rightarrow 0 \rightarrow 1 \rightarrow 0$).

Les pseudo-mesures de la position x_1 ont été générées avec les conditions initiales sur le vecteur d'état continu $x_1 \in [1.1, 1.11]$, $x_2 \in [-0.11, -0.1]$ avec les paramètres $p_1 = 2.1$ et $p_2 = 0.63$ et les hypothèses sur les bornes du bruit $\mathbb{E}_j = [-0.01, 0.01]$. L'horizon temporel est de

4.4UT . Les algorithmes Hybrid-Transition (**algorithme 11**) et φ^{QR} (**algorithme 3**) sont réglés comme suit : le pas de temps est choisi constant $h = 0.1$ et l'intervalle de temps est bissecté jusqu'au seuil $\varepsilon_T = 0.005$.

5.2.2.1 EXPÉRIENCE 1 : MODE INITIAL CONNU ET ÉTAT CONTINU INITIAL INCONNU

Pour la première expérience, nous supposons que le mode initial est connu, et que l'état de départ est incertain $x_1 \in [0.75, 1.5]$, $x_2 \in [-0.2, 0.1]$ dans le mode de départ. Aussi les paramètres $p_1 \in [2.1, 2.2]$ et $p_2 \in [0.63, 0.64]$ sont considérés comme incertains. Les résultats de cette expérience (voir figure 5.6) montrent clairement que notre observateur hybride est capable de reconstruire les modes actifs et les variables d'états. Ces résultats montrent que la méthode que nous proposons est efficace. Ce cas est finalement assez simple car le mode de départ est connu, nous verrons dans la deuxième expérience que les temps de calcul augmentent de manière significative (8 fois plus grand) avec le mode de départ inconnu.

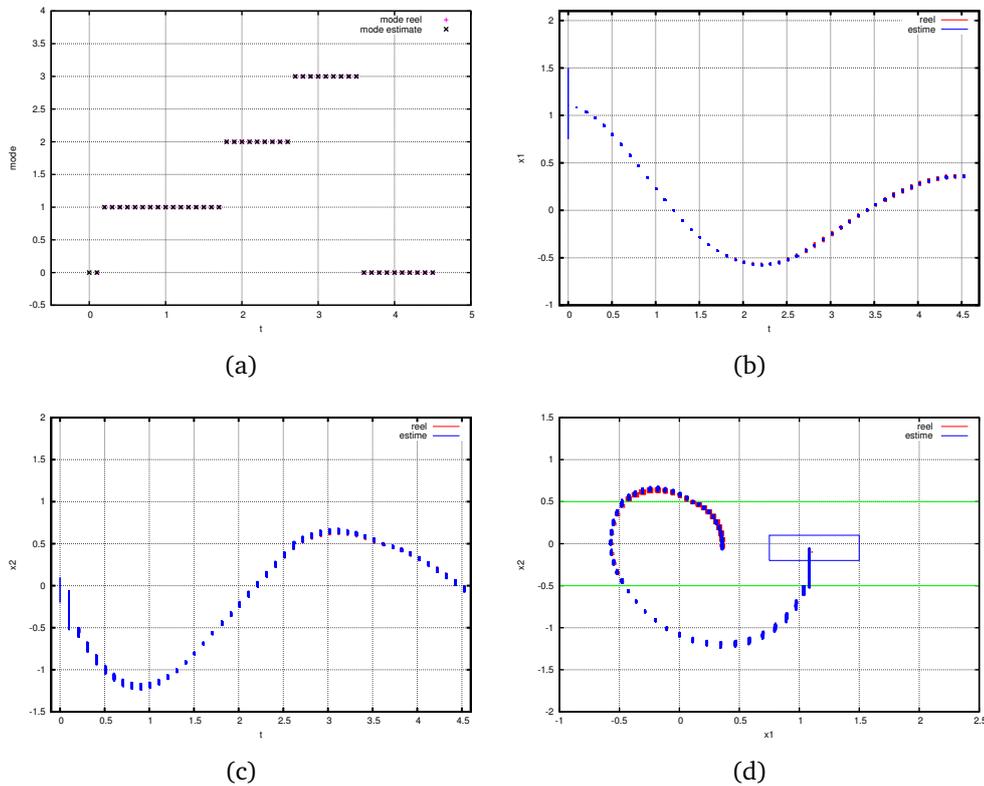


FIGURE 5.6 – Expérience 1 : Estimation du mode (a), l'historique des variables x_1, x_2 estimées en fonction du temps (b)-(c) et la projection de l'ensemble atteignable dans le plan de phase (d) $x_2 \times x_1$, temps CPU=8.012s

5.2.2.2 EXPÉRIENCE 2 : MODE INITIAL ET ÉTAT CONTINU INITIAL INCONNU

Pour la deuxième expérience, nous supposons que le mode initial est inconnu, c'est-à-dire que l'estimateur prédictif/correcteur démarre avec tous les modes possibles du système, et que les conditions initiales de départ sont incertaines $x_1 \in [1, 1.3], x_2 \in [-1, 1]$ pour chaque mode. La vitesse étant intimement liée aux différentes transitions entre les modes, nous avons choisis $x_2 \in [-1, 1]$ de façon à couvrir tous les modes possibles (q_0, q_1, q_2, q_3) au départ de la simulation. Nous avons choisis aussi les paramètres $p_1 = [2.1, 2.2]$ et $p_2 = [0.63, 0.64]$ incertains pour cette expérience.

Les figures 5.7 montrent l'estimation des modes avec les modes de départ possibles $\{q_0, q_1, q_2, q_3\}$, on voit que notre observateur hybride reconstruit le mode actif correctement. Aussi on arrive à reconstruire les vecteurs d'état continu consistants avec les mesures. Ces résultats confirment l'efficacité de notre approche pour l'observation ensembliste des systèmes hybrides dans un contexte à erreur bornée.

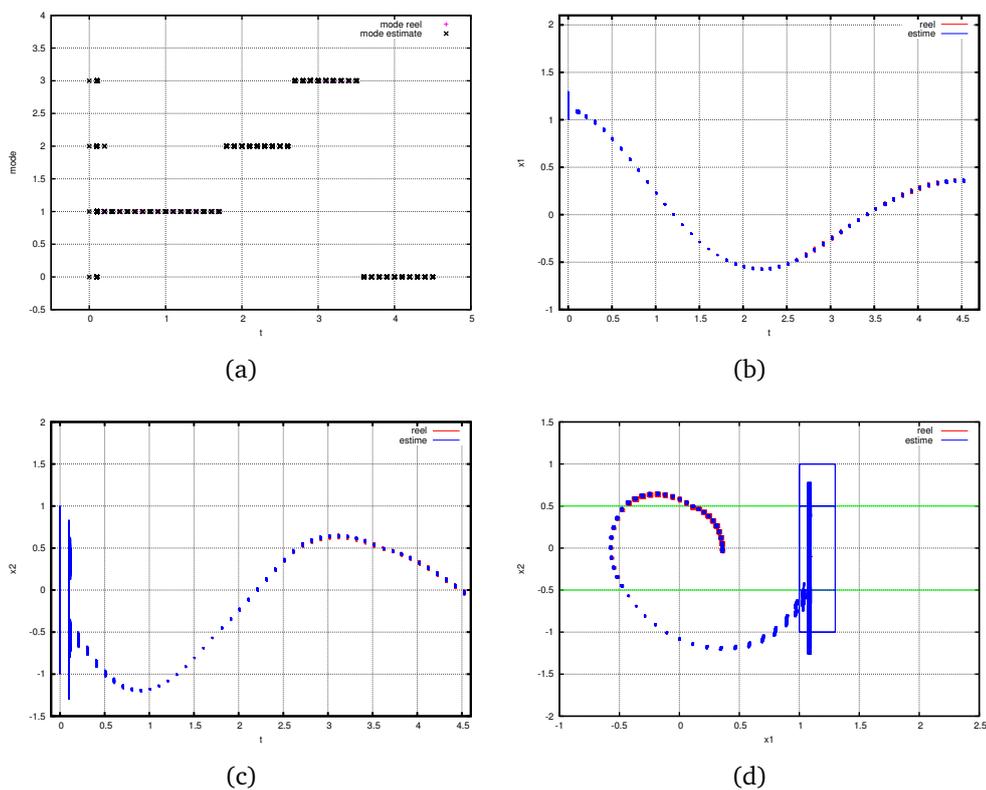


FIGURE 5.7 – Expérience 2 : Estimation du mode (a), l'historique des variables x_1, x_2 estimées en fonction du temps (b)-(c) et la projection de l'ensemble atteignable dans le plan de phase (d) $x_2 \times x_1$, temps CPU=83.888s

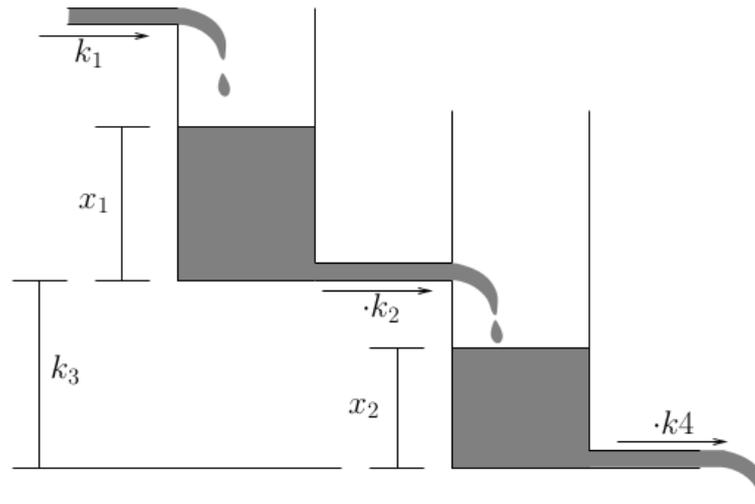


FIGURE 5.8 – Le système à deux réservoirs

5.2.3 EXEMPLE NUMÉRIQUE 2 : SYSTÈME À DEUX RÉSERVOIRS

Nous étudions le système à deux réservoirs (Stursberg et al., 1996) utilisé comme banc d'essai pour l'évaluation de notre méthode d'estimation d'état hybride. Le système est composé de deux réservoirs, le réservoir haut R_1 et le réservoir bas R_2 connectés entre eux par un tuyau horizontal k_2 situé à une hauteur k_3 du bas du réservoir R_1 . Par ailleurs, R_1 est alimenté en eau par une pompe assurant un débit constant k_1 et l'eau peut être évacuée du système par le bas du réservoir R_2 grâce à un tuyau horizontal k_4 . En fonction du niveau d'eau x_2 dans le réservoir R_2 par rapport à la hauteur entre les deux réservoirs k_3 , on a deux modes : "mode_haut" et "mode_bas". Le système, représenté sur la figure 5.8, est modélisé par

Pour $x_2 > k_3$: mode_haut

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} k_1 - k_2 \sqrt{x_1 - x_2 + k_3} \\ k_2 \sqrt{x_1 - x_2 + k_3} - k_4 \sqrt{x_2} \end{pmatrix}$$

Pour $x_2 \leq k_3$: mode_bas

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} k_1 - k_2 \sqrt{x_1} \\ k_2 \sqrt{x_1} - k_4 \sqrt{x_2} \end{pmatrix}$$

où $k_1 = 0.75, k_2 = 1, k_3 = 0.5, k_4 = 1$.

Dans cette expérience, le mode initial et les états continus sont inconnus. Nous avons procédé à la mesure de la variable x_1 . Les pseudo-données ont été obtenues à partir des conditions initiales $x_1 = 0.3, x_2 = 0.6$ avec les paramètres $k_1 = 0.75, k_2 = 1, k_3 = 0.5, k_4 = 1$ et les hypothèses sur les bornes du bruit $\mathbb{E}_j = [-0.01, 0.01]$. L'horizon temporel est de 7.5. Les algorithmes

Hybrid-Transition (**algorithme 11**, page 101) et φ^{QR} (**algorithme 3**, page 32) sont réglés comme suit : le pas de temps est choisi constant $h = 0.05$ et l'intervalle de temps est bisecté jusqu'au seuil $\varepsilon_T = 0.05$. Le label "mode_haut" correspond au mode 0 et le label "mode_bas" correspond au mode 1 (voir la figure 5.9-a). Nous souhaitons reconstruire la séquence de commutation et les trajectoires des composantes continues à partir de trois mesures de x_1 aux instants $t = 0.1$, $t = 0.85$, et $t = 5.45$. Les conditions initiales sont incertaines $x_1 \in [0.285, 0.315]$, $x_2 \in [0.45, 0.65]$ pour les deux modes 0 et 1. Les figures 5.9 illustrent les résultats obtenus, elles montrent que les états continus estimés convergent vers les états réels. L'état discret est correctement estimé, aussi on remarque l'ensemble des modes réels est inclus dans l'ensemble des mode estimés (voir la figure 5.9-a). Notre approche permet de reconstruire la séquence de commutation et les trajectoires des composantes continues, donc l'observateur hybride remplit l'objectif avec une identification du mode actif et une estimation, en temps fini, de l'état continu

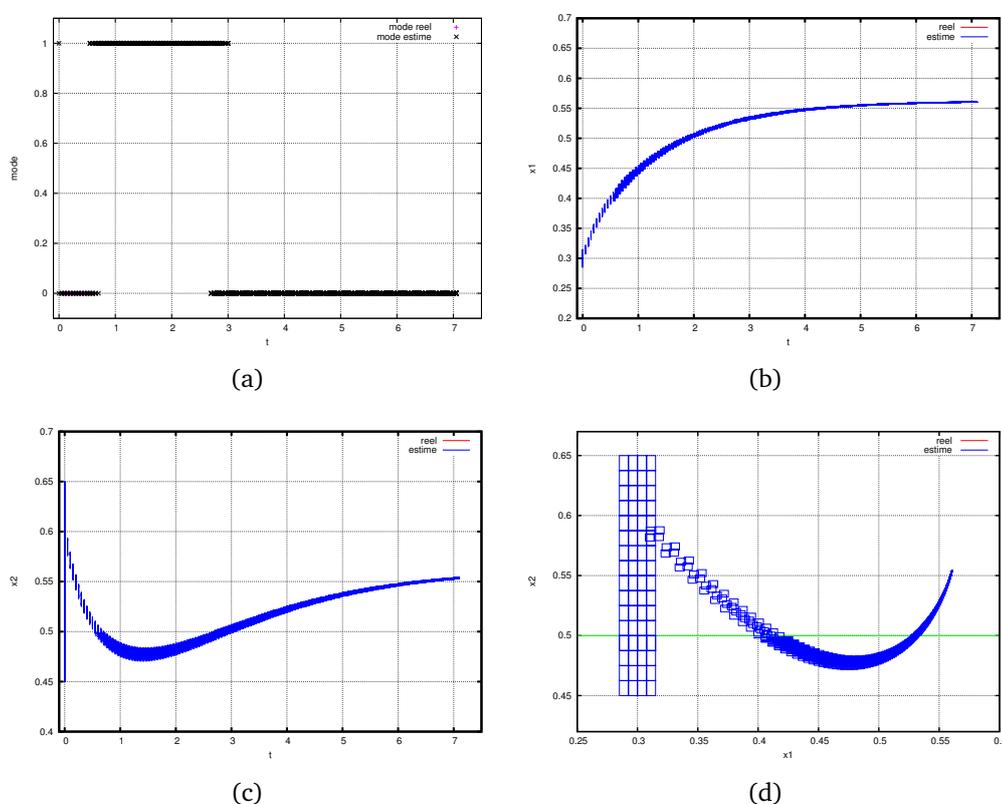


FIGURE 5.9 – Estimation du mode (a), l'historique des variables x_1, x_2 estimées en fonction du temps (b)-(c) et la projection de l'ensemble atteignable dans le plan de phase (d) $x_2 \times x_1$, Temps CPU=1092.66s

5.2.4 EXEMPLE DE SCÉNARIO DE DÉFAUT

Afin de tester la méthode de détection de défauts, un biais est ajouté sur la mesure pour simuler un défaut. Seule la position x_1 est mesurée. Par conséquent, les défauts sont introduits

sur la mesure de la position. Le défaut apparaît à l'instant ($t = 2.6s$).

Nous considérons que les paramètres $p_1 \in [2.1, 2.1]$ et $p_2 \in [0.63, 0.63]$ sont nominaux. Nous considérons le paramètre p_1 comme une entrée des équations différentielles (voir équation 5.13) dans chaque mode, ainsi à un instant donné de la grille de temps ($t = 2.6s$), nous injectons un biais d'amplitude $\delta = 0.5$ dans p_1 qui se traduit comme une faute ($p_1 + \delta$). Les pseudo-données sont ainsi générées. Pour détecter la faute, il suffit que l'intersection entre l'ensemble prédit et l'ensemble obtenu par inversion des mesures soit vide (voir équation 5.1).

Pour cette expérience, nous supposons que le mode initial est connu, et que l'état de départ est incertain $x_1 \in [0.75, 1.5]$, $x_2 \in [-0.2, 0.1]$ dans le mode de départ. Aussi notre estimateur prend comme paramètres initiaux, les paramètres nominaux $p_1 \in [2.1, 2.1]$ et $p_2 \in [0.63, 0.63]$.

La figure 5.10 illustre les résultats obtenus pour la détection de défaut. Les ensembles des états prédits et estimés sont représentés à la fois pour la position et pour la vitesse. Comme nous l'avons montré dans la section 5.2.1, si l'intersection des deux ensembles est vide, alors un défaut est détecté. Le défaut est détecté à l'instant ($t = 2.9s$).

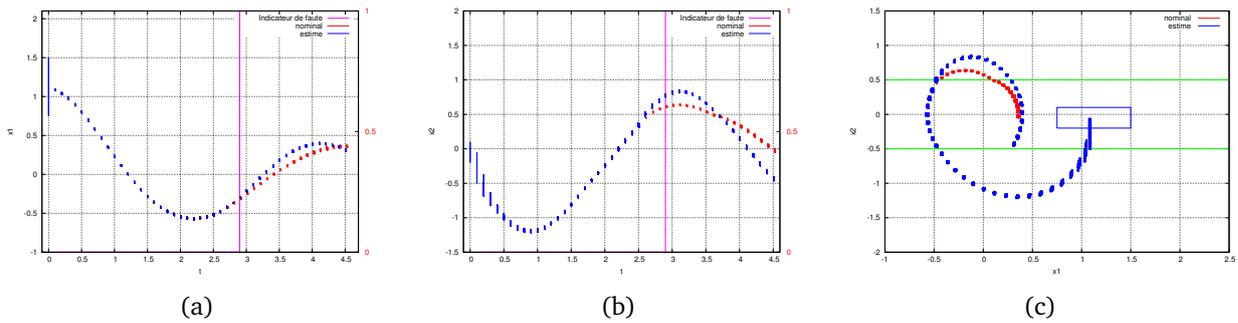


FIGURE 5.10 – Détection de défaut par approche prédiction/correction

5.3 MÉTHODE ENSEMBLISTE POUR L'ESTIMATION PARAMÉTRIQUE DES SHIB

Dans cette section, nous allons étudier le problème de l'estimation des paramètres des systèmes hybrides dans le contexte à "erreur bornée" dans une optique de diagnostic.

En pratique, l'application de l'algorithme d'inversion ensembliste SIVIA ne se limite pas aux seuls modèles décrits par des fonctions explicites en les variables, mais elle s'étend aussi aux cas des modèles du type représentation d'état à temps continu pourvu que nous disposions d'évaluations numériques d'une fonction d'inclusion pour les variables d'état $[x_j]$ aux instants de mesures t_j (Raïssi et al., 2004a; Raïssi, 2004). Ces estimations peuvent être calculées par une des méthodes d'intégration numérique garanties présentées dans le premier chapitre. Ainsi, l'utilisation conjointe d'un partitionneur d'espace initial de recherche des paramètres $\mathcal{P} = [\mathbf{p}]$, de

SIVIA et notre méthode d'atteignabilité hybride non linéaire permet de construire une nouvelle méthode pour l'estimation ensembliste des paramètres d'un système hybride dans un contexte à erreurs bornées.

A noter que pour un vecteur de paramètres dont la largeur est assez grande, notre algorithme d'atteignabilité hybride au bout d'un certain pas de temps diverge. Nous avons donc considéré que si la taille du pavé est inférieur à un seuil ϵ_p (voir ligne 6 **algorithme 22**, page 137), nous pouvons estimer le vecteur de paramètres correspondant, sinon on bissecte ce vecteur de paramètres et l'on met les vecteurs de paramètres résultants dans la liste courante \mathcal{L}_c (voir ligne 35,36 **algorithme 22**), d'où le nom SIVIA-H (SIVIA pour système hybride).

L'utilisation de SIVIA-H requiert l'utilisation d'un test d'inclusion. Nous allons à ce titre proposer deux versions de test d'inclusion que nous définissons ci-après (**algorithme 19** et **algorithme 20**). Le premier test d'inclusion utilise l'arithmétique des intervalles et le second les zonotopes.

5.3.1 TESTS D'INCLUSION

5.3.1.1 TEST D'INCLUSION PAR ARITHMÉTIQUE DES INTERVALLES

Ce test d'inclusion utilise l'arithmétique des intervalles. Étant donné la frontière⁷ F_j (c'est à dire l'ensemble des boîtes solutions) à un instant donné t_j , la frontière contient l ($l = \text{taille}(F_j)$) éléments $[\boldsymbol{x}]_l = \{[z]_l, \hat{z}_l, [v]_l, [r]_l, A_l\}$, et la mesure $[y_j]$ à t_j .

Le test d'inclusion est vrai, si pour tous les éléments de F_j , on a

$$\forall k(1 \leq k \leq l), [g]([z_k^j]) \subseteq [y_j] \quad (5.14)$$

Et le test d'inclusion est faux, s'il existe au moins un seul élément de F_j pour lequel

$$\exists k, (1 \leq k \leq l), [g]([z_k^j]) \cap [y_j] = \emptyset \quad (5.15)$$

avec $[g]$ est la fonction d'inclusion intervalle de g .

Dans le cas où le test d'inclusion est différent de vrai et de faux, le test d'inclusion est incertain (voir **algorithme 19**).

5.3.1.2 TEST D'INCLUSION BASÉ SUR LES ZONOTOPES

Nous utilisons cette fois-ci une alternative pour le test d'inclusion en utilisant les zonotopes. Considérons à nouveau la frontière F_j à un instant donné t_j .

7. Frontière désigne l'ensemble solution (union de boîtes ou de zonotopes) à un instant t_j

Algorithme 19 : Test d'inclusion V.1

```

input :  $F_j, [y_j]$ 
output :  $true, false, uncertain$ 
1 /* Test d'inclusion entre la frontière  $F_j$  et la mesure  $[y_j]$  à l'instant  $j$ 
   */
2 inclus :=0;
3 nointer :=0;
4 for  $k = 1$  to  $l$  do
5    $[z_k^j] := pop(F_j)$ ;
6    $[y_k^j] := [g]([z_k^j])$ ;
7   if  $\forall k (y_k^j \subseteq [y_j])$  then
8     | inclus++;
9   end if
10  if  $([y_k^j] \cap [y_j] == \emptyset)$  then
11    | nointer++;
12  end if
13 end for
14 if  $(inclus == l)$  then
15   | return true ;
16 end if
17 if  $nointer > 0$  then
18   | return false ;
19 end if
20 return uncertain ;

```

Proposition 7 (Vicino and Zappa, 1996) Étant donné $Z = c \oplus RB^p \in \mathbb{R}^n$ et un strip⁸ $\mathcal{S} = |\eta^\top x - d| \leq \sigma$, le support strip (voir figure 5.11) du zonotope est défini par

$$\mathcal{S}_Z = \{x \in \mathbb{R}^n | q_d \leq \eta^\top x \leq q_u\}, \quad (5.16)$$

où q_u et q_d sont définis comme

$$q_u = \max_{x \in Z} \eta^\top x \quad (5.17)$$

$$q_d = \min_{x \in Z} \eta^\top x \quad (5.18)$$

qui sont calculés par

$$q_u = \eta^\top c + \|R^\top \eta\|_1 \quad (5.19)$$

$$q_d = \eta^\top c - \|R^\top \eta\|_1 \quad (5.20)$$

8. Bande en français

Algorithme 20 : Test d'inclusion V.2

```

input :  $F_j, [y_j]$ 
output :  $true, false, uncertain$ 
1 /* Test d'inclusion entre la frontière  $F_j$  et la mesure  $[y_j]$  à l'instant  $j$ 
   */
2 inclus :=0;
3 nointer :=0;
4 for  $k = 1$  to  $l$  do
5    $\{A_k^j, [r_k^j], [v_k^j]\} := pop(F_j)$ ;
6    $Z_k^j := \text{Box2zonotope}(A_k^j, [r_k^j], [v_k^j])$ ;
7    $Z_k^{j*} := [g]([Z_k^j])$ ;
8    $\mathcal{S}_j^i = \{|\eta_j^{i\top} x - d_j^i| \leq \sigma_j^i\}$ ;
9   if  $\forall k (Z_k^{j*} \subseteq \mathcal{S}_j^i)$  then
10    | inclus++;
11  end if
12  if  $(Z_k^{j*} \cap \mathcal{S}_j^i == \emptyset)$  then
13    | nointer++;
14  end if
15 end for
16 if  $(inclus == l)$  then
17 | return true;
18 end if
19 if  $nointer > 0$  then
20 | return false;
21 end if
22 return uncertain;

```

Théorème 6 *Le zonotope, $Z = c \oplus RB^p \in \mathbb{R}^n$ est inclus dans le strip $\mathcal{S}_j = |\eta^\top x - d| \leq \sigma$ lié à la mesure $[y_j]$ si et seulement si :*

$$(q_d \geq d_j - \sigma_j) \wedge (q_u \leq d_j + \sigma_j) \quad (5.21)$$

où $d_j = \text{mid}([y_j])$, $\sigma_j = \text{rad}([y_j])$

Dans le cas où l'on a n strips ($i = 1, \dots, n$) $\mathcal{S}_j^i = |\eta_j^{i\top} x - d_j^i| \leq \sigma_j^i$, on définit le test d'inclusion et l'intersection vide entre le zonotope $Z = c \oplus RB^p$ et les n strips (voir ligne 9 et ligne 12 respectivement **algorithme 20**) en utilisant l'**algorithme 21** avec $n =$ nombre de variables mesurées.

En utilisant le Théorème 2 de la section 2.3.1.3 (page 29), on peut calculer l'image du zonotope Z_k^j par la fonction d'inclusion $[g]$ de g (voir ligne 7 **algorithme 20**)

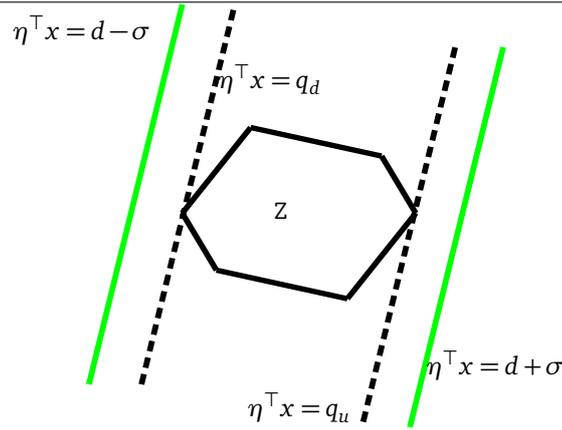


FIGURE 5.11 – le support strip d'un zonotope.

Théorème 7 *Le zonotope $Z = c \oplus RB^p \in \mathbb{R}^n$ n'intersecte pas le strip $\mathcal{S}_j = |\eta^\top x - d| \leq \sigma$ si et seulement si :*

$$(q_d > d_j + \sigma_j) \vee (q_u < d_j - \sigma_j) \quad (5.22)$$

où $d_j = \text{mid}([y_j])$, $\sigma_j = \text{rad}([y_j])$

L'**algorithme 20** implémente les Théorèmes 6 et 7 et considère le zonotope comme incertain lorsque ni l'un ni l'autre ne sont satisfaits.

5.3.2 SIVIA POUR SYSTÈME HYBRIDE (SIVIA-H)

Pour rappel, étant donné le modèle hybride (2.1–2.2–2.3), les mesures (2.7) et les bornes des incertitudes de mesure, le but de l'estimation de paramètres consiste à estimer le vecteur de paramètres \mathbf{p} en déterminant l'ensemble \mathbb{S} de tous les paramètres acceptables.

$$\mathbb{S} = \{\mathbf{p} \in \mathcal{P}_0 \mid (\forall t \in [t_0, T_{end}], \mathcal{R}([t_0, T_{end}]; \mathcal{Z}_0) \wedge (\forall t_j \in \{t_1, t_2, \dots, t_n\}, g_q(x, \mathbf{p}, t) \in \mathbb{Y}_j)\} \quad (5.23)$$

Notre algorithme (**algorithme 22**) génère trois listes $\mathcal{L}_a, \mathcal{L}_r, \mathcal{L}_i$.

La première \mathcal{L}_a contient tous les pavés $[\mathbf{p}_i] \subset [\mathbf{p}]$ pour lesquels le test d'inclusion est vrai (voir figure 5.12(a)) et 5.12(d)), c'est-à-dire :

$$\forall t_j \in \{t_1, t_2, \dots, t_n\}, k = 1, 2, \dots, l, \mathcal{R}([t_0, T_{end}]; \mathcal{Z}_0) \wedge g_q([x_k^j], [\mathbf{p}_i]) \in \mathbb{Y}_j \quad (5.24)$$

Notons qu'à un pavé $[\mathbf{p}_i]$ correspondent plusieurs boîtes $[x_k^j]$ ou zonotopes Z_k^j sur lesquels est appliqué le test d'inclusion illustré par la figure 5.12.

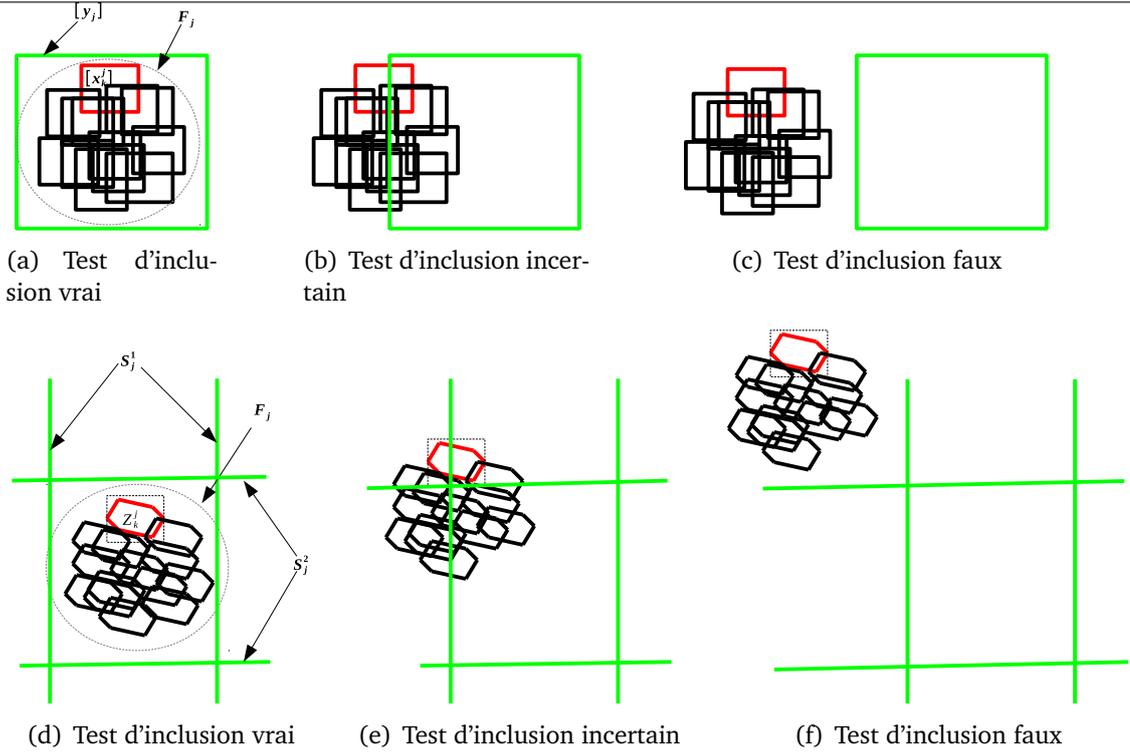


FIGURE 5.12 – (en haut) Test d'inclusion par arithmétique des intervalles (a-b-c) pour la frontière F_j à l'instant t_j , (en bas) test d'inclusion avec les zonotopes (d-e-f) pour la frontière F_j à l'instant t_j

L'union de tous les pavés de la première liste \mathcal{L}_a donne une sur-approximation intérieure \mathbb{S} de l'ensemble S .

La seconde liste \mathcal{L}_r contient tous les pavés pour lesquels le test d'inclusion est faux (voir figure 5.12(c) et 5.12(f)) c'est-à-dire que les pavés pour lesquels la structure du modèle, les données expérimentales et les bornes des erreurs sont inconsistantes :

$$\exists t_j \in \{t_1, t_2, \dots, t_n\}, k = 1, 2, \dots, l, \mathcal{R}([t_0, T_{end}]; \mathcal{X}_0) \wedge g_q([x_k^j], [\mathbf{p}_i]) \cap \mathbb{Y}_j = \emptyset \quad (5.25)$$

où F_j est la frontière et $[x_k^j] = F_j(k)$ une boîte solution de F_j avec $k = 1, 2, \dots, l = \text{length}(F_j)$.

Dans les situations où nous ne pouvons pas conclure sur le résultat du test d'inclusion, on dit que le pavé est incertain (voir figure 5.12(b) ou 5.12(e)), car ni la condition (5.24), ni la condition (5.25) ne sont satisfaites. Pour lever cette ambiguïté, l'algorithme *SIVIA-H* bissecte ce pavé $[\mathbf{p}_i]$, et ainsi teste encore les deux nouveaux pavés. Le choix de la direction de bisection est basé par exemple sur la taille du pavé à partitionner (Jaulin and Walter, 1993). Nous répétons ce processus jusqu'à un seuil fixé a priori ϵ . Ainsi, la troisième liste, notée \mathcal{L}_i , contient tous les

Algorithme 21 : Test d'inclusion et test d'intersection entre zonotope et n strips

```

input :  $Z, \mathcal{S}_j^i, n$ 
output :  $true, false$ 
1 inclus := 0;
2 noninclus := 0;
3 for  $i = 1$  to  $n$  do
4    $q_d^i$  et  $q_u^i$  selon les formules 5.20 et 5.19;
5   if  $(q_d^i > d_j^i - \sigma_j^i) \wedge (q_u^i < d_j^i + \sigma_j^i)$  then
6     | inclus++;
7   end if
8   if  $(q_d^i > d_j^i + \sigma_j^i) \vee (q_u^i < d_j^i - \sigma_j^i)$  then
9     | noninclus++;
10  end if
11 end for
12 if  $(inclus == n)$  then
13   | return true ;
14 end if
15 if  $noninclus > 0$  then
16   | return false ;
17 end if

```

pavés indéterminés et dont la taille est plus petite que ϵ .

Pour résumer le principe de notre approche pour l'estimation des paramètres des SHIB, nous considérons en premier lieu un pavé de paramètre $[\mathbf{p}_i]$, pour lequel nous calculons l'ensemble atteignable hybride sur l'horizon $[t_0, t_{end}]$ en utilisant notre algorithme d'atteignabilité hybride non linéaire (**algorithme 11**, page 101). Si la condition (5.24) est satisfaite, le pavé paramètre est consistant avec les mesures, ainsi on le met dans la liste \mathcal{L}_a . Par contre, si la condition (5.25) est satisfaite, le paramètre n'est pas consistant avec les mesures et il est donc mis dans la liste \mathcal{L}_r . Si aucune des deux conditions n'est satisfaite et que la taille du pavé est inférieure à un seuil ϵ , le pavé paramètre est mis dans la liste \mathcal{L}_i . Sinon il est bisecté et l'on met les deux pavés résultants dans la liste courante \mathcal{L}_c (voir ligne 27,28 **algorithme 22**).

5.3.3 EXEMPLE NUMÉRIQUE : SYSTÈME MASSE-RESSORT HYBRIDE

Nous considérons le système masse-ressort hybride présenté en Section 5.2.2. Considérons trois paramètres $p_1 = \frac{k}{m}$, $p_2 = \frac{c}{m}$ et le seuil de la condition de garde $p_3 = v_0$ que nous voulons identifier. Pour illustrer la performance de notre approche, nous considérons deux cas d'études.

Dans le premier scénario, nous allons identifier $p_1 = \frac{k}{m}$ et $p_2 = \frac{c}{m}$, avec $p_3 = v_0$ connu. Et

dans le second scénario, nous allons identifier $p_1 = \frac{k}{m}$ et $p_3 = v_0$ avec $p_2 = \frac{c}{m}$ connu. Nous choisissons de mesurer la position du système car le système masse-ressort hybride est gouverné par la vitesse. Nous avons la possibilité d'obtenir 45 mesures de la position sur l'horizon de temps $T_{end} = 4.4 \text{ unitdetemps}$, mais pour illustrer la robustesse de notre approche, seules 12 mesures ont été considérées. Les conditions initiales pour les vecteurs d'état sont connues : $x_1 \in [1, 1.1]$, $x_2 \in [0.1, 0.1]$.

Les pseudo-données ont été collectées avec le vecteur de paramètre physique $\mathbf{p}^* = (2.1 \times 0.63 \times 0.5)$ et les hypothèses sur les bornes du bruit sont $\mathbb{E}_j = [-0.01, 0.01]$. La boîte de recherche initiale est prise est : $[\mathcal{P}_0] = [1.5; 3] \times [0; 1.5]$ pour le premier scénario et pour le second scénario $[\mathcal{P}_0] = [1.5; 3] \times [0.35; 0.65]$, avec $\epsilon = 0.01$

SCÉNARIO 1 L'approximation intérieure et la couche d'incertitude obtenues sont illustrées dans la figure 5.13-a pour le test d'inclusion en utilisant les intervalles et la figure 5.13-b pour le test d'inclusion en utilisant les zonotopes dans l'espace des paramètres $p_2 \times p_1$. Nous avons aussi affiché l'ensemble atteignable dans le plan de phase, c'est-à-dire la vitesse vs la position pour tous les paramètres qui sont consistants avec les mesures (voir les figures 5.13-c et 5.13-d).

	Volume (\mathcal{L}_a)	NBox(\mathcal{L}_a)	Volume (\mathcal{L}_i)	NBox(\mathcal{L}_i)	CPU times
Test d'inclusion V.1	1.347×10^{-3}	39	1.860×10^{-3}	127	126.23mn
Test d'inclusion V.2	1.347×10^{-3}	40	1.948×10^{-3}	133	154.92mn

TABLE 5.1 – Comparaison entre les deux méthodes pour le test d'inclusion (scénario 1)

SCÉNARIO 2 L'approximation intérieure et la couche d'incertitude obtenues sont illustrées dans la figure 5.14-a pour le test d'inclusion en utilisant les intervalles et la figure 5.14-b pour le test d'inclusion en utilisant les zonotopes dans l'espace des paramètres $p_2 \times p_1$. Nous avons aussi affiché l'ensemble atteignable dans le plan de phase, c'est-à-dire la vitesse vs la position pour tous les paramètres qui sont consistants avec les mesures (voir les figures 5.14-c et 5.14-d).

	Volume (\mathcal{L}_a)	NBox(\mathcal{L}_a)	Volume (\mathcal{L}_i)	NBox(\mathcal{L}_i)	CPU times
Inclusion test V.1	1.875×10^{-3}	68	2.734×10^{-3}	280	37,33mn
Inclusion test V.2	1.796×10^{-3}	68	2.822×10^{-3}	289	39,54mn

TABLE 5.2 – Comparaison entre les deux méthodes pour le test d'inclusion (scénario 2)

Ces figures (5.13,5.14) montrent clairement que notre approche peut résoudre d'une manière très naturelle le problème d'estimation ensembliste des paramètres gouvernant les transitions discrètes dans les systèmes hybrides, c'est-à-dire l'identification de la garde ou l'hyper-surface de commutation du système dynamique non linéaire. Comme le montrent les tableaux (5.1,5.2), les deux méthodes de tests d'inclusion donnent des résultats similaires. Nous pensons cependant qu'en dimensions élevées, le test d'inclusion utilisant les zonotopes offre des performances meilleures que celui utilisant l'arithmétique d'intervalles.

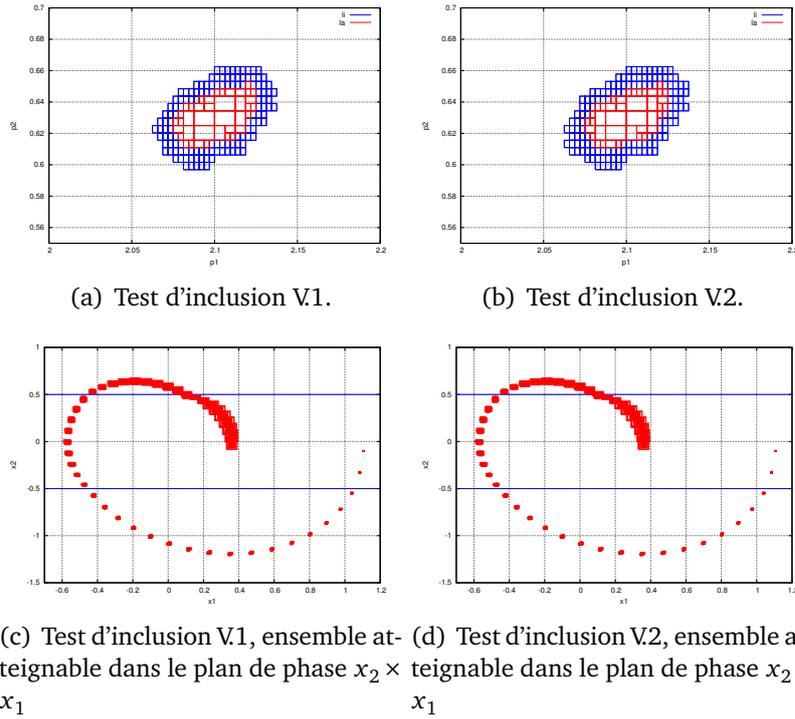


FIGURE 5.13 – Approximation intérieure de la solution (boîte rouge) dans l'espace des paramètres $p_2 \times p_1$, la couche d'incertitude (boîte bleu) et la projection de l'ensemble atteignable dans le plan de phase $x_2 \times x_1$

5.3.4 EXEMPLE DE SCÉNARIOS DE DÉFAUT

Nous allons considérer deux cas d'études. Pour le premier cas, on considère qu'il y a une intersection non vide entre l'ensemble des paramètres estimés et l'ensemble des paramètres nominaux donc pas de défaut, et pour le second cas on suppose que l'intersection entre l'ensemble des paramètres estimés et l'ensemble des paramètres nominaux est vide.

5.3.4.1 CAS 1

Les pseudo-données ont été collectées avec le vecteur de paramètres en défaut $\mathbf{p}^f = (2.075 \times 0.6 \times 0.5)$ et les hypothèses sur les bornes du bruit sont $\mathbb{E}_j = [-0.01, 0.01]$. Le vecteur de paramètre nominal est $\mathbf{p}^* = (2.1 \times 0.63 \times 0.5)$. La boîte de recherche initiale est $[\mathcal{P}_0] = [1.5; 3] \times [0; 1.5]$ avec $\epsilon = 0.01$. Pour rappel, nous commençons par déterminer l'ensemble de valeurs de paramètres nominaux \mathbb{P}_N avec le vecteur de paramètre nominal $\mathbf{p}^* = (2.1 \times 0.63 \times 0.5)$, ensuite nous estimons l'ensemble \mathbb{S} via la formule (2.10) avec des données de défauts sur un horizon donné, ainsi si $\mathbb{S} \cap \mathbb{P}_N = \emptyset$, un défaut sera détecté.

Pour ce premier scénario, comme on peut le remarquer sur la figure 5.15-a, l'intersection

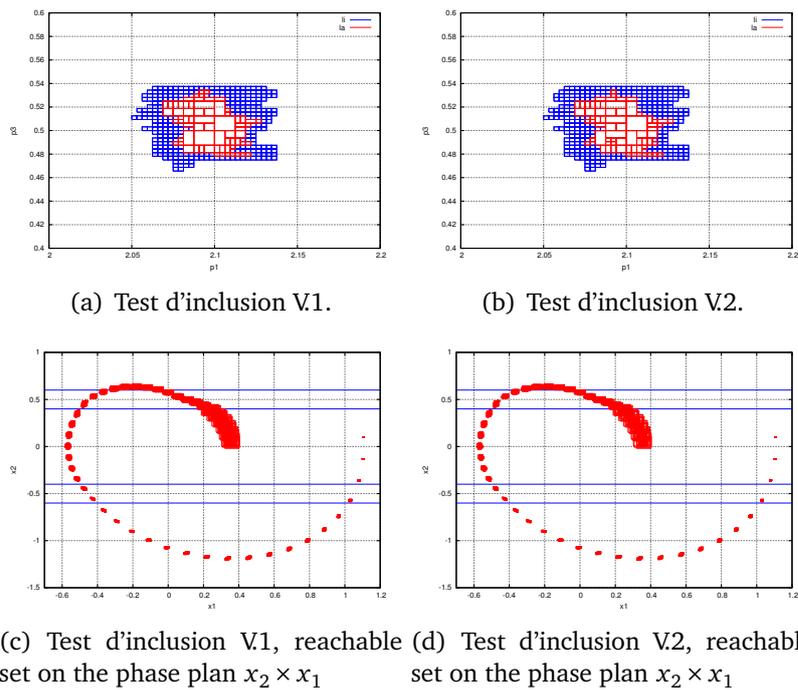


FIGURE 5.14 – Approximation intérieure de la solution (boîte rouge) dans l'espace des paramètres $p_3 \times p_1$, la couche d'incertitude (boîte bleu) et la projection de l'ensemble atteignable dans le plan de phase $x_2 \times x_1$

entre l'ensemble des valeurs de paramètres nominaux (boîtes en couleur grise) et l'ensemble des valeurs de paramètres estimés (boîtes en couleur verte) est non vide, donc il n'y a pas de défaut.

5.3.4.2 CAS 2

Cette fois-ci, les pseudo-données ont été collectées avec le vecteur de paramètres en défaut $\mathbf{p}^f = (2.05 \times 0.67 \times 0.5)$ et les hypothèses sur les bornes du bruit sont $\mathbb{E}_j = [-0.01, 0.01]$. Le vecteur de paramètre nominal est $\mathbf{p}^* = (2.1 \times 0.63 \times 0.5)$. La boîte de recherche initiale est : $[\mathcal{P}_0] = [1.5; 3] \times [0; 1.5]$ avec $\epsilon = 0.01$.

Comme illustré sur la figure 5.15-b, on remarque que l'intersection entre l'ensemble des valeurs de paramètres nominaux (boîtes en couleur grise) et l'ensemble des valeurs de paramètres estimés (boîtes en couleur verte) est vide, donc on détecte aucun défaut.

5.4 CONCLUSION

Dans ce chapitre, nous avons utilisé notre méthode d'atteignabilité non linéaire hybride afin de proposer un observateur hybride ensembliste.

Nous avons commencé par motiver la détection de défaut et le diagnostic pour les systèmes hybrides. Deux approches pour la détection de défaut et le diagnostic existent qui reposent soit sur un observateur hybride de type prédiction/correction ou sur une approche paramétrique.

Nous avons proposé en premier lieu un observateur basé sur l'approche prédiction/correction. La phase de prédiction consiste à calculer l'ensemble atteignable hybride à un instant t_{j+1} en partant de la solution à t_j ; cette étape produit alors un ensemble χ_{j+1}^F . La correction consiste à contracter ce dernier ensemble en utilisant la mesure disponible à t_{j+1} . On obtient alors un ensemble χ_{j+1}^c consistant avec la mesure à l'instant t_{j+1} et la borne d'erreur fixée a priori. Les performances de cet observateur ont été testées sur deux exemples numériques académiques. Les résultats de simulation montrent que l'observateur remplit correctement l'objectif d'observation des systèmes hybrides.

Dans un second temps, nous avons étudié l'estimation des paramètres. Nous avons alors associé notre algorithme d'atteignabilité non linéaire hybride avec SIVIA-H qui est une version de SIVIA dédiée aux systèmes hybrides. Nous avons ainsi pu mettre en place deux tests d'inclusion, l'un utilisant l'arithmétique d'intervalles et l'autre les zonotopes. L'exemple numérique étudié montre que cette méthode permet de résoudre le problème d'estimation avec un temps de calcul raisonnable.

La détection de faute et le diagnostic des systèmes hybrides pourraient ainsi être réalisés en utilisant les deux méthodes proposées dans ce chapitre.

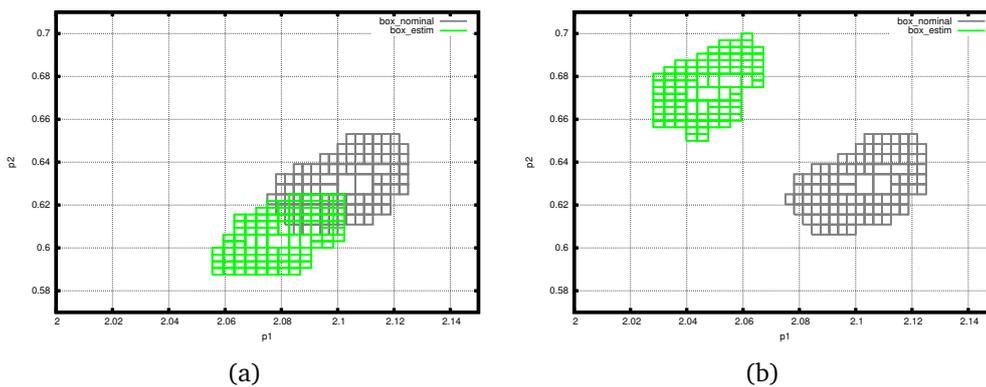


FIGURE 5.15 – Détection de défaut par estimation de paramètres, (a) : scénario de non défaut, intersection non vide entre l'ensemble des paramètres nominaux et estimés ; (b) : scénario de défaut, intersection vide entre l'ensemble des paramètres nominaux et estimés

Algorithme 22 : Estimation de paramètres des systèmes hybride

```

input :  $[p], T_{end}, \Delta_T, hybrid\_reach(), F_{(1:n)}, R_{(1:n)}, \mathcal{L}_m, \epsilon, \epsilon_p, n = \frac{T_{end}}{\Delta_T} + 1$ 
output :  $\mathcal{L}_a, \mathcal{L}_r, \mathcal{L}_i$ 
1  $\mathcal{L}_c \leftarrow [p]$ ;
3 while  $\mathcal{L}_c \neq \emptyset$  do
4    $inclusion := 0$ ;
5    $[p] := pop(\mathcal{L}_c)$ ;
6   if  $w([p]) \leq \epsilon_p$  then
7     /* Calcul de l'ensemble atteignable hybride sur tout l'horizon
8       temporel  $T_{end}$  */
9      $(F_{(1:n)}, R_{(1:n)}) \leftarrow hybrid\_reach([p], \Delta_T, T_{end})$ ;
10    for  $j = 1$  to  $n$  do
11      /*  $F_j$ , Frontier at  $j$  */
12       $test \leftarrow Inclusion-Test(F_j, \mathcal{L}_m(j))$ ;
13      if  $(test == false)$  then
14        break;
15      else if  $(test == true)$  then
16         $inclusion := inclusion + 1$ ;
17      end if
18    end for
19    if  $(test == false)$  then
20       $\mathcal{L}_r \leftarrow [p]$ ;
21    else
22      if  $(inclusion == n)$  then
23         $\mathcal{L}_a \leftarrow [p]$ ;
24      else if  $w([p]) \leq \epsilon$  then
25         $\mathcal{L}_i \leftarrow [p]$ ;
26      else
27         $\{[p]_l, [p]_r\} := Partition([p])$ ;
28         $\mathcal{L}_c \leftarrow [p]_l$ ;
29         $\mathcal{L}_c \leftarrow [p]_r$ ;
30      end if
31    end if
32  end if
33 else
34    $\{[p]_l, [p]_r\} := Partition([p])$ ;
35    $\mathcal{L}_c \leftarrow [p]_l$ ;
36    $\mathcal{L}_c \leftarrow [p]_r$ ;
37 end if
38 end while

```

CONCLUSION ET PERSPECTIVES

Dans ce mémoire nous nous sommes intéressé à l'atteignabilité hybride non linéaire et à l'estimation des systèmes dynamiques hybrides à base de techniques ensemblistes. L'objectif qui a guidé nos travaux a été le développement d'algorithmes génériques dans le but de réaliser la détection de défauts et le diagnostic pour les systèmes hybrides incertains.

Les contributions de la thèse sont multiples. Elles portent sur l'amélioration technique de l'existant pour le passage ensembliste des gardes, la réduction de la sur-approximation introduite par la fonction de reset, et une méthode efficace pour la fusion des trajectoires. L'utilisation conjointe de ces améliorations nous a permis de développer une méthode d'atteignabilité rapide et efficace dédiée aux systèmes hybrides à incertitudes bornées. La méthode d'atteignabilité hybride ainsi développée a été utilisée pour construire un observateur hybride qui nous a permis de reconstruire à partir du modèle hybride, des mesures, et des bornes connues a priori sur les mesures, le mode actif et l'état continu du système hybride. De plus, nous avons mis en place une méthode permettant d'estimer les paramètres inconnus d'un système hybride dans un contexte à erreurs bornées. Cette méthode est la résultante de notre méthode d'atteignabilité hybride et d'une méthode de partitionnement que nous avons nommée SIVIA-H. Les deux méthodes présentées dans ce mémoire (observateur hybride, et estimateur de paramètres) ont montré leur intérêt pour la détection de défauts et le diagnostic. Rappelons pour chaque chapitre, les points et contributions principales et pour finir les perspectives qui pourraient guider des travaux futurs.

Le premier chapitre présente le contexte du travail et motive le sujet de recherche.

Dans le deuxième chapitre, nous avons présenté les problèmes abordés et outils fondamentaux jugés indispensables pour la bonne compréhension de la suite de cette thèse. Les différentes problématiques abordées dans la thèse sont l'atteignabilité hybride non linéaire, l'estimation de paramètres, l'observation d'état hybride. Quant aux outils mathématiques utilisés, il s'agit de l'arithmétique des intervalles et des zonotopes, ainsi que des méthodes d'intégration numérique garantie des équations différentielles ordinaires incertaines et les méthodes d'inversion ensembliste. Pour l'intégration numérique nous avons présenté le principe de base du développement en série de Taylor intervalle, puis, nous avons introduit les techniques de factorisation QR de Lohner qui nous ont permis de contrôler la propagation du pessimisme dû à l'effet d'enveloppe-

ment.

Dans le troisième chapitre, nous avons proposé un tour d'horizon de la littérature portant sur les principales techniques permettant de calculer l'ensemble atteignable hybride. Nous avons pu distinguer trois approches pour le passage des gardes. La première combine une approximation polynomiale de la condition de garde avec les algorithmes de recherche de zéro d'un polynôme. Cette approche est efficace mais non garantie. De plus, elle ne tient pas compte de la présence d'incertitudes dans les valeurs initiales des vecteurs d'état et des paramètres du modèle. La seconde approche utilise le potentiel offert par les zonotopes, les fonctions support ou les polytopes. Ces méthodes ont montré un intérêt réel pour l'analyse d'atteignabilité des systèmes hybrides linéaires. Elles sont de plus capables de passer à l'échelle. Cependant, elles ne restent applicables qu'aux systèmes linéaires. Dans le cas non linéaires, il est possible de procéder à une linéarisation garantie, mais au prix de sur-approximations. La troisième approche utilise les méthodes de propagation de contraintes et l'intégration numérique d'équations différentielles non linéaires. Elle est directement applicable aux systèmes non linéaires.

Dans le quatrième chapitre, nous avons présenté nos contributions originales permettant de faire du calcul d'ensemble atteignable hybride non linéaire. Dans la première partie, nous avons proposé des méthodes pour le passage ensembliste de gardes non linéaires et linéaires, c'est-à-dire la résolution du problème de la détection et localisation (temporelle) de l'intersection du tube de trajectoires continues avec des gardes ensemblistes. Nous avons considéré deux cas de figure ; pour le premier cas, la condition de garde est linéaire. Ces méthodes sont basées sur des opérations géométriques d'intersection entre zonotope et hyperplan/demi-espace. L'avantage de ces méthodes réside dans le fait qu'elles sont réduites à des calculs d'algèbre linéaire simples qui assurent une complexité polynomiale et des temps de calcul réduits qui favorisent le passage à l'échelle. Ensuite, pour le cas des gardes non linéaires, nous avons proposé une méthode rapide basée sur la résolution d'un problème de satisfaction de contraintes. Elle utilise la bisection dans la seule direction du temps et la collaboration de plusieurs contracteurs intervalle pour réduire le domaine des vecteurs d'état localisés sur la garde durant la tranche de temps étudiée. Dans la deuxième partie, nous avons proposé une méthode pour le saut ensembliste. Ce problème revient à calculer l'image d'un ensemble par une fonction qui peut être non linéaire. A ce titre, nous avons pu élaborer une expression analytique de la fonction de reset en fonction d'une représentation zonotopique (domaine de solution de notre intégrateur). Cela nous a permis de réduire la sur-approximation introduite lors du saut. Dans la troisième partie, nous avons proposé une méthode de fusion des trajectoires qui répond à une conséquence directe du problème de passage de gardes ensemblistes. Notre méthode proposée est basée sur l'utilisation des zonotopes car le domaine de solution de notre intégrateur représente la solution sous forme d'une MSBP qui est un zonotope particulier. Les boîtes sont en premier lieu représentées par leurs sommets et ce nuage de points est ensuite englobé dans le zonotope calculé. Cette méthode produit des encadrements réalisant une sur-approximation bien moindre que la méthode classique qui consiste à prendre l'enveloppe convexe des boîtes, comme nous avons pu le remarquer sur certains exemples. Ceci s'explique facilement par le fait que le zonotope est calculé pour se positionner suivant les directions correspondant aux composantes principales

du nuage de points. Après le saut, nous fusionnons toutes les trajectoires dans une MSPB qui est compatible avec notre approche d'atteignabilité continue. L'approche de fusion de trajectoires s'appuie sur un paramètre de réglage qui est choisis à la volée selon trois critères que nous avons définis. L'évaluation de la méthode sur les systèmes hybrides non linéaires a des performances prometteuses et en particulier des temps de calcul relativement faibles. Ces méthodes utilisées conjointement nous ont permis de calculer de façon rapide et efficace l'ensemble atteignable par un système dynamique hybride dont la dynamique continue est décrite par un système d'équations différentielles non linéaires et incertaines. De plus, nous avons fait une analyse de complexité de notre algorithme d'atteignabilité. Nous avons vu comment on pouvait contrôler la complexité du calcul d'atteignabilité hybride, c'est-à-dire réduire et contrôler le temps de calcul requis par les méthodes développées (passage ensembliste des gardes et fusion des trajectoires). Des évaluations numériques ont été réalisées sur des benchmarks, ce qui nous permis de comparer les performances de notre outil avec l'outil Flow* de la littérature. Les temps de calcul obtenus avec notre outil sont comparables, voire meilleurs que ceux obtenus avec l'outil Flow* sur certaines instances.

Dans le cinquième chapitre, nous avons montré que les méthodes de détection de défauts et de diagnostic pouvaient être construites à partir de deux types d'approches : l'approche estimateur d'état hybride ou l'approche estimation des paramètres. Nous avons présenté notre observateur hybride ensembliste basé sur l'approche prédiction/correction qui utilise notre méthode d'atteignabilité non linéaire hybride. La phase de prédiction consiste à calculer l'ensemble atteignable hybride. En ce qui concerne, la phase de correction nous avons utilisé une approche de la littérature permettant de majorer l'intersection de deux zonotopes. Ensuite, nous avons présenté l'estimation des paramètres. Nous avons alors associé notre algorithme d'atteignabilité non linéaire hybride avec SIVIA-H. L'algorithme SIVIA-H est une version de SIVIA adaptée au système dynamique hybride avec deux versions de tests d'inclusion, l'un utilisant l'arithmétique des intervalles et l'autre les zonotopes

Le travail réalisé laisse ouvert un certain nombre de questions et suggère des perspectives de travail. Dans cette optique, nous proposons comme suite de ce travail de thèse trois directions de recherche. La première direction porte sur l'amélioration des performances des méthodes d'atteignabilité hybride non linéaire, la deuxième sur la fusion des trajectoires, et la troisième sur le diagnostic et le pronostic des systèmes hybrides .

Pour la première direction de recherche :

1. Nous avons commencé à explorer une piste qui devrait apporter des améliorations supplémentaires, sans remettre en cause la solution proposée dans ce mémoire. L'idée prend en compte la géométrie liée au passage des gardes. En effet, on comprend intuitivement que le positionnement spatial du tube de trajectoires par rapport au sous-espace défini par une garde peut avoir une conséquence sur le temps requis pour le passage. Ainsi, notre idée est de procéder à une orientation de la boîte représentant le tube de trajectoires de

manière à positionner sa plus petite facette suivant la direction normale à la garde⁹. Cette méthode pourrait se mettre en œuvre en utilisant localement, au voisinage de la condition de garde, un critère de permutation similaire à celui proposé par Lohner pour l'intégration numérique.

2. Nous comptons interfacier notre méthode de passage de gardes avec des outils de la littérature à savoir VNODE-LP et si possible Flow*. En effet, notre outil d'atteignabilité actuel utilise un pas d'intégration de taille fixe, ce qui est amplifier le pessimisme.

Nous avons fait le choix de travailler avec notre propre outil d'atteignabilité, tout en sachant que ces outils (VNODE-LP et flow*) gèrent très bien des pas d'intégration dynamique, car nous avons mis l'accent sur le passage ensembliste de gardes.

Pour la seconde direction de recherche :

3. Nous souhaitons compléter la méthode de fusion des trajectoires en réalisant plusieurs fusions plutôt qu'une seule. En effet, si l'ensemble de boîtes se répartit en ensembles non connexes dans l'espace, il est plus pertinent de procéder à plusieurs fusions afin de minimiser la sur-approximation.

Nous pourrions nous baser sur une approche qui tirerait parti des machines à vecteur support (SVM) pour réaliser un clustering des boîtes en présence. Le critère des SVM étant basé sur la maximisation de la marge, c'est à dire sur la formation des clusters les plus distants possible, il nous semble que cela permettrait d'identifier correctement les sous-ensembles pour lesquels la fusion doit être réalisée. Une fusion serait en effet associée à chaque cluster.

Pour la troisième direction de recherche :

4. Pour l'estimation d'état hybride, il peut être intéressant de rajouter une phase backward (rétropropagation) dans notre observateur hybride forward (propagation) actuel construit au dessus de notre méthode d'atteignabilité hybride non linéaire. Ceci pourrait se faire en considérant le sens chronologique inverse, c'est-à-dire avec un pas d'intégration $-h$, aussi il faut inverser l'équation différentielle de chaque mode (ou en changeant le signe dans l'équation différentielle de chaque mode). On pourrait alors appliquer successivement la propagation et la rétro-propagation jusqu'à un critère d'arrêt des allers-retours à définir.
5. Nous nous baserons sur notre approche d'estimation ensembliste hybride, éventuellement améliorée par la phase de rétro-propagation, pour proposer un algorithme robuste de détection et de diagnostic de défauts. Par ailleurs, nous souhaitons étudier la faisabilité du pronostic de défauts ainsi que l'évaluation du Remaining Useful Life¹⁰ (RUL) des systèmes hybrides. En effet le pronostic détermine en général l'état futur des composants à partir de modèles de vieillissement/dégradation du système surveillé. Le cadre ensembliste des incertitudes bornées nous semble particulièrement pertinent pour le pronostic. En effet, le pronostic consistant à déterminer le RUL peut être réalisé par des méthode ensemblistes d'estimation de paramètres (comme ceux développé dans ce mémoire). Nous aborderons donc le pronostic par le biais d'une approche récursive incluant :

9. Dans le cas d'une garde linéaire

10. Durée de vie résiduelle en français

-
- une étape d'estimation paramétrique ensembliste (utilisant les méthodes développées dans ce mémoire sur un horizon glissant) pour déterminer le modèle d'évolution courant du système hybride supervisé, tout en tenant compte de toutes les sources d'incertitudes bornées,
 - une étape de prédiction ensembliste, et donc un calcul ensembliste d'atteignabilité hybride afin de déterminer le Remaining Useful Life (RUL).

BIBLIOGRAPHIE

- Alamo, T., Bravo, J., and Camacho, E. (2005). Guaranteed state estimation by zonotopes. *Automatica*, 41(6), 1035 – 1043.
- Althoff, M. (2010). *Reachability Analysis and its Application to the Safety Assessment of Autonomous Cars*. Ph.D. thesis.
- Althoff, M. and Krogh, B.H. (2011). Zonotope bundles for the efficient computation of reachable sets. In *CDC-ECE*, 6814–6821.
- Althoff, M. and Krogh, B.H. (2012). Avoiding geometric intersection operations in reachability analysis of hybrid systems. In *Proceedings of the 15th ACM international conference on Hybrid Systems : Computation and Control*, HSCC '12, 45–54. ACM, New York, NY, USA. doi :10.1145/2185632.2185643. URL <http://doi.acm.org/10.1145/2185632.2185643>.
- Althoff, M., Stursberg, O., and Buss, M. (2008). Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In *CDC*, 4042–4048.
- Althoff, M., Stursberg, O., and Buss, M. (2010). Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes. *Nonlinear Analysis : Hybrid Systems*, 4(2), 233 – 249. doi :<http://dx.doi.org/10.1016/j.nahs.2009.03.009>. URL <http://www.sciencedirect.com/science/article/pii/S1751570X09000442>. {IFAC} World Congress 2008.
- Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T., Ho, P.H., Nicollin, X., Olivero, A., Sifakis, J., and Yovine, S. (1995). The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138, 3–34.
- Alur, R., Courcoubetis, C., Henzinger, T., and P.-H.Ho (1993). Hybrid automata : an algorithmic approach to the specification and verification of hybrid systems. In *Workshop on theory of hybrid systems, vol. 736 in LNCS*, 209–229. Springer-Verlag.
- Alur, R., Dang, T., Esposito, J., Hur, Y., Ivancic, F., Kumar, V., Mishra, P., Pappas, G., and Sokolsky, O. (2003). Hierarchical modeling and analysis of embedded systems. *Proceedings of the IEEE*, 91(1), 11–28. doi :10.1109/JPROC.2002.805817.

- Alur, R., Courcoubetis, C., Henzinger, T.A., and Ho, P.H. (1992). Hybrid automata : An algorithmic approach to the specification and verification of hybrid systems. 209–229. Springer-Verlag.
- Alur, R. and Pappas, G.J. (eds.) (2004). *Hybrid Systems : Computation and Control, 7th International Workshop, HSCC 2004, Philadelphia, PA, USA, March 25-27, 2004, Proceedings*, volume 2993 of *Lecture Notes in Computer Science*. Springer.
- Ames, A., Sinnet, R., and Wendel, E. (2009). Three-dimensional kneed bipedal walking : A hybrid geometric approach. In R. Majumdar and P. Tabuada (eds.), *Hybrid Systems : Computation and Control*, volume 5469 of *Lecture Notes in Computer Science*, 16–30. Springer Berlin Heidelberg. doi :10.1007/978-3-642-00602-9_2. URL http://dx.doi.org/10.1007/978-3-642-00602-9_2.
- Anguelov, R. and Markov, S. (1998). Wrapping effect and wrapping function. *Reliable Computing*, 4(4), 311–330. doi :10.1023/A:1024414730820. URL <http://dx.doi.org/10.1023/A%3A1024414730820>.
- Antsaklis, P.J. and Nerode, A. (1998). Hybrid control systems : An introductory discussion to the special issue. *Automatic Control, IEEE Transactions on*, 43(4), 457–460.
- Asarin, E., Dang, T., and Maler, O. (2001). d/dt : a verification tool for hybrid systems. In *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, volume 3, 2893–2898 vol.3. doi :10.1109/.2001.980715.
- Asarin, E., Dang, T., Frehse, G., Girard, A., Le Guernic, C., and Maler, O. (2006). Recent progress in continuous and hybrid reachability analysis. In *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*, 1582–1587. doi :10.1109/CACSD-CCA-ISIC.2006.4776877.
- Aster, R., Borchers, B., and Thurber, C. (2013). *Parameter Estimation and Inverse Problems*. Academic Press. Academic Press. URL http://books.google.fr/books?id=ES4wWol_WucC.
- Aswani, A., Ding, J., Huang, H., Vitus, M., Gillula, J., Bouffard, P., and Tomlin, C. (2012). Verification and control of hybrid systems using reachability analysis with machine learning. In *Proceedings of the 15th ACM International Conference on Hybrid Systems : Computation and Control, HSCC '12*, 1–2. ACM, New York, NY, USA. URL <http://doi.acm.org/10.1145/2185632.2185634>.
- Balluchi, A., Benvenuti, L., Benedetto, M.D.D., and Sangiovanni-vincentelli, A.L. (2002). Non-linear and Hybrid Systems in Automotive Control.
- Bar-Shalom, Y. (1987). *Tracking and Data Association*. Academic Press Professional, Inc., San Diego, CA, USA.
- Barber, C.B., Dobkin, D.P., and Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. *ACM Trans. on Mathematical Software*, 22(4), 469–483.

- Barequet, G. and Har-Peled, S. (2001). Efficiently approximating the minimum-volume bounding box of a point set in three dimensions. *J. Algorithms*, 38, 91–109.
- Bayouhd, M., Trave-Massuyes, L., and Olive, X. (2009). On-line analytic redundancy relations instantiation guided by component discrete-dynamics for a class of non-linear hybrid systems. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, 6970–6975.
- Bazart, L., Maquin, D., Bèle, B., Khelassi, A., and Ragot, J. (2013a). Détection de changement de mode de fonctionnement. In *5e Journées Doctorales / Journées Nationales MACS, JD-JN-MACS 2013*, CDROM. Strasbourg, France. URL <https://hal.archives-ouvertes.fr/hal-00922054>.
- Bazart, L., Maquin, D., Khelassi, A., Bèle, B., and Ragot, J. (2013b). Operating mode recognition : Application in continuous casting. In *2nd International Conference on Control and Fault-Tolerant Systems, SysTol'13*, CDROM. Nice, France. URL <https://hal.archives-ouvertes.fr/hal-00846341>.
- Belard, N., Pencolé, Y., and Combacau, M. (2011). A theory of meta-diagnosis : Reasoning about diagnostic systems. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, 731–737. URL <http://ijcai.org/papers11/Papers/IJCAI11-129.pdf>.
- Benazera, E. and Travé-Massuyès, L. (2009). Set-theoretic estimation of hybrid system configurations. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 39(5), 1277–1291.
- Bendtsen, C. and Stauning, O. (1996). FADBAD, a flexible C++ package for automatic differentiation. Technical Report IMM-REP-1996-17, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark.
- Benhamou, F., Goualard, F., Granvilliers, L., and Puget, J.F. (1999). Revising hull and box consistency. In *INT. CONF. ON LOGIC PROGRAMMING*, 230–244. MIT press.
- Berz, M. (2002). COSY INFINITY Version 8.1 — User’s Guide and Reference Manual. Department of Physics and Astronomy MSUHEP-20704, Department of Physics and Astronomy, Michigan State University. URL <http://cosy.pa.msu.edu/cosymanu/index.html>.
- Berz, M., Hoffstätter, G., and Atter, G.H. (1998). Computation and application of taylor polynomials with interval remainder bounds. *Reliable Computing*, 4, 83–97.
- Berz, M. and Makino, K. (1998a). Verified integration of odes and flows using differential algebraic methods on high-order taylor models. *Reliable Computing*, 4(4), 361–369. doi : 10.1023/A:1024467732637. URL <http://dx.doi.org/10.1023/A%3A1024467732637>.
- Berz, M. and Makino, K. (1998b). Verified integration of odes and flows using differential algebraic methods on high-order taylor models. *Reliable Computing*, 4(4), 361–369.

- Berz, M. and Makino, K. (2006). Performance of Taylor model methods for validated integration of ODEs. In J. Dongarra, K. Madsen, and J. Waśniewski (eds.), *Applied Parallel Computing. State of the Art in Scientific Computing*, volume 3732 of *Lecture Notes in Computer Science*, 65–73. Springer Berlin Heidelberg. doi :10.1007/11558958_8. URL http://dx.doi.org/10.1007/11558958_8.
- Bezzaoucha, S., Marx, B., Maquin, D., and Ragot, J. (2013a). Nonlinear joint state and parameter estimation : application to a wastewater treatment plant. *Control Engineering Practice*, 21(10), 1377–1385. doi :10.1016/j.conengprac.2013.06.009. URL <https://hal.archives-ouvertes.fr/hal-00831876>.
- Bezzaoucha, S., Marx, B., Maquin, D., and Ragot, J. (2013b). State and Parameter Estimation for Nonlinear Systems : a Takagi-Sugeno Approach. In *American Control Conference, ACC 2013*, CDROM. Washington, United States. URL <https://hal.archives-ouvertes.fr/hal-00782929>.
- Bhatia, A. and Frazzoli, E. (2004). Incremental search methods for reachability analysis of continuous and hybrid systems. In R. Alur and G. Pappas (eds.), *Hybrid Systems : Computation and Control*, volume 2993 of *Lecture Notes in Computer Science*, 142–156. Springer Berlin Heidelberg.
- Birta, L.G., Oren, T.I., and Kettenis, D.L. (1985). A robust procedure for discontinuity handling in continuous system simulation. *Trans. Soc. Comput. Simul. Int.*, 2(3), 189–205. URL <http://dl.acm.org/citation.cfm?id=5263.5264>.
- Blesa, J., Puig, V., and Saludes, J. (2011). Identification for passive robust fault detection using zonotope-based set-membership approaches. *International Journal of Adaptive Control and Signal Processing*, 25(9), 788–812.
- Blesa, J., Puig, V., and Saludes, J. (2012a). Robust fault detection using polytope-based set-membership consistency test. *Control Theory & Applications, IET*, 6(12), 1767–1777.
- Blesa, J., Puig, V., and Saludes, J. (2012b). Robust identification and fault diagnosis based on uncertain multiple input–multiple output linear parameter varying parity equations and zonotopes. *Journal of Process Control*, 22(10), 1890–1912.
- Bouissou, O., Chapoutot, A., Djaballah, A., and Kieffer, M. (2014). Computation of Parametric Barrier Functions for Dynamical Systems using Interval Analysis. In *IEEE CDC 2014*, 1–4. Los Angeles, United States. URL <https://hal-supelec.archives-ouvertes.fr/hal-01073673>.
- Bouissou, O., Mimram, S., and Chapoutot, A. (2012). Hyson : Set-based simulation of hybrid systems. In *Rapid System Prototyping (RSP), 2012 23rd IEEE International Symposium on*, 79–85. IEEE.

- Braems, I., Ramdani, N., Boudenne, A., Kieffer, M., Jaulin, L., Ibos, L., Walter, E., and Candau, Y. (2005). New set-membership techniques for parameter estimation in presence of model uncertainty. In *Proceedings of the 5th International Conference on Inverse Problems in Engineering : Theory and Practice, Cambridge, Leeds University Press, ISBN 085316242, Volume I*, pp. B09.
- Bravo, J.M., Alamo, T., and Camacho, E.F. (2006). Robust MPC of constrained discrete-time nonlinear systems based on approximated reachable sets. *Automatica*, 42(10), 1745–1751. doi : 10.1016/j.automatica.2006.05.003. URL <http://dx.doi.org/10.1016/j.automatica.2006.05.003>.
- Brockett, R.W. (1982). *Control theory and singular Riemannian geometry*. Springer.
- Brockett, R. and Dai, L. (1993). Non-holonomic kinematics and the role of elliptic functions in constructive controllability. In *Nonholonomic motion planning*, 1–21. Springer.
- Büeler, B., Enge, A., and Fukuda, K. (2000). Exact volume computation for polytopes : A practical study. In G. Kalai and G. Ziegler (eds.), *Polytopes – Combinatorics and Computation*, volume 29 of *DMV Seminar*, 131–154. Birkhäuser Basel. doi :10.1007/978-3-0348-8438-9_6. URL http://dx.doi.org/10.1007/978-3-0348-8438-9_6.
- Burkard, R.E., Hamacher, H.W., and Rote, G. (1991). Sandwich approximation of univariate convex functions with an application to separable convex programming. *Naval Research Logistics (NRL)*, 38(6), 911–924. doi :10.1002/nav.3800380609. URL <http://dx.doi.org/10.1002/nav.3800380609>.
- Cellier, F.E. (1977). Combined continuous/discrete system simulation languages : usefulness, experiences and future development. *SIGSIM Simul. Dig.*, 9(1), 18–21. doi :10.1145/1102505.1102514. URL <http://doi.acm.org/10.1145/1102505.1102514>.
- Chabert, G. (2007). IBEX : Interval Based EXplorer, available at , <http://www.ibex-lib.org/>.
- Chabert, G. (2013). IBEX : Interval Based EXplorer, available at <http://www.ibex-lib.org/>.
- Chabert, G. and Jaulin, L. (2009). Contractor programming. *Artificial Intelligence*, 173(11), 1079 – 1100.
- Chen, X. (2013). User’s manual of flow v1. 2.0.
- Chen, X., Abraham, E., and Sankaranarayanan, S. (2012a). Taylor model flowpipe construction for non-linear hybrid systems. In *33rd IEEE Real-Time Systems Symposium (RTSS 12)*, 183–192.
- Chen, X., Abraham, E., and Sankaranarayanan, S. (2012b). Taylor model flowpipe construction for non-linear hybrid systems. In *RTSS*, 183–192.
- Chen, X., Abraham, E., and Sankaranarayanan, S. (2013). Flow* : An analyzer for non-linear hybrid systems. In *Computer Aided Verification*, 258–263. Springer.

- Chernousko, F. (2005). Ellipsoidal state estimation for dynamical systems. *Nonlinear Analysis : Theory, Methods Applications*, 63(5–7), 872 – 879.
- Chutinan, A. and Krogh, B. (2003). Computational techniques for hybrid system verification. *Automatic Control, IEEE Transactions on*, 48(1), 64–75. doi :10.1109/TAC.2002.806655.
- Clarke, E.M., Fehnker, A., Han, Z., Krogh, B.H., Stursberg, O., and Theobald, M. (2003). Verification of hybrid systems based on counterexample-guided abstraction refinement. In H. Garavel and J. Hatcliff (eds.), *TACAS*, volume 2619 of *Lecture Notes in Computer Science*, 192–207. Springer.
- Cleary, J.C. (1987). Logical arithmetic. *Future Computing Systems*, 2, 125–149.
- Collavizza, H., Delobel, F., and Rueher, M. (1999). Comparing partial consistencies. *Reliable computing*, 5, 213–228.
- Combastel, C., Zhang, Q., and Lalami, A. (2008). Fault diagnosis based on the enclosure of parameters estimated with an adaptive observer. In *17th IFAC World Congress*. Seoul, Korea.
- Davis, E. (1987). Constraint propagation with interval labels. *Artificial Intelligence*, 32, 281–331.
- de Alfaro, L., Henzinger, T.A., and Kupferman, O. (2007). Concurrent reachability games. *Theor. Comput. Sci.*, 386(3), 188–217. doi :10.1016/j.tcs.2007.07.008. URL <http://dx.doi.org/10.1016/j.tcs.2007.07.008>.
- Donzé, A., Krogh, B., and Rajhans, A. (2009). Parameter synthesis for hybrid systems with an application to simulink models. In R. Majumdar and P. Tabuada (eds.), *Hybrid Systems : Computation and Control*, volume 5469 of *Lecture Notes in Computer Science*, 165–179. Springer Berlin Heidelberg. doi :10.1007/978-3-642-00602-9_12. URL http://dx.doi.org/10.1007/978-3-642-00602-9_12.
- Dounias, G. and Linkens, D. (2004). Adaptive systems and hybrid computational intelligence in medicine. *Artificial Intelligence in Medicine*, 32(3), 151–155. URL <http://www.ncbi.nlm.nih.gov/pubmed/15531147>.
- Eggers, A., Ramdani, N., Nediakko, N.S., and Fränzle, M. (2012). Improving the SAT modulo ODE approach to hybrid systems analysis by combining different enclosure methods. *Software & Systems Modeling*, 1–28. doi :10.1007/s10270-012-0295-3. URL <http://dx.doi.org/10.1007/s10270-012-0295-3>.
- Eggers, A., Ramdani, N., Nediakov, N.S., and Fränzle, M. (2014). Iolavabe—the isat-ode layer around vnode-lp and bracketing enclosures—a nonlinear reachability library.
- Eijgenraam, P. (1981). *The Solution of Initial Value Problems Using Interval Arithmetic*. Mathematical Centre Tracts No. 144. Stichting Mathematisch Centrum, Amsterdam.
- Esposito, J.M. and Kumar, V. (2007). A state event detection algorithm for numerically simulating hybrid systems with model singularities. *ACM Trans. Model. Comput. Simul.*, 17(1).

- Esposito, J.M., Kumar, V., and Pappas, G.J. (2001). Accurate event detection for simulating hybrid systems. In *HSCC*, 204–217.
- Fisher, K. and Maybeck, P. (2000). Multiple model adaptive estimation with filter spawning. In *American Control Conference, 2000. Proceedings of the 2000*, volume 4, 2326–2331 vol.4. doi :10.1109/ACC.2000.878595.
- Fisher, M.E. (1991). A semiclosed-loop algorithm for the control of blood glucose levels in diabetics. *IEEE Transactions on Biomedical Engineering*, 38(1), 57–61. URL <http://www.ncbi.nlm.nih.gov/pubmed/2026432>.
- Frehse, G. (2005). Phaver : Algorithmic verification of hybrid systems past hytech. In M. Morari and L. Thiele (eds.), *Hybrid Systems : Computation and Control*, volume 3414 of *Lecture Notes in Computer Science*, 258–273. Springer Berlin Heidelberg. doi :10.1007/978-3-540-31954-2_17. URL http://dx.doi.org/10.1007/978-3-540-31954-2_17.
- Frehse, G., Guernic, C.L., Donzé, A., Cotton, S., Ray, R., Lebeltel, O., Ripado, R., Girard, A., Dang, T., and Maler, O. (2011). Spaceex : Scalable verification of hybrid systems. In *CAV*, 379–395.
- Frehse, G. and Ray, R. (2012). Flowpipe-guard intersection for reachability computations with support functions. In *IFAC Conf. Analysis and Design of Hybrid Systems (ADHS)*, 94–101.
- Funiak, S. and Williams, B.C. (2003). Multi-modal particle filtering for hybrid systems with autonomous . . . In *WORKSHOP ON PRINCIPLES OF DIAGNOSIS, 2003*.
- Furler, S.M., Kraegen, E.W., Smallwood, R.H., and Chisholm, D.J. (1985). Blood glucose control by intermittent loop closure in the basal mode : computer simulation studies with a diabetic model. *Diabetes Care*, 8(6), 553–561. URL http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve&db=PubMed&dopt=Citation&list_uids=4075940.
- Gao, Y., Lygeros, J., Quincampoix, M., and Seube, N. (2004). On the control uncertain impulsive systems : Approximate stabilization and controlled invariance. *International Journal of Control*, 77(16), 1393–1407. URL <http://control.ee.ethz.ch/index.cgi?page=publications;action=details;id=2566>.
- Gao, Y., Lygeros, J., and Quincampoix, M. (2006). The reachability problem for uncertain hybrid systems revisited : A viability theory perspective. In J. Hespanha and A. Tiwari (eds.), *Hybrid Systems : Computation and Control*, volume 3927 of *Lecture Notes in Computer Science*, 242–256. Springer Berlin Heidelberg.
- Gilbert, E., Johnson, D., and Keerthi, S. (1988). A fast procedure for computing the distance between complex objects in three-dimensional space. *Robotics and Automation, IEEE Journal of*, 4(2), 193–203. doi :10.1109/56.2083.
- Girard, A. (2002). Detection of event occurrence in piecewise linear hybrid systems.

- Girard, A. (2004). *Algorithmic Analysis of Hybrid Systems*. Theses, Institut National Polytechnique de Grenoble - INPG. URL <https://tel.archives-ouvertes.fr/tel-00007064>.
- Girard, A. (2005). Reachability of uncertain linear systems using zonotopes. In M. Morari and L. Thiele (eds.), *Hybrid Systems : Computation and Control, HSCC 2005, March, 2005*, volume 3414 of *Lecture Notes in Computer Science*, 291–305. Springer, Zurich, Suisse.
- Girard, A., Guernic, C., and Maler, O. (2006). Efficient computation of reachable sets of linear time-invariant systems with inputs. In J. Hespanha and A. Tiwari (eds.), *Hybrid Systems : Computation and Control*, volume 3927 of *Lecture Notes in Computer Science*, 257–271. Springer Berlin Heidelberg. doi :10.1007/11730637_21. URL http://dx.doi.org/10.1007/11730637_21.
- Girard, A. and Guernic, C.L. (2008). Zonotope/hyperplane intersection for hybrid systems reachability analysis. In *HSCC*, 215–228.
- Girard, A. and Le Guernic, C. (2008). Efficient reachability analysis for linear systems using support functions. In *Proceedings of IFAC World Congress*, 8966–8971. IFAC, Séoul, Corée, République De. doi :10.3182/20080706-5-KR-1001.0569. URL <http://hal.archives-ouvertes.fr/hal-00307009>.
- Golub, G.H. and Loan, C.F.V. (1996). *Matrix Computations*. Johns Hopkins University Press, Baltimore, 3rd edition.
- Gonzalez, R., Fiacchini, M., Alamo, T., Guzman, J., and Rodriguez, F. (2011). Online robust tube-based MPC for time-varying systems : a practical approach. *International Journal of Control*, 84(6), 1157–1170. doi :10.1080/00207179.2011.594093. URL <http://www.tandfonline.com/doi/abs/10.1080/00207179.2011.594093>.
- Granvilliers, L., Cruz, J., and Barahona, P. (2004). Parameter estimation using interval computations. *SIAM Journal on Scientific Computing*, 26(2), 591–612.
- Gucik-Derigny, D. (2011). *Contribution au pronostic des systèmes à base de modèles : théorie et application*. Ph.D. thesis, PhD thesis, L'Université paul cézanne aixmarseille III.
- Guéguen, H., Lefebvre, M.A., Zaytoon, J., and O.Nasri (2009). Safety verification and reachability analysis for hybrid systems. *Annual Reviews in Control*, 33(1), 25 – 36.
- Guéguen, H. and Zaytoon, J. (2004). On the formal verification of hybrid systems. *Control Engineering Practice*, 12(10), 1253 – 1267. URL <http://www.sciencedirect.com/science/article/pii/S0967066104000632>. <ce :title>Analysis and Design of Hybrid Systems</ce :title>.
- Guennebaud, G., Jacob, B., et al. (2010). Eigen v3. <http://eigen.tuxfamily.org>.
- Guernic, C. and Girard, A. (2009a). Reachability analysis of hybrid systems using support functions. In *Proceedings of the 21st International Conference on Computer Aided Verification, CAV '09*, 540–554. Springer-Verlag, Berlin, Heidelberg.

- Guernic, C.L. and Girard, A. (2009b). Reachability analysis of hybrid systems using support functions. In *CAV*, 540–554.
- Guerra, P. and Puig, V. (2008). Passive robust fault detection using interval map parity equations : inverse vs direct image tests. In *Proceedings of the 17th World Congress IFAC. Seoul, Korea*.
- Guéguen, H. and Zaytoon, J. (2004). On the formal verification of hybrid systems. *Control Engineering Practice*, 12(10), 1253 – 1267. doi :10.1016/j.conengprac.2004.03.015. URL <http://www.sciencedirect.com/science/article/pii/S0967066104000632>. <ce :title>Analysis and Design of Hybrid Systems</ce :title>.
- Hagemann, W. (2014). Reachability analysis of hybrid systems using symbolic orthogonal projections. In *Computer Aided Verification*, 407–423. Springer.
- Hansen, E. and Walster, G. (2004). *Global optimization using interval analysis*. Marcel Dekker, 2nd edition.
- Hansen, K.A., Ibsen-Jensen, R., and Miltersen, P.B. (2010). The complexity of solving reachability games using value and strategy iteration. *CoRR*, abs/1007.1812.
- Hentenryck, P.V., Mcallester, D., and Kapur, D. (1997). Solving polynomial systems using a branch and prune approach. *SIAM Journal on Numerical Analysis*, 34, 797–827.
- Henzinger, T., Horowitz, B., Majumdar, R., and Wong-Toi, H. (2000a). Beyond HYTECH : Hybrid systems analysis using interval numerical methods. In *HSCC, vol. 1790 in LNCS*, 130–144.
- Henzinger, T.A. and Ho, P.H. (1995). Algorithmic analysis of nonlinear hybrid systems. In *Proceedings of the 7th International Conference on Computer Aided Verification*, 225–238. Springer-Verlag, London, UK, UK. URL <http://dl.acm.org/citation.cfm?id=647764.735689>.
- Henzinger, T.A., Ho, P.H., and Wong-toi, H. (1997). Hytech : A model checker for hybrid systems. *Software Tools for Technology Transfer*, 1, 460–463.
- Henzinger, T., Horowitz, B., Majumdar, R., and Wong-Toi, H. (2000b). Beyond hytech : Hybrid systems analysis using interval numerical methods. In N. Lynch and B. Krogh (eds.), *Hybrid Systems : Computation and Control*, volume 1790 of *Lecture Notes in Computer Science*, 130–144. Springer Berlin Heidelberg. doi :10.1007/3-540-46430-1_14. URL http://dx.doi.org/10.1007/3-540-46430-1_14.
- Hiskens, I. (2001). Nonlinear dynamic model evaluation from disturbance measurements. *Power Systems, IEEE Transactions on*, 16(4), 702–710. doi :10.1109/59.962416.
- Hoefkens, J., Berz, M., and Makino, K. (2003). Computing validated solutions of implicit differential equations. *Advances in Computational Mathematics*, 19(1-3), 231–253. doi : 10.1023/A:1022858921155. URL <http://dx.doi.org/10.1023/A%3A1022858921155>.

- Hofbauer, M. and Williams, B. (2004). Hybrid estimation of complex systems. *Systems, Man, and Cybernetics, Part B : Cybernetics, IEEE Transactions on*, 34(5), 2178–2191. doi :10.1109/TSMCB.2004.835009.
- Hoffman, M., Schaffer, S., and Wedeward, K. (2009). Parameter estimation in delayed-switching hybrid dynamical systems. In *Power Energy Society General Meeting, 2009. PES '09. IEEE*, 1–6. doi :10.1109/PES.2009.5275661.
- Ichalal, D., Marx, B., Maquin, D., and Ragot, J. (2010). Estimation d'état des systèmes non linéaires incertains sous forme multimodèle de type Takagi-Sugeno. In *1er Colloque International Francophone, Ingénierie et Environnement, CIFIE'2010*, CDROM. Annaba, Algeria. URL <https://hal.archives-ouvertes.fr/hal-00548114>.
- Isermann, R. (1984). Process fault detection based on modeling and estimation methods—a survey. *Automatica*, 20(4), 387–404.
- Isermann, R. (1993). Fault diagnosis of machines via parameter estimation and knowledge processing : Tutorial paper. *Automatica*, 29(4), 815–835.
- Isermann, R. (1997). Supervision, fault-detection and fault-diagnosis methods—an introduction. *Control engineering practice*, 5(5), 639–652.
- Ishii, D., Ueda, K., and Hosobe, H. (2009). Simulation of hybrid systems based on hierarchical interval constraints. In *SimuTools*, 37.
- Ishii, D., Ueda, K., and Hosobe, H. (2011). An interval-based SAT modulo ODE solver for model checking nonlinear hybrid systems. *International Journal on Software Tools for Technology Transfer (STTT)*, 1–13. doi :10.1007/s10009-011-0193-y.
- J. J. Gertler (1997). Fault detection and isolation using parity relations. *Control Engineering Practice*, 5, 653–661. doi :{10.1016/S0967-0661(97)00047-6}.
- Jaulin, L., Kieffer, M., Didrit, O., and Walter, E. (2001). *Applied Interval Analysis : with examples in parameter and state estimation, robust control and robotics*. Springer-Verlag, London.
- Jaulin, L. and Walter, E. (1993). Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29, 1053–1064.
- Johansson, Rolf, R.A. (ed.) (2003). *Nonlinear and Hybrid Systems in Automotive Control*. Springer.
- Johnson, T. and Tucker, W. (2008). Rigorous parameter reconstruction for differential equations with noisy data. *Automatica*, (9), 2422–2426. URL <http://dblp.uni-trier.de/db/journals/automatica/automatica44.html#JohnsonT08>.
- Kajdan, R., Aubry, D., and Kratz, F. (2007). Gauss-newton observers for mode and continuous state estimation of nonlinear switched systems. In *Decision and Control, 2007 46th IEEE Conference on*, 3214–3220.

- Kay, S. (1993). *Fundamentals of Statistical Signal Processing : Estimation theory*. Fundamentals of Statistical Signal Processing. Prentice-Hall PTR. URL <https://books.google.co.il/books?id=aFwESQAACAAJ>.
- Kerrigan, E.C., Kerrigan, E.C., Kerrigan, E.C., Maciejowski, J.M., Maciejowski, J.M., and Maciejowski, J.M. (2002). Feedback min-max model predictive control using a single linear program : Robust stability and the explicit solution.
- Kieffer, M., Jaulin, L., and Walter, E. (1998). Guaranteed recursive nonlinear state estimation using interval analysis. In *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, volume 4, 3966–3971. IEEE.
- Kieffer, M. and Walter, E. (2011). Guaranteed estimation of the parameters of nonlinear continuous-time models : contributions of interval analysis. *International Journal of Adaptive Control and Signal Processing*, 25(3), 191–207. doi :10.1002/acs.1194. URL <https://hal.inria.fr/inria-00548147>.
- Kieffer, M., Walter, E., and Simeonov, I. (2006). Guaranteed nonlinear parameter estimation for continuous-time dynamical models. In *Proceedings 14th IFAC Symposium on System Identification, Newcastle, Aus*, volume 18, 843–848.
- Klima, G. (2008-2010). AML++ : Another Matrix Library in c++, available at <http://amlpp.sourceforge.net/>.
- Knüppel, O. (1994). PROFIL/BIAS a fast interval library. *Computing*, 53(3-4), 277–287.
- Kühn, W. (1998). Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61(1), 47–67. doi :10.1007/BF02684450. URL <http://dx.doi.org/10.1007/BF02684450>.
- Kurzanski, A. and Varaiya, P. (2000). Ellipsoidal techniques for reachability analysis. In N. Lynch and B. Krogh (eds.), *Hybrid Systems : Computation and Control*, volume 1790 of *Lecture Notes in Computer Science*, 202–214. Springer Berlin Heidelberg. doi :10.1007/3-540-46430-1_19. URL http://dx.doi.org/10.1007/3-540-46430-1_19.
- Kvasnica, M., Grieder, P., and Baotić, M. (2004). Multi-Parametric Toolbox (MPT). URL <http://control.ee.ethz.ch/~mpt/>.
- Kühn, W. (1998). Rigorously computed orbits of dynamical systems without the wrapping effect. *Computing*, 61(1), 47–67. doi :10.1007/BF02684450. URL <http://dx.doi.org/10.1007/BF02684450>.
- Lafortune, S. (2015). Diagnosis of discrete event systems. In *Encyclopedia of Systems and Control*. doi :10.1007/978-1-4471-5102-9_56-1. URL http://dx.doi.org/10.1007/978-1-4471-5102-9_56-1.

- Lalami, A. (2008). *Diagnostic et approches ensemblistes à base de zonotopes*. Ph.D. thesis. URL http://books.google.fr/books?id=Iu_YXwAACAAJ.
- Lalami, A. and Combastel, C. (2007). A state bounding algorithm for linear systems with bounded input and bounded slew-rate. *European Control Conference*.
- Le, V.T.H. (2012). *Robust predictive control by zonotopic set-membership estimation*. Theses, Supélec. URL <https://tel.archives-ouvertes.fr/tel-00765444>.
- Le, V.T.H., Stoica, C., Alamo, T., Camacho, E.F., and Dumur, D. (2013). Zonotopic guaranteed state estimation for uncertain systems. *Automatica*, 49(11), 3418 – 3424. doi :<http://dx.doi.org/10.1016/j.automatica.2013.08.014>. URL <http://www.sciencedirect.com/science/article/pii/S0005109813004159>.
- le Guernic, C. (2005). *Calcul Efficace de l'ensemble Atteignable des Systèmes Linéaires avec Incertitudes*. Master's thesis, Master Parisien de Recherche en Informatique, ENS.
- Lefebvre, M.A. and Guéguen, H. (2006). Hybrid abstractions of affine systems. *Nonlinear Analysis : Theory, Methods & Applications*, 65(6), 1150 – 1167. Hybrid Systems and Applications Hybrid Systems and Applications.
- Limon, D., Bravo, J., Alamo, T., and Camacho, E. (2005). Robust MPC of constrained nonlinear systems based on interval arithmetic. *Control Theory and Applications, IEE Proceedings -*, 152(3), 325–332. doi :10.1049/ip-cta:20040480.
- Lin, Y. and Stadtherr, M.A. (2007). Validated solutions of initial value problems for parametric odes. *Applied Numerical Mathematics*, 57(10), 1145–1162.
- Lohner, R.J. (1987a). Enclosing the solutions of ordinary initial and boundary value problems. In E.W. Kaucher, U.W. Kulisch, and C. Ullrich (eds.), *Computer Arithmetic : Scientific Computation and Programming Languages*, 255–286. Wiley-Teubner, Stuttgart.
- Lohner, R.J. (1988). *Einschließung der Lösung gewöhnlicher Anfangs- und Randwertaufgaben und Anwendungen*. Ph.D. thesis, Universität Karlsruhe.
- Lohner, R.J. (1987b). Enclosing the solutions of ordinary initial and boundary value problems. In E.W. Kaucher, U.W. Kulisch, and C. Ullrich (eds.), *Computer Arithmetic : Scientific Computation and Programming Languages*, 255–286. Wiley-Teubner Series in Computer Science, Stuttgart.
- Lygeros, J. (2004). On reachability and minimum cost optimal control. *Automatica*, 40(6), 917–927. URL <http://control.ee.ethz.ch/index.cgi?page=publications;action=details;id=2542>.
- Lygeros, J., Godbole, D.N., and Sastry, S. (1998a). Verified Hybrid Controllers for Automated Vehicles. *IEEE Transactions on Automatic Control*, 43(4), 522–539. URL <http://control.ee.ethz.ch/index.cgi?page=publications;action=details;id=2549>.

- Lygeros, J., Tomlin, C., and Sastry, S. (1998b). On controller synthesis for nonlinear hybrid systems. In *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, volume 2, 2101–2106 vol.2. doi :10.1109/CDC.1998.758644.
- Lygeros, J., Tomlin, C.J., and Sastry, S. (1999). Controllers for reachability specifications for hybrid systems. *Automatica*, 349–370. URL <http://control.ee.ethz.ch/index.cgi?page=publications;action=details;id=2547>.
- Maïga, M., Combastel, C., Ramdani, N., and Travé-Massuyès, L. (2014). Nonlinear hybrid reachability using set integration and zonotope enclosures. In *The Proceedings of the 13th European Control Conference*, 234–239. Strasbourg, France.
- Maïga, M., Ramdani, N., and Travé-Massuyès, L. (2013). A fast method for solving guard set intersection in nonlinear hybrid reachability. In *Proc. of 52nd IEEE Conference on Decision and Control, CDC 2013.*, 508–513.
- Marx, B., Maquin, D., and Ragot, J. (2010). State estimation and diagnosis of uncertain systems based on an interval approach. In *Conference on Control and Fault-Tolerant Systems, SysTol'10*, CDROM. Nice, France. doi :10.1109/SYSTOL.2010.5676036. URL <https://hal.archives-ouvertes.fr/hal-00525124>.
- Maybeck, P. and Hanlon, P. (1995). Performance enhancement of a multiple model adaptive estimator. *Aerospace and Electronic Systems, IEEE Transactions on*, 31(4), 1240–1254. doi : 10.1109/7.464348.
- Mazor, E., Averbuch, A., Bar-Shalom, Y., and Dayan, J. (1998). Interacting multiple model methods in target tracking : a survey. *Aerospace and Electronic Systems, IEEE Transactions on*, 34(1), 103–123. doi :10.1109/7.640267.
- Meslem, N. and Ramdani, N. (2011). Interval observer design based on nonlinear hybridization and practical stability analysis. *International Journal of Adaptive Control and Signal Processing*, 25(3), 228–248.
- Meslem, N., Ramdani, N., and Candau, Y. (2010a). Using hybrid automata for set-membership state estimation with uncertain nonlinear continuous-time systems. *Journal of Process Control*, 20, 481–489.
- Meslem, N., Ramdani, N., and Candau, Y. (2010b). Using hybrid automata for set-membership state estimation with uncertain nonlinear continuous-time systems. *Journal of Process Control*, 20(4), 481 – 489. doi :<http://dx.doi.org/10.1016/j.jprocont.2010.02.001>. URL <http://www.sciencedirect.com/science/article/pii/S0959152410000429>.
- Mezyani, T.E. (2005). *Méthodologie de surveillance des systèmes dynamiques hybrides*. Ph.D. thesis. URL <http://www.theses.fr/2005LIL10054>. Thèse de doctorat dirigée par Staroswiecki, Marcel et Cocquempot, Vincent Informatique industrielle et automatique Lille 1 2005.

- Milanese, M., Norton, J., Piet-Lahanier, H., and Walter, É. (1996). *Bounding approaches to system identification*. Springer Science & Business Media.
- Mitchell, I. and Tomlin, C. (2013). Diagnosing multiple faults in dynamic hybrid systems. In A. Abraham and S.M. Thampi (eds.), *Intelligent Informatics*, volume 182 of *Advances in Intelligent Systems and Computing*, 129–139.
- Mokhtari, A. (2007). *Diagnostic des systèmes hybrides : développement d'une méthode associant la détection par classification et la simulation dynamique*. Theses, INSA de Toulouse. URL <https://tel.archives-ouvertes.fr/tel-00200034>.
- Moore, R.E. (1996). *Interval Analysis*. Englewood Cliffs, prentice-hall.
- Müller, M. (1927). Über das Fundamentaltheorem in der Theorie der gewöhnlichen Differentialgleichungen. *Mathematische Zeitschrift*, 26, 619–645.
- Nedialkov, N. (2006). VNODE-LP – A Validated Solver for Initial Value Problems in Ordinary Differential Equations. Technical Report CAS-06-06-NN, Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada. URL <http://www.cas.mcmaster.ca/~nedialk/vnodelp/doc/vnode.pdf>.
- Nedialkov, N.S. (1999a). Computing rigorous bounds on the solution of an initial value problem for an ordinary differential equation. Technical report.
- Nedialkov, N.S. (1999b). *Computing rigorous bounds on the solution of an initial value problem for an ordinary differential equation*. Ph.D. thesis, Toronto, Ont., Canada, Canada. AAINQ41256.
- Nedialkov, N., Jackson, K., and Corliss, G. (1999). validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105, 21–68.
- Neumaier, A. (2004). Complete search in continuous global optimization and constraint satisfaction. In A. Iserles (ed.), *Acta Numerica*, chapter 4. Cambridge University Press.
- O'Rourke, J. (1985). Finding minimal enclosing boxes. *International Journal of Computer Information Sciences*, 14(3), 183–199. doi :10.1007/BF00991005. URL <http://dx.doi.org/10.1007/BF00991005>.
- Park, T. and Barton, P.I. (1996). State event location in differential-algebraic models. *ACM Trans. Model. Comput. Simul.*, 6(2), 137–165.
- Pencolé, Y. (2003). All from one, one for all, failure diagnosis of discrete event systems using representatives. In *15th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2003)*, 3-5 November 2003, Sacramento, California, USA, 208–212. doi :10.1109/TAI.2003.1250192. URL <http://dx.doi.org/10.1109/TAI.2003.1250192>.
- Pepyne, D.L. and Cassandra, C.G. (2000). Optimal control of hybrid systems in manufacturing. *Proceedings of The IEEE*, 88, 1108–1123. doi :10.1109/5.871312.

- Piazza, C., Antoniotti, M., Mysore, V., Policriti, A., Winkler, F., and Mishra, B. (2005). Algorithmic algebraic model checking i : challenges from systems biology. In *Proceedings of the 17th international conference on Computer Aided Verification, CAV'05*, 5–19. Springer-Verlag, Berlin, Heidelberg. doi :10.1007/11513988_3. URL http://dx.doi.org/10.1007/11513988_3.
- Platzer, A. and Clarke, E. (2007). The image computation problem in hybrid systems model checking. In A. Bemporad, A. Bicchi, and G. Buttazzo (eds.), *Hybrid Systems : Computation and Control*, volume 4416 of *Lecture Notes in Computer Science*, 473–486. Springer Berlin Heidelberg. doi :10.1007/978-3-540-71493-4_37. URL http://dx.doi.org/10.1007/978-3-540-71493-4_37.
- Raimondo, D.M., Rivero, S., Jones, C., and Morari, M. (2011). A robust explicit nonlinear MPC controller with input-to-state stability guarantees. In *IFAC World Congress*. Milano. URL <http://control.ee.ethz.ch/index.cgi?page=publications;action=details;id=3758>.
- Raïssi, T., Ramdani, N., and Candau, Y. (2004a). Set membership state and parameter estimation for systems described by nonlinear differential equations. *Automatica*, 40(10), 1771–1777.
- Raïssi, T., Ramdani, N., and Candau, Y. (2006). Robust nonlinear continuous-time state estimation using interval taylor models. In *Proceedings 6th IFAC Symposium on Robust Control*. Toulouse.
- Raïssi, T., Ramdani, N., and Candau, Y. (2009). Set membership parameter estimation in the frequency domain. *International Journal of Control, Automation, and Systems*, 7(5), 824–834.
- Raïssi, T. (2004). *Méthodes ensemblistes pour l'estimation d'état et de paramètres*. Theses, Université Paris XII Val de Marne. URL <https://tel.archives-ouvertes.fr/tel-00292380>.
- Raïssi, T., Ramdani, N., and Candau, Y. (2004b). Set membership state and parameter estimation for systems described by nonlinear differential equations. *Automatica*, 40(10), 1771–1777.
- Rakovic, S., Kouvaritakis, B., Cannon, M., Panos, C., and Findeisen, R. (2012). Parameterized tube model predictive control. *Automatic Control, IEEE Transactions on*, 57(11), 2746–2761. doi :10.1109/TAC.2012.2191174.
- Ramdani, N. and Nedialkov, N.S. (2011). Computing reachable sets for uncertain nonlinear hybrid systems using interval constraint-propagation techniques. *Nonlinear Analysis : Hybrid Systems*, 5(2), 149 – 162.
- Ramdani, N. (2012). Analyse et estimation robustes des systèmes hybrides non linéaires, une approche ensembliste. In *CIFA 2012*, 991–1002. France. URL <https://hal.archives-ouvertes.fr/hal-00719902>.
- Ramdani, N., Meslem, N., and Candau, Y. (2008). Reachability of uncertain nonlinear systems using a nonlinear hybridization. In *HSCC*, 415–428.

- Rauh, A., Kletting, M., Aschemann, H., and Hofer, E. (2006). Interval methods for simulation of dynamical systems with state-dependent switching characteristics. In *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*, 355–360. doi : 10.1109/CACSD-CCA-ISIC.2006.4776672.
- Raïssi, T. (2012a). Techniques robustes pour l’observation - Application au diagnostic et à la stabilisation.
- Raïssi, T. (2012b). Techniques robustes pour l’observation - Application au diagnostic et à la stabilisation. Post-Doctoral Degree (Phd students supervisor) Rapporteur 1 : José Ragot Rapporteur 2 : Louise Travé-Massuyès Rapporteur 3 : Eric Walter Membre du jury : Leonid Fridman Membre du jury : Alain Oustaloup Membre du jury : Daniel Roviras Membre du jury : Ali Zolghadri.
- Rihm, R. (1994a). Interval methods for initial value problems in ODEs. In E.S. in Computational Mathematics (ed.), *Validated computations : proceedings of the IMACS-GAMM International Workshop on Validated Computations*, 173–208. J. Herzberger, Amsterdam.
- Rihm, R. (1998). Implicit methods for enclosing solutions of odes. *Journal of Universal Computer Science*, 42(6), 202–209.
- Rihm, R. (1992). Enclosing solutions with switching points in ordinary differential equations. In L. Atanassova and J. Herzberger (eds.), *Computer Arithmetic and Enclosure Methods*, 419–425. North-Holland, Amsterdam.
- Rihm, R. (1993). Einschließung von Lösungen mit Schaltpunkten bei gewöhnlichen Anfangswertproblemen. *Z. angew. Math. Mech.*, 73, T815–T817.
- Rihm, R. (1994b). Interval methods for initial value problems in ODEs. In J. Herzberger (ed.), *Topics in Validated Computations : Proceedings of the IMACS-GAMM International Workshop on Validated Computations, University of Oldenburg*, Elsevier Studies in Computational Mathematics, 173–207. Elsevier, Amsterdam, New York.
- Riverso, S., Battocchio, A., and Ferrari-Trecate, G. (2013). Pnmpc toolbox.
- Röbenack, K. (2008). Computation of multiple lie derivatives by algorithmic differentiation. *Journal of Computational and Applied Mathematics*, 213(2), 454 – 464. doi :<http://dx.doi.org/10.1016/j.cam.2007.01.036>. URL <http://www.sciencedirect.com/science/article/pii/S0377042707000441>.
- Sanderson, C. (2013). Armadillo : C++ linear algebra library, available at <http://arma.sourceforge.net/>.
- Schneider, R. (1994). Polytopes and brunn-minkowski theory. In T. Bisztriczky, P. McMullen, R. Schneider, and Weiss (eds.), *Polytopes : Abstract, Convex and Computational*, volume 440

- of *NATO ASI Series*, 273–299. Springer Netherlands. doi :10.1007/978-94-011-0924-6_13. URL http://dx.doi.org/10.1007/978-94-011-0924-6_13.
- Schweppe, F. (1968). Recursive state estimation : Unknown but bounded errors and system inputs. *Automatic Control, IEEE Transactions on*, 13(1), 22–28. doi :10.1109/TAC.1968.1098790.
- Scott, J. and Barton, P. (2010a). Convex enclosures for the reachable sets of nonlinear parametric ordinary differential equations. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, 5695–5700. doi :10.1109/CDC.2010.5717641.
- Scott, J.K. and Barton, P.I. (2010b). Tight, efficient bounds on the solutions of chemical kinetics models. *Computers & Chemical Engineering*, 34(5), 717 – 731. doi :<http://dx.doi.org/10.1016/j.compchemeng.2009.11.021>.
- Scott, J.K. and Barton, P.I. (2013). Bounds on the reachable sets of nonlinear control systems. *Automatica*, 49(1), 93 – 100. doi :<http://dx.doi.org/10.1016/j.automatica.2012.09.020>. URL <http://www.sciencedirect.com/science/article/pii/S0005109812004839>.
- Seydou, R., Raïssi, T., Zolghadri, A., and Efimov, D. (2012). Robust Fault Diagnosis with Interval Continuous-time Parity Equations. In *The 8th IFAC International Symposium on Fault Detection, Supervision and Safety of Technical Processes*. Mexico City, Mexico.
- Shampine, L.F., Gladwell, I., and Brankin, R.W. (1987). Reliable solution of special event location problems for ODEs. *ACM transactions on Mathematical Software*, 17, 11–25.
- Shumsky, A. and Zhirabok, A.N. (2012). Redundancy relations for fault diagnosis in hybrid systems. In *Fault Detection, Supervision and Safety of Technical Processes*, volume 8, 1226–1231.
- Stursberg, O. and Krogh, B. (2003a). Efficient representation and computation of reachable sets for hybrid systems. In *HSCC, vol. 2623 in LNCS*, 482–497. Springer-Verlag.
- Stursberg, O., Kowalewski, S., Hoffmann, I., and Preußig, J. (1996). Comparing timed and hybrid automata as approximations of continuous systems. In *Hybrid Systems'96*, 361–377.
- Stursberg, O. and Krogh, B.H. (2003b). Efficient representation and computation of reachable sets for hybrid systems. In *Proceedings of the 6th international conference on Hybrid systems : computation and control, HSCC'03*, 482–497. Springer-Verlag, Berlin, Heidelberg. URL <http://dl.acm.org/citation.cfm?id=1768100.1768137>.
- Subias, A. (2006). *Contribution au diagnostic des systèmes complexes*. Ph.D. thesis, Université Paul Sabatier-Toulouse III.
- Sunaga, T. (2009). Theory of an interval algebra and its application to numerical analysis. *Japan Journal of Industrial and Applied Mathematics*, 26(2-3), 125–143. doi :10.1007/BF03186528. URL <http://dx.doi.org/10.1007/BF03186528>.

- Tabatabaeipour, S.M. and Stoustrup, J. (2013). Set-membership state estimation for discrete time piecewise affine systems using zonotopes. In *Control Conference (ECC), 2013 European*, 3143–3148. IEEE.
- Tami, R., Boutat, D., and Zheng, G. (2013). Nonlinear observer normal form with output injection and extended dynamic. In *9th IFAC Symposium on Nonlinear Control Systems*. IFAC, Toulouse, Ethiopia. URL <https://hal.inria.fr/hal-00910212>.
- Tomlin, C.J., Lygeros, J., and Sastry, S. (2000). A Game Theoretic Approach to Controller Design for Hybrid Systems. *Proceedings of IEEE*, 88, 949–969. URL <http://control.ee.ethz.ch/index.cgi?page=publications;action=details;id=2545>.
- Tomlin, C.J., Mitchell, I.M., Bayen, A.M., and Oishi, M. (2003). Computational techniques for the verification of hybrid systems. *Proceedings of the IEEE*, 91(7), 986–1001.
- Tomlin, C. (2011). Verification and control of hybrid systems using reachability analysis. In *Control Automation (MED), 2011 19th Mediterranean Conference on*, 150–150. doi :10.1109/MED.2011.5983088.
- Tucker, W., Kutalik, Z., and Moulton, V. (2007). Estimating parameters for generalized mass action models using constraint propagation. *Mathematical Biosciences*, 208(2), 607 – 620.
- Tuy, H. (1995). *Handbook of global optimization, D.C. Optimization : Theory, Methods and Algorithms (pp. 149–216)*. Dordrecht : Kluwer Academic Publishers.
- Van Der Schaft, A.J., Schumacher, J.M., van der Schaft, A.J., and van der Schaft, A.J. (2000). An introduction to hybrid dynamical systems.
- Vento, J., Blesa, J., Puig, V., and Sarrate, R. (2015). Set-membership parity space hybrid system diagnosis. *Int. J. Systems Science*, 46(5), 790–807. doi :10.1080/00207721.2014.977978. URL <http://dx.doi.org/10.1080/00207721.2014.977978>.
- Vento, J., Puig, V., Sarrate, R., et al. (2012). Fault detection and isolation of hybrid systems using diagnosers that reason on components. In *Fault Detection, Supervision and Safety of Technical Processes*, volume 8, 1250–1255.
- Vicino, A. and Zappa, G. (1996). Sequential approximation of feasible parameter sets for identification with set membership uncertainty. *Automatic Control, IEEE Transactions on*, 41(6), 774–785. doi :10.1109/9.506230.
- Vivien, F. and Wicker, N. (2004). Minimal enclosing parallelepiped in 3d. *Comput. Geom. Theory Appl.*, 29(3), 177–190. doi :10.1016/j.comgeo.2004.01.009. URL <http://dx.doi.org/10.1016/j.comgeo.2004.01.009>.
- Wahrburg, A. and Adamy, J. (2012). Robust fault isolation observers for non-square systems-a parametric approach. In *Fault Detection, Supervision and Safety of Technical Processes*, volume 8, 1275–1280.

- Walter, E. and Kieffer, M. (2007). Guaranteed nonlinear parameter estimation in knowledge-based models. *Journal of Computational and Applied Mathematics*, 199, 277–285.
- Walter, E. and Piet-Lahanier, H. (1990). Estimation of parameter bounds from bounded-error data : a survey. *Mathematics and Computers in Simulation*, 32(5), 449 – 468. doi :[http://dx.doi.org/10.1016/0378-4754\(90\)90002-Z](http://dx.doi.org/10.1016/0378-4754(90)90002-Z). URL <http://www.sciencedirect.com/science/article/pii/037847549090002Z>.
- Walter, É. and Pronzato, L. (1994). *Identification de modèles paramétriques à partir de données expérimentales*. Masson.
- Waltz, D.L. (1975). *Generating semantic descriptions from drawings of scenes with shadows*, 19–91. McGraw-Hill, New York.
- Wang, W., Li, L., Zhou, D., and Liu, K. (2007a). Robust state estimation and fault diagnosis for uncertain hybrid nonlinear systems. *Nonlinear analysis : Hybrid systems*, 1(1), 2–15.
- Wang, W., Li, L., Zhou, D., and Liu, K. (2007b). Robust state estimation and fault diagnosis for uncertain hybrid nonlinear systems. *Nonlinear Analysis : Hybrid Systems*, 1(1), 2 – 15. doi :<http://dx.doi.org/10.1016/j.nahs.2007.01.002>. URL <http://www.sciencedirect.com/science/article/pii/S1751570X07000611>.
- Wang, W., Zhou, D., and Li, Z. (2006). Robust state estimation and fault diagnosis for uncertain hybrid systems. *Nonlinear Analysis : Theory, Methods & Applications*, 65(12), 2193 – 2215. doi :<http://dx.doi.org/10.1016/j.na.2006.02.047>. URL <http://www.sciencedirect.com/science/article/pii/S0362546X06003889>. *Hybrid Systems and Applications* (8).
- Willsky, A.S. (1976). A survey of design methods for failure detection in dynamic systems. *Automatica*, 12(6), 601 – 611. doi :[http://dx.doi.org/10.1016/0005-1098\(76\)90041-8](http://dx.doi.org/10.1016/0005-1098(76)90041-8). URL <http://www.sciencedirect.com/science/article/pii/0005109876900418>.
- Yazarael, H. and Pappas, G. (2004). Geometric programming relaxations for linear systems reachability. In *Proceedings of 2004 American Control Conference (ACC)*, 553–559.
- Zaytoon, J. and Lafortune, S. (2013). Overview of fault diagnosis methods for discrete event systems. *Annual Reviews in Control*, 37(2), 308–320. doi :[10.1016/j.arcontrol.2013.09.009](https://doi.org/10.1016/j.arcontrol.2013.09.009). URL <http://dx.doi.org/10.1016/j.arcontrol.2013.09.009>.
- Zhang, Y. and Jiang, J. (2001). Integrated active fault-tolerant control using imm approach. *Aerospace and Electronic Systems, IEEE Transactions on*, 37(4), 1221–1235. doi :[10.1109/7.976961](https://doi.org/10.1109/7.976961).
- Zhao, F., Koutsoukos, X., Haussecker, H., Reich, J., and Cheung, P. (2005). Monitoring and fault diagnosis of hybrid systems. *Systems, Man, and Cybernetics, Part B : Cybernetics, IEEE Transactions on*, 35(6), 1225–1240.
- Ziegler, G.M. (1995). *Lectures on polytopes*. Springer-Verlag, New York.

Moussa MAIGA

Surveillance préventive des systèmes hybrides à incertitudes bornées

Résumé :

Cette thèse est dédiée au développement d'algorithmes génériques pour l'observation ensembliste de l'état continu et du mode discret des systèmes dynamiques hybrides dans le but de réaliser la détection de défauts. Cette thèse est organisée en deux grandes parties. Dans la première partie, nous avons proposé une méthode rapide et efficace pour le passage ensembliste des gardes. Elle consiste à procéder à la bisection dans la seule direction du temps et ensuite faire collaborer plusieurs contracteurs simultanément pour réduire le domaine des vecteurs d'état localisés sur la garde, durant la tranche de temps étudiée. Ensuite, nous avons proposé une méthode pour la fusion des trajectoires basée sur l'utilisation des zonotopes. Ces méthodes, utilisées conjointement, nous ont permis de caractériser de manière garantie l'ensemble des trajectoires d'état hybride engendrées par un système dynamique hybride incertain sur un horizon de temps fini. La deuxième partie de la thèse aborde les méthodes ensemblistes pour l'estimation de paramètres et pour l'estimation d'état hybride (mode et état continu) dans un contexte à erreurs bornées. Nous avons commencé en premier lieu par décrire les méthodes de détection de défauts dans les systèmes hybrides en utilisant une approche paramétrique et une approche observateur hybride. Ensuite, nous avons décrit deux méthodes permettant d'effectuer les tâches de détection de défauts. Nous avons proposé une méthode basée sur notre méthode d'atteignabilité hybride non linéaire et un algorithme de partitionnement que nous avons nommé SIVIA-H pour calculer de manière garantie l'ensemble des paramètres compatibles avec le modèle hybride, les mesures et avec les bornes d'erreurs. Ensuite, pour l'estimation d'état hybride, nous avons proposé une méthode basée sur un prédicteur-correcteur construit au dessus de notre méthode d'atteignabilité hybride non linéaire.

Mots clés : Systèmes dynamiques hybrides, diagnostic, zonotope, arithmétique d'intervalles, erreur bornée, identification, observation, atteignabilité hybride non linéaire, équations différentielles ordinaires .

Preventive monitoring of hybrid systems in a bounded-error framework

Summary :

This thesis is dedicated to the development of generic algorithms for the set-membership observation of the continuous state and the discrete mode of hybrid dynamical systems in order to achieve fault detection. This thesis is organized into two parts. In the first part, we have proposed a fast and effective method for the set-membership guard crossing. It consists in carrying out bisection in the time direction only and then makes several contractors working simultaneously to reduce the domain of state vectors located on the guard during the study time slot. Then, we proposed a method for merging trajectories based on zonotopic enclosures. These methods, used together, allowed us to characterize in a guaranteed way the set of all hybrid state trajectories generated by an uncertain hybrid dynamical system on a finite time horizon. The second part focuses on set-membership methods for the parameters or the hybrid state (mode and continuous state) of a hybrid dynamical system in a bounded error framework. We started first by describing fault detection methods for hybrid systems using the parametric approach and the hybrid observer approach. Then, we have described two methods for performing fault detection tasks. We have proposed a method for computing in a guaranteed way all the parameters consistent with the hybrid dynamical model, the actual data and the prior error bound, by using our nonlinear hybrid reachability method and an algorithm for partition which we denote SIVIA-H. Then, for hybrid state estimation, we have proposed a method based on a predictor-corrector, which is also built on top of our non-linear method for hybrid reachability.

Keywords : Hybrid dynamical systems, zonotope, interval arithmetic, bounded error, identification, observation, non linear hybrid reachability, ordinary differential equations, diagnosis.



PRISME EA 4229 Université d'Orléans
8 rue Léonard de Vinci, 45072 Orléans,
LAAS-CNRS
7 Av. du Colonel Roche, 31400 Toulouse

