Atteignabilité hybride des systèmes dynamiques continus par analyse par intervalles : application à l'estimation ensembliste

Nacim Meslem

▶ **To cite this version:**

Nacim Meslem. Atteignabilité hybride des systèmes dynamiques continus par analyse par intervalles : application à l'estimation ensembliste. Autre. Université Paris-Est, 2008. Français. NNT : 2008PEST0022 . tel-00461673

HAL Id: tel-00461673

https://theses.hal.science/tel-00461673

Submitted on 5 Mar 2010

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.

Université Paris Est

École Doctorale : SIMME

Laboratoire : CERTES

Thèse

présentée pour l'obtention du titre de

Docteur de l'Université Paris Est

Spécialité : Sciences de l'Ingénieur

par **Nacim MESLEM**

Atteignabilité hybride des systèmes dynamiques continus par analyse par intervalles. Application à l'estimation ensembliste

Soutenue publiquement le 23 Juin 2008 devant la commission d'examen composée de

Rapporteurs :	Hervé GUÉGUEN	Professeur Supélec Rennes
	Jean-Luc GOUZÉ	DR INRIA Sophia Antipolis - Méditerranée
Examinateur :	Ramine NIKOUKHAH	DR INRIA Rocquencourt
Directeurs de Thèse :	Yves CANDAU	Professeur Université Paris Est
	Nacim RAMDANI	MC Université Paris Est

Centre d'Etude et de Recherche en Thermique, Environnement et Systèmes

Université Paris Est, 61 avenue du général de Gaulle 94000 Créteil

Mis en page avec la classe Phdlasl

Table des matières

Introduction générale **7**

I Calcul par intervalles **13**

I.1 Introduction 13

I.2 Arithmétique d'intervalles 14

I.2.1 Intervalles 14

I.2.2 Vecteurs et matrices intervalles 15

I.2.2.1 Norme 16

I.3 Intervalles et opérations sur les ensembles 17

I.4 Fonction d'inclusion 18

I.4.1 Fonction d'inclusion centrée : Théorème de la valeur moyenne . . . 20

I.4.2 Fonctions d'inclusion de Taylor 20

I.4.2.1 Exemple comparatif 21

I.5 Contracteur 21

I.5.1 Contracteur de Newton 22

I.6 Inversion ensembliste 23

I.6.1 Inversion ensembliste avec contraction 25

I.7 Différentiation automatique 25

I.7.1 Principe de la différentiation automatique 26

I.7.1.1 Le modèle simple 27

I.7.1.2 Le modèle étendu 28

I.7.1.3 Différentiation automatique en mode direct 29

I.8 Conclusion 31

II Intégration numérique garantie d'équations différentielles ordinaires 33

II.1 Introduction . . . 33

II.2 Présentation du problème . . . 34

II.3 Méthodes de Taylor intervalles . . . 36

II.3.1 Vérification de l'existence et de l'unicité de la solution : Etape de prédiction . . . 36

II.3.2 Calcul de l'encadrement garanti de solutions : Etape de correction . 39

II.3.2.1 Principe . . . 39

II.3.2.2 Effet d'enveloppement . . . 40

II.3.2.3 Méthode directe . . . 42

II.3.2.4 La méthode de Lohner . . . 43

II.3.2.4.1 Méthode parallélépipédique . . . 45

II.3.2.4.2 Factorisation **QR** . . . 46

II.3.2.5 La méthode de Rihm . . . 48

II.3.2.6 Cas des EDO où la factorisation **QR** n'est pas préconisé . 52

II.3.2.7 Schéma d'intégration implicite . . . 52

II.3.2.8 Une comparaison simple entre la méthode de la valeur moyenne et la méthode implicite dans le cas des EDO linéaires . . . 55

II.4 Modèle de Hermite-Obreschkoff . . . 57

II.5 Limite des méthodes d'intégration numérique garantie . . . 58

II.6 Conclusion . . . 60

III Théorie de comparaison des inégalités différentielles 61

III.1 Introduction . . . 61

III.2 Définitions et propriétés . . . 62

III.2.1 Systèmes monotones . . . 62

III.2.2 Préservation d'ordre . . . 62

III.2.3 Condition graphique pour la préservation d'ordre . . . 63

III.2.4 Systèmes coopératifs . . . 66

III.3 Intégration numérique garantie des ODE monotones . . . 68

III.3.1 Obtention de solutions minimale et maximale . . . 69

III.3.2 Exemples illustratifs 70
III.4 Systèmes non monotones 76
III.4.1 Théorème de Müller 77
III.4.2 Intégration numérique garantie dans le cas général 78
III.4.2.1 Approximations de solutions minimale et maximale 78
III.4.2.2 Exemples illustratifs 80
III.5 Quelques remarques concernant la précision des encadrements de solutions 86
III.6 Conclusion 87

IV Approche hybride pour l'encadrement des solutions d'EDO incertaines 89
IV.1 Introduction 89
IV.2 L'automate hybride d'encadrement 90
IV.3 Exemple illustratif 92
IV.4 Déroulement de l'automate hybride 95
IV.5 L'approche hybride interfacée avec un algorithme de bissection 99
IV.6 Cas des EDO coopératives 104
IV.6.1 Exemple illustratif 105
IV.6.2 Algorithme d'encadrement dans le cas coopératif 108
IV.7 Conclusion 112

V Identification ensembliste des systèmes dynamiques à temps continu 113
V.1 Introduction 113
V.2 Présentation de l'approche ensembliste pour l'identification 114
V.3 Identification ensembliste des systèmes dynamiques à temps continu 116
V.4 Application 120
V.4.1 Modèle 121
V.4.2 Signal d'excitation du type échelon 122
V.4.2.1 Image directe 122
V.4.2.1.1 Simulation par la méthode HOI 123
V.4.2.1.2 Simulation par la méthode HOIS 125
V.4.2.1.3 Simulation par la méthode TCID 128
V.4.2.2 Inversion ensembliste 133

V.4.3 Signal d'excitation du type harmonique . . . 135
V.4.3.1 Image directe . . . 135
V.4.3.1.1 Simulation par les méthodes HOI et HOIS . . . 135
V.4.3.1.2 Simulation par la méthode TCID . . . 138
V.4.3.2 Inversion ensembliste . . . 141
V.5 Conclusion . . . 143

VI Observateurs intervalles pour les systèmes dynamiques continus incertains 145

VI.1 Introduction . . . 145
VI.2 Première approche : Prédiction-correction . . . 147
VI.2.1 Contexte . . . 147
VI.2.2 Principe . . . 148
VI.2.3 Application . . . 150
VI.2.3.1 Modèle . . . 150
VI.2.3.2 Simulation (tests et résultats) . . . 151
VI.2.3.2.1 Estimation d'état par le biais des méthodes MI . 152
Essai 1 : . . . 152
Essai 2 : . . . 152
Essai 3 : . . . 153
Essai 4 : . . . 153
VI.2.3.2.2 Estimation d'état par le biais des méthodes TCID 155
Essai 1 : . . . 157
Essai 2 : . . . 158
Essai 3 : . . . 160
Essai 4 : . . . 160
Essai 5 : . . . 162
VI.3 Deuxième approche : Observateurs intervalles . . . 164
VI.3.1 Contexte . . . 164
VI.3.2 Observateur intervalle . . . 167
VI.3.3 Positivité de l'erreur d'observation . . . 167
VI.4 Applications . . . 168

VI.4.1 Exemple 1 168
VI.4.1.1 Tests et résultats 169
Essai 1 170
Essai 2 170
Essai 3 171
Essai 4 171
VI.4.2 Exemple 2 173
VI.4.2.1 Tests et résultats 175
Essai 1 175
Essai 2 175
Essai 3 178
Essai 4 178
VI.5 Conclusion 181

VII Atteignabilité des systèmes dynamiques non linéaires incertains 183
VII.1 Introduction 183
VII.2 Définition des systèmes dynamiques hybrides 184
VII.3 Analyse des systèmes dynamiques hybrides 185
VII.3.1 Preuve numérique de propriété 186
VII.4 Approximation conservative de l'espace d'état atteignable 187
VII.5 Applications 188
VII.5.1 Exemple 1 : Système linéaire 188
VII.5.2 Exemple 2 : Système non linéaire coopératif 189
VII.5.2.1 Simulation par les modèles de Taylor intervalles 192
VII.5.2.2 Simulation par la méthode hybride d'encadrement 193
VII.5.3 Exemple 3 : Système non linéaire non coopératif 1 196
VII.5.3.1 Simulation par les méthodes intervalles d'intégration . . . 197
VII.5.3.2 Simulation par les méthodes d'encadrement dynamique . . 197
VII.5.4 Exemple 4 : Système non linéaire non coopératif 2 200
VII.5.4.1 Premier cas : Sans bissection 203
VII.5.4.2 Deuxième cas : Avec bissection 203
VII.6 Conclusion 206

Conclusion générale **207**

Annexes **211**

A Méthode de Hermite-Obreschkoff intervalle **213**

A.1 Polynôme de Hermite-Obreschkoff : Cas ponctuel 213

A.2 Extension aux intervalles 214

A.2.1 Phase de prédiction 215

A.2.2 Phase de correction 216

Bibliographie **233**

Introduction générale

Dans notre vie quotidienne, le traitement de données joue un rôle très important dans la compréhension et la maîtrise de notre environnement. En pratique ces données sont soit directement accessibles à la mesure ou estimées par des algorithmes basés sur des mesures indirectes. Généralement, les données mesurables sont affectées par des incertitudes de nature déterministe ou aléatoire. Par ailleurs, avant d'exécuter les algorithmes d'estimation et de traitement de ces données, on doit procéder à une amélioration de la précision des mesures expérimentales disponibles. Pour cela, on cherche à caractériser explicitement ces incertitudes. Généralement on fait appel à des méthodes statistiques mettant en jeu des lois de probabilités pour établir un modèle pour l'évolution de ces incertitudes. Ceci s'avère difficile voire impossible dans certains cas pratiques. En revanche en s'appuyant sur les bornes d'incertitudes fournies par les constructeurs des appareils de mesure ainsi que les supports physiques des bruits et perturbations agissant sur la chaîne d'acquisition des données, il est naturel et judicieux de représenter ces données expérimentales par des intervalles bornés, de bornes connues *a priori*. Ces intervalles contiennent de manière certaine les vraies valeurs de mesures.

Une autre question importante concerne le problème de la précision des calculs effectués sur les calculateurs numériques induite par la présentation décimale des nombres réels. En effet, dans plusieurs cas la répétition des opérations d'arrondi et de troncature conduit à des résultats avec une très mauvaise précision. Là aussi, la représentation intervalle des nombres réels permet de contrôler la propagation des erreurs dues à l'arrondi et à la troncature.

En pratique, l'évolution continue des systèmes dynamiques physiques ou industriels est modélisée par un ensemble d'équations différentielles. Dans ce contexte, cette thèse porte sur la résolution d'équations différentielles ordinaires en présence d'incertitude affectant

simultanément les conditions initiales et les paramètres. Nous nous intéressons au cas des incertitudes décrites par des intervalles et à l'estimation ensembliste d'état et de paramètres de systèmes dynamiques modélisés par des représentations d'état continues.

Cette thèse est organisée en deux parties. La première partie est consacrée aux outils mathématiques et aux méthodes d'intégration numérique garantie des équations différentielles incertaines. Ces méthodes permettent de caractériser de manière garantie l'ensemble des trajectoires d'état engendrées par un système dynamique incertain dont les incertitudes sont naturellement représentées par des intervalles bornés. Dans la seconde partie, nous montrons comment résoudre les problèmes de l'identification et de l'observation dans un contexte à erreurs-inconnues-mais bornées et de bornes connues *a priori*. Puis nous aborderons le calcul de sur-approximations de l'espace d'état atteignable par des systèmes dynamiques continus incertains.

L'avantage principal des méthodes et des algorithmes de calculs que nous présentons dans cette thèse est la preuve numérique de résultats. En effet, les ensembles solutions obtenus par ces méthodes sont globaux et garantis, c'est-à-dire qu'aucune solution possible du problème traité ne sera perdue et l'absence de solution est détectable. De plus, ces méthodes et algorithmes nécessitent beaucoup moins d'informations sur le système étudié et son environnement par rapport aux méthodes et algorithmes statistiques.

Dans le premier chapitre, nous rappelons les notions de base de l'analyse par intervalles. Nous présentons ensuite le principe de l'inversion ensembliste fondée sur l'analyse par intervalles. Nous rappelons aussi, dans ce même chapitre, la technique de différentiation automatique utilisée dans les méthodes d'intégration numériques garanties introduites dans les chapitres suivants.

Nous présentons, dans le deuxième chapitre, les méthodes d'intégration numérique garantie des équations différentielles ordinaires fondées sur le calcul par intervalles. Ces méthodes s'appuient sur le développement en série de Taylor étendu aux intervalles. Puis, nous rappelons les différentes techniques mises en place pour améliorer et contrôler la qualité des résultats lors du processus d'intégration. Parmi ces techniques, nous présentons en premier lieu, l'utilisation du théorème de la valeur moyenne pour évaluer les coefficients du développement en série de Taylor et le pré-conditionnement matriciel qui consiste à réaliser des changements dynamiques de coordonnées pour les équations différentielles.

Par la suite, nous rappelons le schéma d'intégration numérique garantie implicite et celui basé sur le polynôme de Hermite-Obreschkoff (ce dernier est détaillé dans l'annexe A). Parallèlement, nous donnons les algorithmes qui permettent l'implémentation informatique de ces différents schémas et techniques d'intégration numérique garantie, en particulier l'algorithme de Lohner, l'algorithme de la valeur moyenne étendue de Rihm et l'algorithme de Nedialkov basé sur le polynôme de Hermite-Obreschkoff ...

Le troisième chapitre est consacré aux méthodes d'intégration garantie basées sur les théorèmes de comparaison des inégalités différentielles. Nous commençons ce chapitre par la définition des systèmes dynamiques monotones et systèmes dynamiques coopératifs. Ensuite, nous expliquons le principe de ce type d'approche d'intégration numérique garantie qui consiste à sélectionner parmi l'infinité des cas possibles deux équations différentielles ne mettant en jeu aucune incertitude qui fourniront une solution minimale et une solution maximale pour l'équation différentielle incertaine étudiée. Nous montrons aussi que dans le cas de systèmes monotones les équations différentielles fournissant les solutions minimale et maximale sont naturellement découplées ; elles peuvent donc être évaluées indépendemment. En revanche dans le cas des systèmes non-monotones ces deux équations différentielles sont mutuellement couplées. Dans les deux cas, nous proposons des règles pour la construction des systèmes englobant basée sur l'analyse de la monotonie du second membre de l'équation différentielle par rapport à toutes ses variables ou paramètres incertains. Dans certains cas, ces règles de construction permettent d'obtenir un encadrement optimal des solutions.

Le quatrième chapitre représente la contribution majeure de la thèse à l'intégration numérique garantie des équations différentielles incertaines. Lorsqu'on utilise les théorèmes de comparaison des inégalités différentielles pour l'intégration numérique, on est souvent confronté à des situations où les conditions exigées par les règles de construction présentées dans le troisième chapitre ne sont pas satisfaites sur toute la période de simulation. En revanche, ces conditions sont continûment satisfaites par morceaux. Dès lors, nous proposons d'utiliser une modélisation des systèmes englobants sous la forme d'automates hybrides continu-discret. Il s'agit alors de transformer l'équation différentielle incertaine en un ensemble d'équations différentielles certaines gouvernées par un automate d'état fini et dont l'exécution génère les trajectoires solutions minimales et maximales. Nous

présentons dans le détail cette approche d'encadrement hybride en montrant comment construire les modes de fonctionnement continu de l'automate hybride, les conditions de garde définissant les domaines invariants et comment franchir les frontières des domaines sans perdre la garantie de l'encadrement. Nous présentons également des algorithmes qui résument cette méthode.

Nous consacrons le cinquième chapitre, à l'identification ensembliste des paramètres des systèmes dynamiques à temps continu dans un contexte à erreurs-inconnues-mais bornées et de bornes connues *a priori*. Nous rappelons brièvement le principe de cette approche d'identification ensembliste lorsque les mesures sont disponibles à des instants discrets et les systèmes décrits par des représentations d'état continu. Ensuite, nous montrons comment l'utilisation conjointe des méthodes d'intégration numérique garantie des équations différentielles avec l'inversion ensembliste par le bais du calcul intervalles, permet de résoudre avec succès ce genre de problèmes. Nous présentons comme application, le dispositif d'identification des propriétés thermo-physiques, conductivité et diffusivité, des matériaux composites, développé au Laboratoire. Nous exploitons deux types de signaux d'excitation. Dans le premier cas, nous utilisons un échelon de température comme signal d'excitation et nous montrons l'avantage de travailler avec les méthodes d'intégration numérique garantie des équations différentielles basées sur les théorèmes de comparaison par rapport à celles basées sur le développement en série de Taylor intervalles ou le polynôme de Hermite-Obreschkoff. Par ailleurs, dans le second cas nous considérons un signal d'excitation multi-harmonique et nous montrons la nécessité d'utiliser notre approche hybride d'intégration numérique garantie pour résoudre ce genre de problèmes.

Dans le sixième chapitre, nous étudions les observateurs intervalles. Nous rappelons le cas des observateurs intervalles en boucle ouverte où les mesures sont disponibles à des instants discrets. Puis nous traitons le cas où les mesures sont des signaux continus. Nous proposons ainsi une méthode générique pour la construction des observateurs intervalles en utilisant la méthode d'encadrement hybride de l'ensemble des trajectoires d'état générées par un système dynamique incertain, la propriété de positivité des systèmes dynamiques coopératifs et sur un choix approprié de la matrice gain d'observation afin d'assurer la convergence de l'erreur d'observation. Nous présentons plusieurs exemples issus du domaine des bioréacteurs afin d'illustrer cette approche.

Nous montrons dans le septième chapitre l'intérêt de l'intégration numérique garantie des équations différentielles incertaines pour la caractérisation garantie de l'espace d'état atteignable par les systèmes dynamiques continus. En effet, une des approches de vérification de certaines propriétés des systèmes dynamiques telles que la sûreté, les performances ou la stabilité s'appuie sur le calcul de l'espace d'état atteignable et sa comparaison avec les zones de l'espace d'état interdites. Nous présentons plusieurs exemples de calcul d'approximations conservatives de cet espace d'état atteignable pour différents type de systèmes dynamiques continus (linéaire, non linéaire, coopératif, non coopératif) en présence d'incertitude paramétrique.

Nous concluons ce mémoire par quelques perspectives qui portent sur : (i) l'amélioration des performances des méthodes d'intégration numérique garantie des équations différentielles incertaines, notamment la précision des encadrements qui contiennent toutes les solution possibles à chaque pas d'intégration et le temps de calcul ; (ii) l'extension du domaine d'application de ces méthodes ensemblistes à d'autres problèmes de l'automatique.

Chapitre I

Calcul par intervalles

I.1 Introduction

L'arithmétique par intervalles est l'extension logique de l'arithmétique sur les nombres réels aux intervalles [50, 68, 92]. Elle a été initialement introduite pour remédier au problème de la propagation des erreurs d'arrondis induite par la représentation décimale de nombres réels (sur les calculateurs numériques c'est la représentation en virgule flottante). La précision de cette représentation décimale est donc proportionnelle au nombre de chiffres significatifs pris en considération. Cependant, en pratique, ce nombre de chiffres significatifs est limité, ce qui introduit des erreurs d'arrondis. Dans certain cas, la répétition de l'opération d'arrondi, inférieur ou supérieur, sur les résultats intermédiaires du calcul conduit à des approximations très éloignées des solutions recherchées pour le problème traité. Dans ce contexte, la représentation intervalle des nombres réels et l'arithmétique par intervalles sont présentées comme étant des alternatives garanties, dans le sens où les solutions obtenues sont des intervalles contenant de manière certaine les vraies solutions recherchées.

Par ailleurs, au delà de ce problème pratique de représentation et de calcul sur les nombres réels, la représentation intervalle s'avère naturelle et adéquate pour caractériser les mesures et les grandeurs physiques incertaines [102, 103]. En effet, en s'appuyant sur l'incertitude maximale sur les mesures fournies par les fabricants de capteurs ainsi que sur notre connaissance *a priori* des supports physiques des grandeurs aléatoires ou déterministes mal connues, nous pouvons facilement borner ces grandeurs incertaines entre

deux bornes connues.

Dans la suite de ce chapitre, nous allons introduire les notions et les principes de bases de l'arithmétique et du calcul par intervalle, ainsi que la différentiation automatique qui calcule les dérivées partielles successives d'une fonction représentée par un programme informatique.

I.2 Arithmétique d'intervalles

I.2.1 Intervalles

Un intervalle $[x] = [\underline{x}, \overline{x}]$ est un ensemble fermé et convexe de nombres réels $\mathbb{R}$, où $\underline{x}$ et $\overline{x}$ représentent respectivement les bornes inférieure et supérieure.

$$[x] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\} \tag{I.1}$$

On note l'ensemble des intervalles réels par $\mathbb{IR}$:

$$\mathbb{IR} = \{[\underline{x}, \overline{x}] \mid \underline{x}, \overline{x} \in \mathbb{R},\ \underline{x} \leq \overline{x}\} \tag{I.2}$$

Pour les deux raisons citées précédemment, c'est-à-dire la garantie du calcul et la représentation de grandeurs physiques incertaines, les opérations sur les nombres réels sont étendues aux intervalles $[50, 68, 92]$. En effet, tout opérateur binaire $\circ \in \{+, -, \times, \div\}$ sur les nombres réels se généralise aux intervalles de la façon suivante :

$$[x] \circ [y] = \{x \circ y \mid x \in [x] \text{ et } y \in [y]\} \tag{I.3}$$

Ainsi, le résultat de chacun de ces quatre opérateurs intervalles de base est l'image exacte de l'opérateur réel correspondant sur les intervalles de départ $[x]$ et $[y]$. Bien que l'équation (I.3) caractérise mathématiquement ces quatre opérateurs intervalles, son expression en

fonction des opérateurs réels est équivalente à :

$$[x] + [y] = [\underline{x} + \underline{y}, \overline{x} + \overline{y}], \tag{I.4}$$

$$[x] - [y] = [\underline{x} - \overline{y}, \overline{x} - \underline{y}], \tag{I.5}$$

$$[x] \times [y] = [min\{\underline{x} \times \underline{y}, \underline{x} \times \overline{y}, \overline{x} \times \underline{y}, \overline{x} \times \overline{y}\}, max\{\underline{x} \times \underline{y}, \underline{x} \times \overline{y}, \overline{x} \times \underline{y}, \overline{x} \times \overline{y}\}], \tag{I.6}$$

$$\frac{1}{[x]} = \emptyset \text{ si } [x] = [0, 0], \tag{I.7}$$

$$= [\frac{1}{\overline{x}}, \frac{1}{\underline{x}}] \text{ si } 0 \notin [x], \tag{I.8}$$

$$= [\frac{1}{\overline{x}}, +\infty] \text{ si } \underline{x} = 0 \text{ et } \overline{x} > 0, \tag{I.9}$$

$$= [-\infty, \frac{1}{\overline{x}}] \text{ si } \underline{x} < 0 \text{ et } \overline{x} = 0, \tag{I.10}$$

$$= [-\infty, +\infty] \text{ si } \underline{x} < 0 \text{ et } \overline{x} > 0, \tag{I.11}$$

$$[x] \div [y] = [x] \times \frac{1}{[y]}. \tag{I.12}$$

Par ailleurs, nous notons par

$$w([x]) = \overline{x} - \underline{x} \tag{I.13}$$

la taille de l'intervalle $[x]$ et par

$$m([x]) = (\overline{x} + \underline{x})/2 \tag{I.14}$$

son centre. Nous appelons intervalle dégénéré tout intervalle de taille nulle, soit $w(.) = 0$. L'amplitude d'un intervalle est le maximum entre ses bornes en valeur absolue donné par :

$$|[x]| = \max\{|\underline{x}|, |\overline{x}|\}.$$

I.2.2 Vecteurs et matrices intervalles

Un pavé, ou vecteur intervalle $[\mathbf{x}]$ de $\mathbb{R}^n$, est le produit cartésien de n intervalles. Il peut s'écrire sous la forme suivante :

$$[\mathbf{x}] = [\underline{x}_1, \overline{x}_1] \times ... \times [\underline{x}_n, \overline{x}_n] = [\underline{\mathbf{x}}, \overline{\mathbf{x}}], \tag{I.15}$$

où $\underline{\mathbf{x}} = (\underline{x}_1, ..., \underline{x}_n)^t$ et $\overline{\mathbf{x}} = (\overline{x}_1, ..., \overline{x}_n)^t$. L'ensemble de tous les pavés de $\mathbb{R}^n$ sera noté $\mathbb{IR}^n$. De même, nous notons par $[\mathbf{A}] \in \mathbb{IR}^{n \times m}$ une matrice intervalle de n lignes et de m colonnes dont les éléments sont des intervalles $[a_{i,j}], i = 1, ..., n$ et $j = 1, ..., m$.

Ainsi le calcul sur les vecteurs et matrices intervalles est le même que dans le cas de

nombres réels, à l'exception que les opérateurs arithmétiques sur les réels seront remplacés par leurs correspondants intervalles définis par les équations (I.4)-(I.12).

Par exemple, si $[\mathbf{A}] \in \mathbb{IR}^{n\times n}$ d'éléments $[a_{i,j}]$ et $[\mathbf{b}] \in \mathbb{IR}^n$ d'éléments $[b_k]$, alors les éléments de $[\mathbf{c}] = [\mathbf{A}] \times [\mathbf{b}]$ sont donnés par : $[c_i] = \sum_{k=1}^{n} [a_{i,k}][b_k]$.

Exemple : Soient la matrice intervalle $[\mathbf{A}]$ de dimension 2×2 et le vecteur intervalle $[\mathbf{b}]$ de dimension 2×1

$$[\mathbf{A}] = \begin{bmatrix} [0.5,1] & [-1,0] \\ [2,2.5] & [0,1] \end{bmatrix}, [\mathbf{b}] = \begin{bmatrix} [0,0.5] \\ [1,2] \end{bmatrix}$$

alors le produit matrice vecteur intervalles $[\mathbf{A}] \times [\mathbf{b}]$ est :

$$\begin{bmatrix} [0.5,1] & [-1,0] \\ [2,2.5] & [0,1] \end{bmatrix} \times \begin{bmatrix} [0,0.5] \\ [1,2] \end{bmatrix} = \begin{bmatrix} [0.5,1][0,0.5] + [-1,0][1,2] \\ [2,2.5][0,0.5] + [0,1][1,2] \end{bmatrix} = \begin{bmatrix} [-2,0.5] \\ [0,3.25] \end{bmatrix}$$

I.2.2.1 Norme

Dans la littérature, on appelle norme vectorielle d'un vecteur intervalle toute application $||.|| : \mathbb{IR}^n \to \mathbb{R}$ qui respecte les propriétés suivantes :

1. $||[\mathbf{x}]|| \geq 0,\ \forall [\mathbf{x}] \in \mathbb{IR}^n$ et $||[\mathbf{x}]|| = 0 \Leftrightarrow [\mathbf{x}] = 0$.
2. $||[a][\mathbf{x}]|| = |[a]|||[\mathbf{x}]||,\ \forall [\mathbf{a}] \in \mathbb{IR},\ \forall [\mathbf{x}] \in \mathbb{IR}^n$.
3. $||[\mathbf{x}] + [\mathbf{y}]|| \leq ||[\mathbf{x}]|| + ||[\mathbf{y}]||,\ \forall [\mathbf{x}],\ [\mathbf{y}] \in \mathbb{IR}^n$.

Ainsi, on peut facilement vérifier que la norme maximale de vecteurs intervalles définie par

$$||[\mathbf{b}]|| = \max_{1\leq i\leq n} \{|[b_i]|\} \tag{I.16}$$

satisfait les propriétés citées ci-dessus.

Comme pour les vecteurs intervalles de $\mathbb{IR}^n$, les matrices intervalles possèdent une norme maximale. En voici la formule,

$$||[\mathbf{A}]|| = \max_{1\leq i\leq n} \sum_{j=1}^{n} |[a_{ij}]| \tag{I.17}$$

qui satisfait les propriétés ci-dessous exigées par la définition de normes matricielles.

Définition : Soient $[\mathbf{A}], [\mathbf{B}] \in \mathbb{M}_{n\times n}$, ensemble des matrices intervalles de dimension $n \times n$. Alors, l'application $||.|| : \mathbb{M}_{n\times n} \to \mathbb{R}$ est une norme matricielle si elle satisfait les propriétés suivantes :

1. $|||[\mathbf{A}]|| \geq 0,\ \forall [\mathbf{A}] \in \mathbb{M}_{n\times n}$ et $|||[\mathbf{A}]|| = 0 \Leftrightarrow [\mathbf{A}] = 0$.
2. $|||[a][\mathbf{A}]|| = |[a]|||[\mathbf{A}]||$ et ce, pour tout intervalle $[\mathbf{a}] \in \mathbb{IR}$ et tout $[\mathbf{A}]$ dans $\mathbb{M}_{n\times n}$.
3. $|||[\mathbf{A}] + [\mathbf{B}]|| \leq |||[\mathbf{A}]|| + |||[\mathbf{B}]||$ et ce, pour tout $[\mathbf{A}]$ et $[\mathbf{B}]$ dans $\mathbb{M}_{n\times n}$.
4. $|||[\mathbf{A}][\mathbf{B}]|| \leq |||[\mathbf{A}]|||||[\mathbf{B}]||$ et ce, pour tout $[\mathbf{A}]$ et $[\mathbf{B}]$ dans $\mathbb{M}_{n\times n}$.

Par ailleurs, le nombre de conditionnement d'une matrice non-singulière $[\mathbf{A}]$ relatif à une certaine norme est

$$\text{cond}([\mathbf{A}]) = |||[\mathbf{A}]|||||[\mathbf{A}^{-1}]|| \tag{I.18}$$

En pratique, si une matrice est bien conditionnée, alors son nombre de conditionnement sera proche de 1.

I.3 Intervalles et opérations sur les ensembles

Comme les intervalles ont une double nature, c'est-à-dire qu'ils peuvent être considérés comme des ensembles ou comme des paires de nombre réels, les opérations classiques sur les ensembles telles que l'intersection, l'union, l'inclusion ..., sont facilement interprétées par des inégalités conçues à partir des bornes d'intervalles ou vecteurs intervalles. Pour l'inclusion par exemple, on a l'équivalence suivante :
Soient $[\mathbf{x}]$ et $[\mathbf{y}]$ deux pavés de dimension n

$$[\mathbf{x}] \subseteq [\mathbf{y}] \Leftrightarrow \forall i \leq n\ \ \underline{x}_i \geq \underline{y}_i \wedge \overline{x}_i \leq \overline{y}_i$$

et pour l'intersection on a la relation,

$$[\mathbf{x}] \cap [\mathbf{y}] = [\mathbf{z}] = \begin{cases} \forall\ i \leq n \\ [z_i] = \emptyset, \quad \text{si} \quad \overline{x}_i \leq \underline{y}_i \text{ ou } \overline{y}_i \leq \underline{x}_i, \quad \text{sinon,} \\ [z_i] = [\max\{\underline{x}_i, \underline{y}_i\}, \min\{\overline{x}_i, \overline{y}_i\}] \end{cases}$$

L'union entre ces deux pavés, notée $[\mathbf{x}] \sqcup [\mathbf{y}]$, est le plus petit pavé qui contient $[\mathbf{x}] \cup [\mathbf{y}]$,

$$[\mathbf{x}] \sqcup [\mathbf{y}] = [\mathbf{z}] = \begin{cases} \forall\ i \leq n \\ [z_i] = [\min\{\underline{x}_i, \underline{y}_i\}, \max\{\overline{x}_i, \overline{y}_i\}] \end{cases}$$

Dans ce contexte, la distance de Hausdorff entre deux pavés (ensembles), notée $q([\mathbf{x}], [\mathbf{y}])$, est donnée par la relation ci-dessous

$$q([\mathbf{x}], [\mathbf{y}]) = \max_{1\leq i\leq n} \{q([x_i], [y_i])\}$$

où $q([x_i],[y_i])$ représente la distance de Hausdorff entre deux intervalles :

$$q([x_i],[y_i]) = \max\{|\underline{x}_i - \underline{y}_i| - |\overline{x}_i - \overline{y}_i|\}.$$

I.4 Fonction d'inclusion

Soit $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^p$ une fonction vectorielle constituée exclusivement par un nombre fini de constantes, variables, opérateurs arithmétiques et de fonctions standards sin, cos, log, exp, ...Nous définissons l'image de $\mathbf{f}$ sur un pavé $\mathbf{x}$ par $\mathcal{I}m(\mathbf{f},[\mathbf{x}]) = \{\mathbf{f}(\mathbf{x}) \,|\, \mathbf{x} \in [\mathbf{x}]\}$. L'application ensembliste $[\mathbf{f}] : \mathbb{IR}^n \to \mathbb{IR}^p$ est une *fonction d'inclusion* de $\mathbf{f}$ si pour tout $[\mathbf{x}] \in \mathbb{IR}^n$

$$\mathcal{I}m(\mathbf{f},[\mathbf{x}]) \subseteq [\mathbf{f}]([\mathbf{x}]) \tag{I.19}$$

Une fonction d'inclusion est convergente si, pour toute suite de pavés $[\mathbf{x}]$ de $\mathbb{IR}^n$, nous avons

$$w([\mathbf{x}]) \to 0 \Rightarrow w([\mathbf{f}]([\mathbf{x}])) \to \mathbf{0} \tag{I.20}$$

et son taux de convergence est le plus large β tel que

$$\exists \alpha \mid w([\mathbf{f}]([\mathbf{x}])) - w(\mathcal{I}m(\mathbf{f},[\mathbf{x}])) \leq \alpha w([\mathbf{x}])^{\beta}. \tag{I.21}$$

Nous définissons la monotonie d'une fonction d'inclusion $[\mathbf{f}](.)$ par

$$[\mathbf{x}] \subset [\mathbf{y}] \Rightarrow [\mathbf{f}]([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{y}]) \tag{I.22}$$

En pratique, $[\mathbf{f}]([\mathbf{x}])$ n'est pas unique. En effet, elle dépend de la façon avec laquelle la fonction $\mathbf{f}$ est évaluée dans le contexte de l'arithmétique par intervalles. Par exemple, les deux expressions mathématiques $x(y+z)$ et $xy+xz$ sont équivalentes dans le cas de nombres réels. Cependant, dans le contexte intervalles, elles peuvent donner des résultats différents. Par exemple :
si $[x] = [0,1]$, $[y] = [-3,-2]$ et $[z] = [1,2]$ alors

$$\begin{cases} [x]([y]+[z]) = [0,1]([-3,-2]+[1,2]) = [-2,0] \\ [x][y]+[x][z] = [0,1][-3,-2]+[0,1][1,2] = [-3,2] \end{cases} \tag{I.23}$$

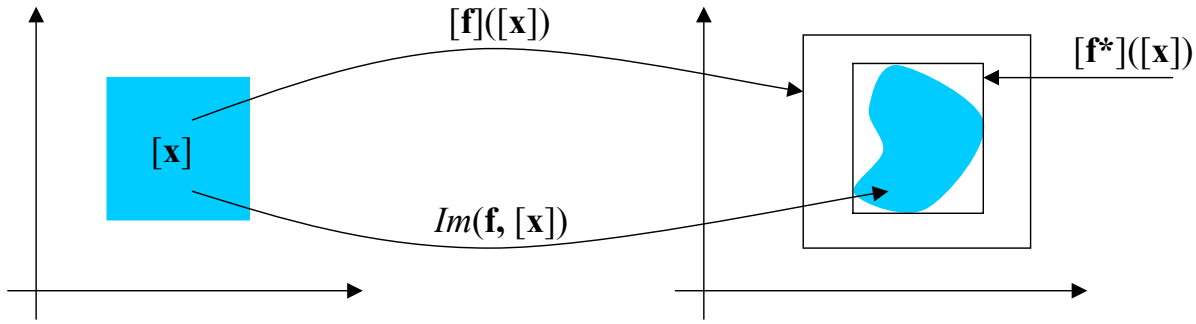


Fig. I.1 *: Exemple de fonctions d'inclusion.* $[\mathbf{f}^*]([\mathbf{x}])$ *fonction d'inclusion minimale*

Généralement, on a $[x]([y]+[z]) \subseteq [x][y]+[x][z]$. Ceci est dû à la multi-occurrence de la variable x dans la seconde expression. Ce phénomène est appelé problème de dépendance des données ou décorrélation des données, c'est-à-dire que chaque occurrence de l'intervalle $[x]$ est considérée, dans l'arithmétique par intervalles, comme étant un intervalle différent. Autrement dit, l'égalité entre $[\mathbf{x}]$ et lui même est perdue. De ce fait, ce phénomène introduit un pessimisme (sur-approximation de l'encadrement de l'image du pavé $[\mathbf{x}]$ par la fonction $\mathbf{f}$) qui est l'inconvénient majeur de l'arithmétique par intervalles.

Dans la littérature, il existe plusieurs méthodes pour le calcul de fonctions d'inclusion convergentes [50]. La plus simple d'entre elles et qui permet de traiter une très grande classe de fonctions consiste à remplacer toutes les variables d'entrées de la fonction par les intervalles correspondants et tous les opérateurs et les fonctions élémentaires définissant la fonction par les opérateurs et fonctions intervalles associés. La fonction d'inclusion ainsi obtenue est appelée fonction d'inclusion naturelle. Une fonction d'inclusion est dite minimale, notée $[\mathbf{f}^*]$, si elle calcule l'encadrement le plus étroit possible de l'image $\mathcal{I}m(\mathbf{f}, [\mathbf{x}])$.

Remarque :

Toute fonction d'inclusion naturelle dans laquelle toutes les variables d'entrée n'apparaissent qu'une seule fois, est minimale. Ainsi, un travail préalable sur l'expression mathématique de la fonction réelle $\mathbf{f}$ est préconisé avant son évaluation intervalles afin de réduire le nombre d'occurrence des variables.

I.4.1 Fonction d'inclusion centrée : Théorème de la valeur moyenne

Une des méthodes de calcul de fonctions d'inclusion, qui réduit sensiblement le pessimisme dû au phénomène de dépendance, est la fonction d'inclusion centrée [50]. Elle s'applique aux fonctions $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^p$ différentiables sur un pavé $[\mathbf{x}] \subseteq \mathbb{R}^n$ et son principe est basé sur le théorème de la valeur moyenne,

$$\forall \mathbf{x} \in [\mathbf{x}], \exists \mathbf{y} \in [\mathbf{x}] \mid \mathbf{f}(\mathbf{x}) = \mathbf{f}(m([\mathbf{x}])) + \mathbf{J}(\mathbf{y})(\mathbf{x} - m([\mathbf{x}])) \tag{I.24}$$

où $\mathbf{J}$ signifie le Jacobien de la fonction $\mathbf{f}$ de dimension $n \times p$. Ainsi on peut affirmer que :

$$\forall \mathbf{x} \in [\mathbf{x}], \mathbf{f}(\mathbf{x}) \in \mathbf{f}(m([\mathbf{x}])) + [\mathbf{J}]([\mathbf{x}])(\mathbf{x} - m([\mathbf{x}])) \tag{I.25}$$

où $[\mathbf{J}]$ est la fonction d'inclusion de J, on obtient alors :

$$\mathbf{f}([\mathbf{x}]) \subseteq \mathbf{f}(m([\mathbf{x}])) + [\mathbf{J}]([\mathbf{x}])([\mathbf{x}] - m([\mathbf{x}])). \tag{I.26}$$

Donc, la fonction intervalle suivante

$$[\mathbf{f}_c]([\mathbf{x}]) \equiv \mathbf{f}(m([\mathbf{x}])) + [\mathbf{J}]([\mathbf{x}])([\mathbf{x}] - m([\mathbf{x}])) \tag{I.27}$$

est une fonction d'inclusion de $\mathbf{f}$.

I.4.2 Fonctions d'inclusion de Taylor

Une idée naturelle consiste à s'intéresser aux développements de Taylor, dans l'espoir d'obtenir de meilleures approximations de $Im(\mathbf{f}, [\mathbf{x}])$. Dans la suite, pour des raisons de simplicité, nous nous limitons aux développements de Taylor d'ordre deux ci-dessus :

$$[\mathbf{f}_T]([\mathbf{x}]) = \mathbf{f}(\hat{\mathbf{x}}) + \mathbf{J}(\hat{\mathbf{x}})([\mathbf{x}] - \hat{\mathbf{x}}) + \frac{1}{2}([\mathbf{x}] - \hat{\mathbf{x}})^T[\mathbf{H}]([\mathbf{x}])([\mathbf{x}] - \hat{\mathbf{x}}), \tag{I.28}$$

où $\hat{\mathbf{x}} = m([\mathbf{x}])$, $\mathbf{J}$ est le gradient de $\mathbf{f}$ et $[\mathbf{H}]([\mathbf{x}])$ est la matrice Hessian intervalle de $\mathbf{f}$. L'avantage d'une telle fonction d'inclusion est de n'utiliser l'arithmétique d'intervalle qu'à l'ordre supérieur, en général sur un nombre réduit d'opérations, et donc elle réduit considérablement l'effet de dépendance.

Pour illustrer ceci, nous présentons un exemple permettant de comparer les performances de chacune de ces fonctions d'inclusion sur un cas d'une fonction mono-variable.

I.4.2.1 Exemple comparatif

Considérons la fonction f définie par

$$f(x) = x^2 + \sin(x),$$

où la variable $x \in [x] = [\frac{2\pi}{3}, \frac{4\pi}{3}]$.
Les fonctions d'inclusion naturelle, centrée et de Taylor d'ordre deux associées à f sont données respectivement par les relations suivantes :

$$\begin{aligned}
[f_n]([x]) &= [x]^2 + \sin([x]) = [3.52046, 18.41199], \\
[f_c]([x]) &= f(\pi) + ([x] - \pi)[f']([x]) = [1.62022, 18.11899], \\
[f_T]([x]) &= f(\pi) + ([x] - \pi)[f'](\pi) + \tfrac{([x]-\pi)^2}{2}[f'']([x]) = [4.33706, 16.97362].
\end{aligned}$$

Par ailleurs, pour cet exemple, comme f est croissante sur l'intervalle $[x]$, il est possible de déterminer sa fonction d'inclusion minimale $[f^*]$:

$$[f^*]([x]) = [\underline{x}^2 + \sin(\underline{x}), \overline{x}^2 + \sin(\overline{x})] = [5.25251, 16.67994].$$

Selon ces quatre approximations de $\mathcal{I}m(f, [x])$, il est évident que la fonction d'inclusion de Taylor est la plus précise. En effet, celle-ci est la plus proche de la fonction d'inclusion minimale. Par contre, pour cet exemple, on remarque que la fonction d'inclusion naturelle donne un meilleur encadrement de $\mathcal{I}m(f, [x])$ que celui obtenu par la fonction d'inclusion centrée. Cela est vrai généralement pour les pavés de taille relativement importante. Cependant, pour les pavés de faible taille il est préférable d'utiliser la fonction d'inclusion centrée plutôt que la fonction d'inclusion naturelle. Par ailleurs, le taux de convergence de la fonction d'inclusion naturelle est au moins linéaire $\beta = 1$ et ceux de des fonctions d'inclusion centrée et Taylor sont au moins quadratiques $\beta = 2$.

I.5 Contracteur

Considérons un problème de satisfaction de contraintes continues formulé comme suit [50] :

$$\mathcal{P} : (\mathbf{f}(\mathbf{x}) = 0, \mathbf{x} \in [\mathbf{x}]). \tag{I.29}$$

où $[\mathbf{x}] = [x_1] \times [x_2] \times ... \times [x_n]$ et $\mathbf{f}$ est une fonction vectorielle de dimension m.

L'ensemble des solutions de $\mathcal{P}$ est défini par la relation

$$\mathbb{S} = \{\mathbf{x} \in [\mathbf{x}] | \mathbf{f}(\mathbf{x}) = 0\} \tag{I.30}$$

En pratique, la majorité des algorithmes basés sur l'arithmétique d'intervalles conçus pour résoudre ce problème consiste à calculer une approximation extérieure de l'ensemble $\mathbb{S}$ en utilisant des techniques de consistance. En effet, contracter $\mathcal{P}$ signifie qu'on cherche à remplacer le pavé initial $[\mathbf{x}]$ par le plus petit pavé possible $[\mathbf{x}]_c$ qui contient aussi l'ensemble $\mathbb{S}$, c'est-à-dire :

$$\mathbb{S} \subset [\mathbf{x}]_c \subset [\mathbf{x}] \tag{I.31}$$

Notons par $\mathcal{C}$ l'opération de contraction. Par ailleurs, parmi les propriétés des contracteurs nous rappelons les trois suivantes :

1. contraction : $\forall [\mathbf{x}]\ \mathcal{C}([\mathbf{x}]) \subseteq [\mathbf{x}]$
2. correction : $\forall [\mathbf{x}]\ [\mathbf{x}] \cap \mathbb{S} \subseteq \mathcal{C}([\mathbf{x}])$
3. monotonie : $[\mathbf{x}] \subset [\mathbf{y}] \Rightarrow \mathcal{C}([\mathbf{x}]) \subset \mathcal{C}([\mathbf{y}])$
4. point fixe : $\exists [\mathbf{y}] \,|\, \mathcal{C}([\mathbf{y}]) = [\mathbf{y}]$.

Dans la littérature, il existe plusieurs types de contracteurs basés sur différents principes [50]. Par exemple, contracteurs de Gauss, de Gauss-Seidel, de Krawczyk, Par propagation de contraintes, Linéarisation parallèle . . . Nous présentons dans la section suivante le principe du contracteur de Newton.

I.5.1 Contracteur de Newton

Le contracteur de Newton est l'extension intervalles de l'algorithme de Newton de l'analyse numérique qui permet de localiser les zéros d'une fonction $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ de classe $\mathcal{C}^1$. Une itération de l'algorithme de Newton dans le cas des nombres réels est définie par

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{J}^{-1}(\mathbf{x}^k)\mathbf{f}(\mathbf{x}^k) \tag{I.32}$$

où $\mathbf{x}^k$ est l'itéré courant et $\mathbf{J}(\mathbf{x}^k)$ le Jacobien de $\mathbf{f}$.

En pratique, cette formule ne peut pas être transposée au calcul par intervalle du fait de son pessimisme qui provoque la divergence de l'algorithme. En effet, il est facile de montrer que dans le cas intervalle $w([\mathbf{x}^{k+1}]) \geq w([\mathbf{x}^k])$.

Pour surmonter ce problème pratique, l'itération (I.32) est remplacée par [39] :

$$[\mathbf{x}^{k+1}] = (m([\mathbf{x}^k]) - [\mathbf{J}]^{-1}([\mathbf{x}^k])\mathbf{f}(m([\mathbf{x}^k]))) \cap [\mathbf{x}^k] \tag{I.33}$$

Il apparaît clairement que cette itération est monotone décroissante pour l'inclusion puisque par construction on a $[\mathbf{x}^{k+1}] \subset [\mathbf{x}^k]$.

I.6 Inversion ensembliste

Le problème du calcul de l'image réciproque, réciproque dans le sens ensembliste, d'un ensemble par une fonction est reconnu comme étant un problème de grande importance. En effet, il est souvent rencontré dans la pratique. Par exemple, le problème de la détermination par des modèles implicites de grandeurs physiques inaccessibles à la mesure peut être formulé comme un problème d'inversion ensembliste. L'idée est de reconstruire par la fonction $\mathbf{f}$ l'ensemble de départ $\mathbb{X}$ en connaissant l'ensemble d'arrivée $\mathbb{Y}$

$$\mathbb{X} = \mathbf{f}^{-1}(\mathbb{Y}) = \{\mathbf{x} \in \mathbb{R}^n \,|\, \mathbf{f}(\mathbf{x}) \in \mathbb{Y}\} \tag{I.34}$$

L'algorithme **SIVIA** (*Set Inversion Via Interval Analysis*) [51] basé sur l'arithmétique d'intervalles permet d'obtenir une approximation garantie de cette image réciproque. En effet, en utilisant un test d'inclusion, **SIVIA** calcule une approximation intérieure $\underline{\mathbb{X}}$ et une approximation extérieure $\overline{\mathbb{X}}$ de l'ensemble $\mathbb{X}$ tel que :

$$\begin{gathered} \underline{\mathbb{X}} \subseteq \mathbb{X} \subseteq \overline{\mathbb{X}} \\ vol(\underline{\mathbb{X}}) \leq vol(\mathbb{X}) \leq vol(\overline{\mathbb{X}}) \end{gathered} \tag{I.35}$$

où la fonction $vol(\mathbb{S})$ représente le volume de l'ensemble $\mathbb{S}$.

Le principe de cet algorithme consiste à partitionner le domaine initial de recherche $\mathbb{D} \in \mathbb{R}^n$ en trois régions disjointes. La première région est l'approximation intérieure $\underline{\mathbb{X}}$, elle contient tous les sous-pavés de $\mathbb{D}$ sur lesquels le test inclusion est satisfait. Par ailleurs, la seconde région est constituée de l'ensemble de sous-pavés dont le test d'inclusion est faux. La troisième et dernière région est formée par l'union de sous-pavés indéterminés, notée $\Delta\mathbb{X}$, dans lesquels aucune décision sur le test d'inclusion ne peut être prise. Pour lever cette ambiguïté sur les sous-pavés indéterminés, l'algorithme **SIVIA** procède à leur bissection puis refait le test d'inclusion jusqu'à un seuil μ fixé *a priori*. Ce seuil de la

bissection a un double rôle : d'une part, il assure l'arrêt de l'algorithme après un nombre fini d'itérations, et d'autre part, il fixe la précision de l'approximation extérieure

$$\overline{\mathbb{X}} = \underline{\mathbb{X}} \cup \Delta\mathbb{X} \tag{I.36}$$

La version récursive de cet algorithme est résumée dans les 5 lignes suivantes :

Algorithme SIVIA(entrées : $\mathbf{f}$, $\mathbb{Y}$, $[\mathbf{x}]$, μ ; sorties : $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$)

1. si $[\mathbf{f}]([\mathbf{x}]) \cap \mathbb{Y} = \emptyset$ {fin} ;
2. si $[\mathbf{f}]([\mathbf{x}]) \subseteq \mathbb{Y}$ alors
3. $\quad \{\underline{\mathbb{X}} := \underline{\mathbb{X}} \cup [\mathbf{x}];\ \overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}];\ \text{fin}\}$;
4. si $w([\mathbf{x}]) < \mu$ alors $\{\overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}];\ \text{fin}\}$;
5. **SIVIA**($\mathbf{f}$, $\mathbb{Y}$, $L[\mathbf{x}]$, μ, $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$) ; **SIVIA**($\mathbf{f}$, $\mathbb{Y}$, $R[\mathbf{x}]$, μ, $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$)

où $L[\mathbf{x}]$ et $R[\mathbf{x}]$ sont les résultats de la bissection du pavé $[\mathbf{x}]$

$$[\mathbf{x}] = L[\mathbf{x}] \cup R[\mathbf{x}].$$

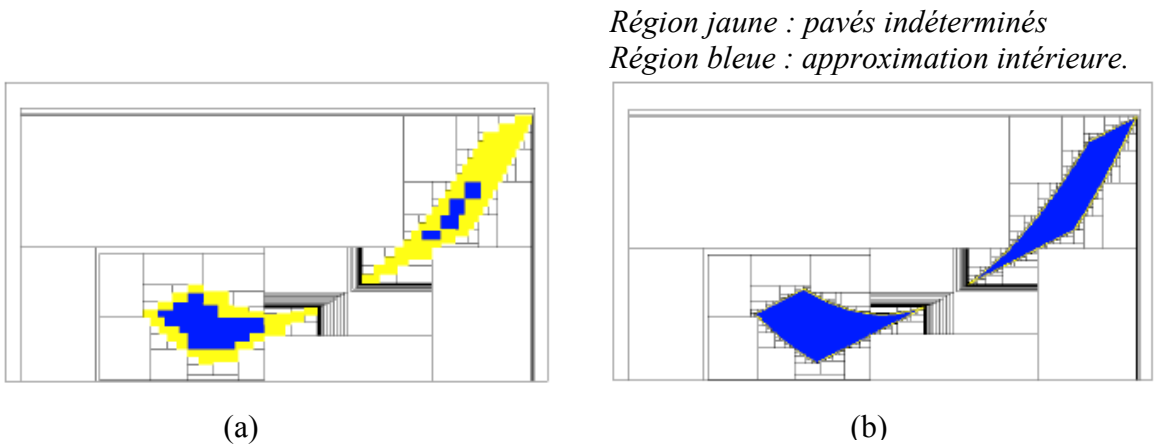


Fig. I.2 *: Exemple d'inversion ensembliste par SIVIA (a) seuil de la bissection μ_1 (b) seuil de la bissection $\mu_2 < \mu_1$*

Les figures I.2.a et I.2.b montrent un exemple où l'algorithme SIVIA a été utilisé pour caractériser un ensemble $\mathbb{X}$ défini par une relation du type (I.34). La région bleue représente l'approximation intérieure $\underline{\mathbb{X}}$ de cet ensemble et la région jaune est l'ensemble des sous-pavés indéterminés et de taille inférieure à μ_1 pour la figure I.2.a et de taille

inférieure à μ_2 ($\mu_2 < \mu_1$) pour la figure I.2.b. De plus, ces deux figures mettent en évidence que la précision de l'approximation extérieure $\overline{\mathbb{X}}$ est inversement proportionnelle au seuil de la bissection μ.

En pratique, le nombre de bissections utilisé dans **SIVIA** est exponentiellement lié à la dimension du problème traité. Ce qui nécessite un temps de calcul et un espace mémoire très importants pour résoudre les problèmes de grande dimension. Ceci est l'inconvénient majeur de l'algorithme **SIVIA**.

I.6.1 Inversion ensembliste avec contraction

Pour pouvoir réduire le nombre de bissections nécessaire pour résoudre un problème d'inversion ensembliste donné par le biais de **SIVIA**, il est intéressant d'interfacer ce dernier avec un contracteur à base de l'arithmétique d'intervalles. En effet, avant l'évaluation du test d'inclusion de **SIVIA**, il est judicieux de contracter le domaine de recherche courant $[\mathbf{x}]$. Ceci réduit généralement le nombre de bissections. L'algorithme ci-dessous montre comment on peut coupler **SIVIA** avec un contracteur intervalle.

Algorithme SIVIA-Contraction(entrées : $\mathbf{f}$, $\mathbb{Y}$, $[\mathbf{x}]$, μ, $\mathcal{C}$; sorties : $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$)

1. $[\mathbf{x}] = \mathcal{C}([\mathbf{x}])$
2. si $[\mathbf{x}] = \emptyset$ alors $\{\text{fin}\}$;
3. si $[\mathbf{f}]([\mathbf{x}]) \subseteq \mathbb{Y}$ alors
4. $\quad \{\underline{\mathbb{X}} := \underline{\mathbb{X}} \cup [\mathbf{x}];\ \overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}];\ \text{fin}\}$;
5. si $w([\mathbf{x}]) < \mu$ alors $\{\overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}];\ \text{fin}\}$;
6. **SIVIA**($\mathbf{f}$, $\mathbb{Y}$, $L[\mathbf{x}]$, μ, $\mathcal{C}$, $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$) ; **SIVIA**($\mathbf{f}$, $\mathbb{Y}$, $R[\mathbf{x}]$, μ, $\mathcal{C}$, $\underline{\mathbb{X}}$, $\overline{\mathbb{X}}$)

Enfin, les figures I.2.a et I.2.b illustrent bien l'effet de la contraction sur les pavés traités à chaque itération.

I.7 Différentiation automatique

Le calcul de la dérivée ou des dérivées partielles d'une fonction est de très grande importance pour l'analyse et la résolution de certains problèmes scientifiques. Par exemple, plusieurs méthodes d'optimisation nécessitent l'accès au gradient voire au Hessien de la

fonction coût à optimiser. Dans la littérature, il existe trois méthodes qui permettent l'implémentation de ce calcul différentiel sur des calculateurs numériques. La première méthode est la différentiation par différences finies (ou différences divisées). Son principe consiste à approcher les dérivées partielles de la fonction considérée en faisant varier insensiblement et successivement chacune des variables, tout en gardant les autres fixées. Cette méthode introduit des erreurs de troncature qui ne sont pas maîtrisables et dans certain cas conduit à des résultats éloignés des vraies solutions recherchées.

La deuxième méthode est la différentiation symbolique basée sur le calcul formel. Celle-ci suppose que la fonction que l'on cherche à dériver admet une représentation symbolique au moyen de formules mathématiques. Elle obtient alors une représentation, également symbolique, de la fonction dérivée. Ceci peut être vu comme une automatisation de ce qu'un être humain a appris à faire sur une feuille de papier. Or, dans certains cas, devoir construire une représentation symbolique des fonctions complexes serait une tâche fastidieuse et coûteuse en terme d'espace mémoire nécessaire.

Dans la suite, c'est la troisième méthode connue sous le nom de différentiation automatique [78, 27] qui s'avère être le mieux adapté. La différentiation automatique est une technique permettant d'obtenir les dérivées exactes (aux erreurs d'arrondis près) d'une fonction représentée par un programme informatique. De plus, si celle-ci est utilisée conjointement avec l'arithmétique d'intervalles, elle fournit des encadrements garantis des solutions. Par ailleurs, il ne sera en général pas question d'essayer d'obtenir une représentation visuelle ou symbolique de la fonction dérivée.

I.7.1 Principe de la différentiation automatique

Pour des raisons de simplicité, nous considérons le cas des fonctions scalaires réelles

$$f : \mathbb{R}^n \to \mathbb{R}$$

de n variables réelles $x_1, ..., x_n$, que nous appellerons variables indépendantes. L'hypothèse de base de la différentiation automatique est que cette fonction est connue par la donnée d'un programme de calcul qui, en un point $\mathbf{x} = (x_1, ..., x_n)$ de $\mathbb{R}$ fournit la valeur $f(\mathbf{x})$. On cherche alors à générer un programme qui en un point $\mathbf{x} \in \mathbb{R}$ donné, calculera la valeur numérique $\nabla f(\mathbf{x})$ du gradient de f en $\mathbf{x}$, c'est-à-dire l'ensemble de ses dérivées partielles premières $\frac{\partial f(\mathbf{x})}{\partial x_1}, ..., \frac{\partial f(\mathbf{x})}{\partial x_n}$.

En pratique, un programme informatique peut représenter une fonction très complexe. En particulier, cela implique que le différentiateur doit pouvoir traiter des fonctions qui ne sont pas nécessairement représentables par des formules mathématiques explicites. Pour contourner cette complexité, il est judicieux de procéder par étapes. Un code de différentiation automatique est en effet basé sur une méthodologie dictant les actions à prendre en chaque circonstance. Cette méthodologie repose elle-même sur un modèle de programme. On rencontre généralement deux types de modèles, modèle simple et modèle étendu. Le premier est moins général que le second dans le sens où il demande plus de transformations du programme (fonction) à différencier. Par ailleurs, tous deux supposent que celui-ci est formé d'une suite de K instructions d'affectation.

I.7.1.1 Le modèle simple

Dans le cas le plus simple, un programme calculant une fonction f de n variables indépendantes $x_1, ..., x_n$, le fait en utilisant un certain nombre de variables auxiliaires $x_{n+1}, ..., x_N$ $(N > n)$, que nous appellerons variables intermédiaires. Si le programme est consistant, chaque fois qu'une nouvelle variable intermédiaire x_i est introduite, sa valeur est calculée à partir d'autres variables dont la valeur doit être connue au moment du calcul de x_i. En adoptant une numérotation convenable des variables, on peut supposer que la variable introduite x_i a un numéro d'ordre supérieur à celui des variables déjà calculées. Par conséquent, la première fois que x_i recevra une valeur, ce sera au moyen d'une instruction d'affectation de la forme :

$$x_i := \varphi_i(x_1, ..., x_{i-1}),$$

où φ_i ne dépend que de $x_1, ..., x_{i-1}$. Les fonctions φ_i sont appelées fonctions intermédiaires.

L'hypothèse de base du modèle simple est qu'une nouvelle variable est introduite à chaque instruction d'affectation. Autrement dit, un programme obéissant à ce modèle pourra être schématisé de la manière suivante :

$$\begin{cases} \textbf{for } i := n+1 \textbf{ to } N \textbf{ do } x_i := \varphi_i(x_1, ..., x_{i-1}); \\ f := x_N \end{cases}$$

Ce programme manipule donc N variables et la valeur de $f(\mathbf{x})$ est récupérée dans la variable x_N. Il y a $K = N - n$ instructions d'affectation. En général, les fonctions intermédiaires φ_i ne dépendent que d'une partie des variables calculées précédemment : il

existe une partie P_i de $\{1, ..., i-1\}$ telle que $x_i = \varphi_i(x_{P_i})$. On note $x_{P_i} = \{x_j : j \in P_i\}$. Avec ces notions, l'algorithme précédent peut se réduire :

$$\begin{cases} \textbf{for } i := n+1 \textbf{ to } N \textbf{ do } x_i := \varphi_i(x_{P_i}); \\ f := x_N \end{cases} \tag{I.37}$$

Il est fréquent de trouver dans les programmes des variables intermédiaires qui sont redéfinies, c'est-à-dire des variables qui reçoivent des valeurs différentes en divers endroits du programme. Cela n'est pas permis dans le modèle simple. Dès lors, les algorithmes de différentiation automatique basés sur ce modèle se voient forcés de transformer le programme original en introduisant des variables intermédiaires supplémentaires à chaque redéfinition de variables.

I.7.1.2 Le modèle étendu

Dans ce qui suit, nous cherchons à remédier à l'inconvénient majeur du modèle simple qui ne lui permet pas de prendre en compte, sans modification, des programmes où des variables intermédiaires seraient redéfinies.

Dans le contexte du modèle étendu, nous noterons aussi par $\mathbf{x} = (x_1, ..., x_n)$ le vecteur des n variables indépendantes et nous noterons par $\tilde{\mathbf{x}} = (x_1, ..., x_n, ..., x_N)$ le vecteur formé de toutes les variables intervenant dans le programme. Dans ce modèle, les fonctions intermédiaires φ_k peuvent dépendre de toutes les variables $x_1, ..., x_n, ..., x_N$. On numérote de 1 à K les fonctions intermédiaires et on note μ_k l'indice de la variable modifiée par l'instruction d'affectation d'ordre k. Avec ces conventions, le modèle étendu est schématisé par l'algorithme suivant :

$$\begin{cases} \textbf{for } k := 1 \textbf{ to } K \textbf{ do } x_i := \varphi_{\mu_k}(x_{P_k}); \\ f := x_K \end{cases} \tag{I.38}$$

Nous allons faire sur le programme (I.38) l'hypothèse de consistance suivante : chaque instruction d'affectation n'utilise que des variables déjà définies par les instructions d'affectation précédentes. Afin d'être plus précis, désignant par ν_i l'indice de la première instruction qui donne une valeur à la variable x_i. Autrement dit, $k = \nu_i$ si et seulement si $\mu_k = i$ et $\nu_l \neq i$, pour $l = 1, ..., k-1$. Avec ces notations, l'hypothèse de consistance s'écrit :

$$\text{pour } n+1 \leq i \leq N \text{ et } 1 \leq k \leq \nu_i, \text{ on a } i \notin P_k. \tag{I.39}$$

Notons que dans ce modèle les variables indépendantes peuvent être modifiées au cours du programme.

Remarque : afin de pouvoir implémenter aisément un différentiateur automatique, on est souvent amené à décomposer f en opérations élémentaires, de telle sorte que les fonctions intermédiaires φ_i soient les opérations et les fonctions de base : addition, soustraction, multiplication, division, exponentielle, logarithme, fonctions trigonométriques, . . . etc.

Exemple :

Soit la fonction $f : \mathbb{R}^4 \to \mathbb{R}$

$$f(\mathbf{x}) = \frac{x_1 e^{x_2} + x_3}{3 sin x_4}$$

pourra être décomposée en

$$\begin{cases} x_5 := exp(x_2); \\ x_6 := x_1 * x_5; \\ x_7 := x_6 + x_3; \\ x_8 := sin(x_4); \\ x_9 := 3 * x_8; \\ x_{10} := \frac{x_7}{x_8}; \\ f := x_{10} \end{cases}$$

I.7.1.3 Différentiation automatique en mode direct

Le mode direct de la différentiation automatique consiste à calculer la fonction et son gradient simultanément. Chaque fois qu'une variable intermédiaire est modifiée par l'instruction $x_{\mu_k} := \varphi_k(x_{P_k})$, son gradient ∇x_{μ_k} est calculé en utilisant la règle de dérivation des fonctions composées.

Désignons par ∇x_i, $1 \leq i \leq N$, le gradient de la variable (indépendante ou intermédiaire) x_i par rapport aux variables indépendantes $x_1, ..., x_n$. Il s'agit donc de N vecteurs ayant chacun n composantes. On a bien sûr,

$$\nabla x_i = e_i, \text{ pour } 1 \leq i \leq n$$

où $e_i := (0, ..., 1, 0, ..., 0)^T$ est le i-ième vecteur de la base canonique de $\mathbb{R}$. Pour $1 \leq k \leq K$, la définition de x_{μ_k} et la règle de dérivation des fonctions composées donnent :

$$\nabla x_{\mu_k} = \sum_{j \in P_k} \frac{\partial \varphi_k}{\partial x_j} \nabla x_j, \quad 1 \leq k \leq K \tag{I.40}$$

Le mode direct consiste donc à faire ce type de calcul pour chaque instruction du code original. Par convention, la formule (I.40) s'écrit $\nabla x_{\mu_k} = 0$ si $P_k = \emptyset$. Dès lors, $\nabla x_i = 0$ chaque fois qu'une variable intermédiaire x_i est initialisée.

Si on note $\varphi_{k,j} \equiv \frac{\partial \varphi_k}{\partial x_j}$, l'algorithme de calcul du gradient ∇f s'écrit :

$$\begin{cases} \textbf{for } i := 1 \textbf{ to } n \textbf{ do } \nabla x_i := e_i; \\ \textbf{for } k := 1 \textbf{ to } K \textbf{ do } \{ \\ \quad \nabla x_{\mu_k} := \sum_{j \in P_k} \varphi_{k,j} \nabla x_j; \\ \quad x_{\mu_k} := \varphi_k(x_{P_k}); \\ \} \\ \nabla f := \nabla x_N; \\ f := x_N. \end{cases}$$

Alors, on voit qu'il suffit de compléter le programme original par quelques instructions supplémentaires sans modifier le sens de l'exécution. On dit que le calcul du gradient est propagé de façon progressive.

Note : Il existe d'autres algorithmes qui permettent la mise en œuvre informatique de ce mode direct de la différentiation automatique. Leurs objectifs est de réduire le temps de calcul ou l'espace mémoire nécessaire pour réaliser ce calcul différentiel. De plus, la différentiation automatique a un mode de fonctionnement, dit mode inverse, qui s'apparente au calcul du gradient par la méthode de l'état adjoint dans les problèmes de contrôle optimal.

La différentiation automatique permet aussi le calcul des dérivées d'ordre supérieur qui est utile dans nombreuses situations, par exemple pour le calcul de fonctions d'inclusion de Taylor d'ordre supérieur. De plus, il existe plusieurs outils logiciels qui implémentent les méthodes de différentiation automatique, par exemple : la boite à outils INTLAB [39] de MATLAB, le package FADBAD++ (http ://www2.imm.dtu.dk/ km/FADBAD/) développé en C++ par Ole Stauning et Claus Bendtsen, ...
Dans le cadre de cette thèse, nous utiliserons le package FADBAD++ pour générer les coefficients du développement en série de Taylor utilisés dans les différents schémas d'intégration numérique des équations différentielles.

I.8 Conclusion

Dans ce premier chapitre nous avons introduit quelques notions de base de l'arithmétique et du calcul intervalle, et nous avons évoqués les raisons pour lesquelles la représentation intervalles ou vecteurs intervalles, des grandeurs incertaines est intéressante. De plus, nous avons rappelé le principe de la différentiation automatique qui permet de calculer les dérivées exactes et garanties de fonctions si elle est utilisée conjointement avec l'arithmétique d'intervalles. Dans le chapitre suivant, nous allons traiter le problème de l'intégration numérique des équations différentielles incertaines dont les incertitudes sont représentées par des pavés d'état initial et de paramètres.

Chapitre II

Intégration numérique garantie d'équations différentielles ordinaires

II.1 Introduction

Le problème de la valeur initiale (PVI) des équations différentielles ordinaires (EDO) est souvent rencontré dans plusieurs applications scientifiques et technologiques. Généralement, l'état initial et/ou les paramètres de ces EDO sont mal connus, mais ils peuvent être naturellement représentés par des intervalles bornés conçus à partir de la connaissance *a priori* sur les problèmes traités. Dans ce contexte, il est souhaitable de pouvoir calculer ou caractériser un encadrement de toutes les solutions possibles de ces EDO sur un horizon de temps fini. Cependant, les méthodes classiques d'intégration numérique des EDO ne sont pas adaptées. En effet, pour résoudre ce problème avec ces méthodes classiques, il faudrait que l'on intègre l'EDO incertaine une infinité de fois, ce qui est irréalisable. Par ailleurs, le calcul par intervalles, en particulier les méthodes d'intégration numérique garantie permettent d'atteindre cet objectif d'une manière garantie, dans le sens où aucune solution de l'EDO ne sera perdue.

D'autre part, même dans le cas des EDO certaines, c'est-à-dire pour lesquelles l'état initial et les paramètres sont parfaitement connus, les méthodes classiques ne calculent qu'une approximation de la vraie solution avec une tolérance fixée par l'utilisateur, sans pouvoir fournir une borne garantie de l'erreur de troncature. En revanche, les méthodes intervalles non seulement donnent un encadrement garanti de cette erreur, mais de plus

vérifient l'existence et l'unicité de la solution et ainsi elles détectent l'absence de solution si l'EDO n'en admet aucune.

Beaucoup de travaux abordant le problème de la valeur initiale, en utilisant le calcul par intervalles, ont été réalisés. Par exemple, les méthodes qui s'appuient sur le développement en série de Taylor par rapport au temps [68, 59, 69, 84, 85]. Dans cette classe plusieurs logiciels qui implémentent ces méthodes sont disponibles : AWA [60], VNODE [69, 71]. Il existe aussi des méthodes qui reposent sur le développement en série de Taylor par rapport au temps et par rapport aux variables [10, 47, 11] ; dans cette catégorie de méthodes on trouve le code COSY INFINITY [62] qui permet leur mise en œuvre informatique.

Dans ce chapitre, nous allons présenter les méthodes d'intégration garantie basées sur les modèles de Taylor intervalles par rapport au temps [68, 59, 84, 70].

II.2 Présentation du problème

Considérons le système d'équations différentielles autonomes du premier ordre suivant :

$$\begin{cases} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}) \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \in [\mathbf{x}_0] \end{cases} \tag{II.1}$$

où $t \in [t_0, t_f]$, avec $t_f \geq t_0$. La fonction vectorielle $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est supposée $(k-1)$ continûment différentiable par rapport aux composantes $x_i,\ i = 1, ..., n$ du vecteur $\mathbf{x}$. Cela signifie que $\mathbf{f}$ est constituée exclusivement d'un nombre fini de constantes, variables, opérations élémentaires et fonctions standards différentiables. Le vecteur $\mathbf{x}_0$ représente le vecteur de conditions initiales qui sont incertaines et pour cette raison on les représente par un vecteur intervalles $[\mathbf{x}_0]$ qui contient toutes les valeurs possibles de $\mathbf{x}$ à l'instant t_0.

Dans ce contexte, l'objectif des méthodes d'intégration numérique garantie basées sur l'arithmétique par intervalles, consiste à caractériser, à chaque instant d'une séquence temporelle croissante $t_0 < t_1 < ... < t_f$, construite avec un pas d'intégration $h_j = t_{j+1} - t_j$ qui n'est pas nécessairement constant, toutes les solutions possibles de (II.1). Nous notons par $\mathbf{x}(t; t_j, \mathbf{x}_j)$ la solution de $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x})$ pour une condition initiale $\mathbf{x} = \mathbf{x}_j$ à l'instant $t = t_j$. L'ensemble de solutions engendrées par un domaine de conditions initiales $[\mathbf{x}_j]$ est

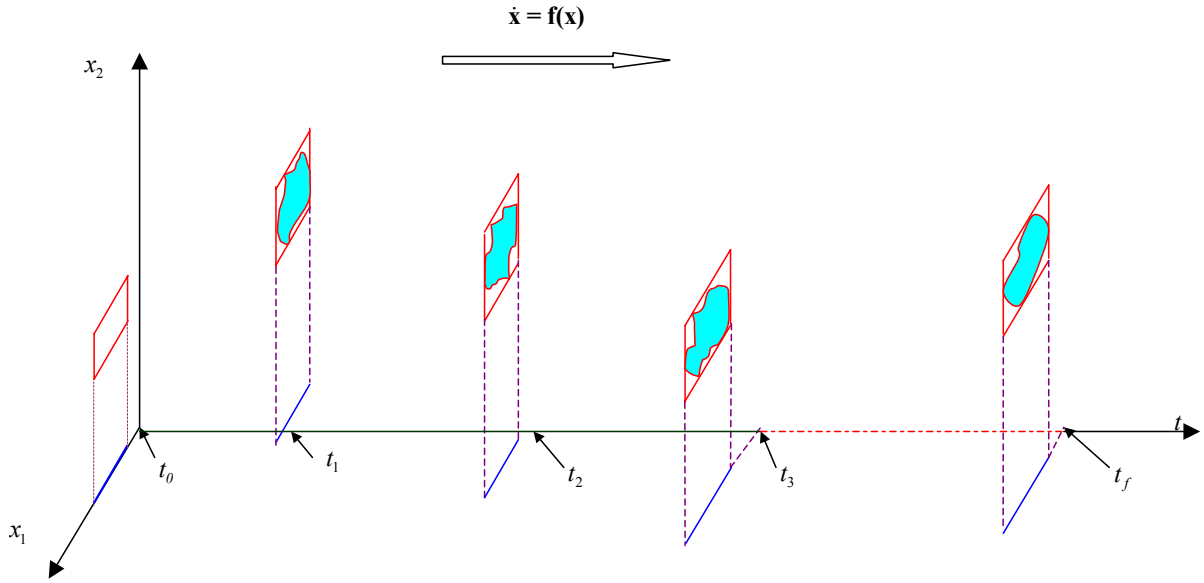


Fig. II.1 *: Principe d'intégration numérique garantie*

noté par la relation :

$$\mathbf{x}(t; t_j, [\mathbf{x}_j]) = \{\mathbf{x}(t; t_j, \mathbf{x}_j) \mid \mathbf{x}_j \in [\mathbf{x}_j]\} \tag{II.2}$$

Donc, nous cherchons à chaque pas d'intégration un encadrement $[\mathbf{x}_j], j = 1, ..., f$ tel que

$$\mathbf{x}(t_j; t_0, [\mathbf{x}_0]) \subseteq [\mathbf{x}_j] \tag{II.3}$$

Pour réaliser cet objectif, le développement en série de Taylor étendu aux intervalles est bien adapté pour les deux raisons suivantes : (i) ce développement fournit une expression explicite de l'erreur de troncature, ce qui permet de la borner d'une façon garantie en utilisant le calcul par intervalles, (ii) le développement de Taylor réduit efficacement le pessimisme dû au phénomène de dépendance lié au calcul par intervalles. Il faut noter que ce phénomène de dépendance est la cause principale pour laquelle les différentes tentatives d'étendre les méthodes classiques d'intégration aux intervalles ont échoué.

Pour ces raisons, les méthodes d'intégration numérique garantie que nous allons présenter dans la suite reposent sur le principe du développement de Taylor intervalles par rapport au temps t. Notons que les coefficients du développement de Taylor $\mathbf{f}^{[i]}$ de $\mathbf{f}$ sont calculables en utilisant conjointement l'arithmétique d'intervalles et la différentiation au-

tomatique selon l'algorithme récursif suivant :

$$\begin{aligned} \mathbf{f}^{[0]}([\mathbf{x}_j]) &= [\mathbf{x}_j] \\ \mathbf{f}^{[1]}([\mathbf{x}_j]) &= \mathbf{f}([\mathbf{x}_j]) \\ \mathbf{f}^{[i]}([\mathbf{x}_j]) &= \frac{1}{i}(\frac{\partial \mathbf{f}^{[i-1]}}{\partial \mathbf{x}}\mathbf{f})([\mathbf{x}_j]) \end{aligned} \tag{II.4}$$

Remarque : Le fait de considérer (II.1) autonome et sans paramètres incertains n'est pas restrictif. En effet, il est possible de transformer tout système non autonome avec paramètres incertains

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{p}, t)$$

en un système du type (II.1). Pour ce faire, on doit inclure le temps t et le vecteur de paramètres incertains $\mathbf{p} \in [\mathbf{p}]$ dans le vecteur $\mathbf{x}$. Ainsi, les dynamiques et les conditions initiales de ces nouvelles composantes du vecteur d'état sont données par,

$$\begin{aligned} &f_{n+1}(\mathbf{x}) = 1, \text{ et } x_{0,n+1} = 0 \\ &f_m(\mathbf{x}) = 0, \text{ pour } m = n+2, ..., n+n_p \text{ et } x_{0,m} \in [p_m] \end{aligned}$$

où n_p est la dimension du vecteur de paramètres $\mathbf{p}$.

II.3 Méthodes de Taylor intervalles

Les méthodes de Taylor intervalles sont des méthodes à un pas durant lequel deux algorithmes sont exécutés l'un après l'autre. On peut les considérer comme les étapes *prédiction-correction* des méthodes classiques d'intégration. En effet, le premier algorithme consiste à vérifier l'existence et l'unicité de la solution de l'EDO sur un pas d'intégration et de calculer, en même temps, un encadrement garanti de toute la trajectoire solution le long de ce pas d'intégration. Par la suite, le deuxième algorithme réduit cet encadrement de solutions par un développement de Taylor intervalles pour fournir un encadrement de la vraie solution à chaque pas d'intégration.

II.3.1 Vérification de l'existence et de l'unicité de la solution : Etape de prédiction

Supposons qu'à un instant donné t_j nous disposons d'un encadrement $[\mathbf{x}_j]$ de solutions $\mathbf{x}(t_j; t_0, [\mathbf{x}_0])$ de (II.1). La première étape (*Algorithme I*) des méthodes de Taylor intervalles

consiste à trouver un pas d'intégration $h_j = t_{j+1} - t_j > 0$ et un encadrement $[\tilde{\mathbf{x}}_j]$, qu'on appelle solution *a priori*, tel qu'on peut garantir l'existence et l'unicité de solutions $\mathbf{x}(t)$ pour tout $\mathbf{x}_j \in [\mathbf{x}_j]$ et $t \in [t_j, t_{j+1}]$, c'est-à-dire cette solution *a priori* doit satisfaire l'inclusion suivante :

$$\mathbf{x}(t; t_j, \mathbf{x}_j) \subseteq [\tilde{\mathbf{x}}_j], \forall t \in [t_j, t_{j+1}] \wedge \forall \mathbf{x}_j \in [\mathbf{x}_j]. \tag{II.5}$$

Pour cela, nous devrons généraliser l'utilisation de l'opérateur de Picard-Lindelöf

$$(P\mathbf{x})(t) = \mathbf{x}_i + \int_{t_i}^{t} \mathbf{f}(\mathbf{x}(s))ds \tag{II.6}$$

et le théorème de point-fixe de Banach [68] au cas intervalles.

L'un des avantages du calcul par intervalles, dans ce contexte, est sa capacité de vérifier numériquement le théorème de point fixe de Banach. En effet, le calcul par intervalles peut facilement vérifier qu'une application $\mathbf{\Phi}$ de $\mathbb{R}^n$ dans $\mathbb{R}^n$ est contractante en calculant simplement sa fonction inclusion.

Theorème du Point Fixe : *Soient $\mathbb{D}$ un espace métrique complet (non vide) fermé dans $\mathbb{R}^n$. Si l'application continue $\mathbf{\Phi} : \mathbb{D} \to \mathbb{D}$ est telle que l'image $\mathcal{I}m(\mathbf{\Phi}, \mathbb{D}) \subseteq \mathbb{D}$, alors $\mathbf{\Phi}$ possède un point fixe, c'est-à-dire qu'il existe $\mathbf{x} \in \mathbb{D}$ tel que $\mathbf{\Phi}(\mathbf{x}) = \mathbf{x}$.*

Par conséquent, en posant $\mathbf{\Phi}(\mathbf{x}(t)) = (P\mathbf{x})(t)$, il suffira de montrer que l'application intervalles $[\mathbf{\Phi}]$ est contractante sur un domaine $[\tilde{\mathbf{x}}_j]$,

$$[\mathbf{\Phi}]([\tilde{\mathbf{x}}_j]) \subseteq [\tilde{\mathbf{x}}_j] \tag{II.7}$$

pour assurer l'existence et l'unicité d'une solution de (II.1) pour une condition initiale $\mathbf{x}_0 = \mathbf{x}_i$ et affirmer que quel que soit $t \in [t_j, t_j + h_j]$, on a $\mathbf{x}(t) \in [\tilde{\mathbf{x}}_j]$.

Par ailleurs, l'application intervalle $[\mathbf{\Phi}]$ peut être construite de la façon suivante :

$$\begin{array}{rcl} \mathbf{\Phi}(\mathbf{x}(t)) = (P\mathbf{x})(t) & = & \mathbf{x}_i + \int_{t_i}^{t} \mathbf{f}(\mathbf{x}(s))ds \\ & \in & \mathbf{x}_i + \int_{t_i}^{t} \mathbf{f}([\tilde{\mathbf{x}}_j])ds \\ & \subseteq & \mathbf{x}_i + [0, h_j]\mathbf{f}([\tilde{\mathbf{x}}_j]) = [\mathbf{\Phi}]([\tilde{\mathbf{x}}_j]) \end{array} \tag{II.8}$$

Dés lors, pour vérifier l'existence et l'unicité de solutions de (II.1) pour tout $\mathbf{x}_j \in [\mathbf{x}_j]$ et $t \in [t_j, t_j + h_j]$, on doit raisonner de la même manière, c'est-à-dire que l'on cherche

toujours à vérifier l'inclusion (II.7), mais en calculant $[\mathbf{\Phi}]$ par la relation suivante :

$$[\mathbf{\Phi}]([\tilde{\mathbf{x}}_j]) = [\mathbf{x}_i] + [0, h_j]\mathbf{f}([\tilde{\mathbf{x}}_j]) \tag{II.9}$$

Ainsi, grâce au calcul par intervalles on peut prouver numériquement l'existence et l'unicité de solutions du problème (II.1) en vérifiant l'inclusion ci-dessous :

$$[\mathbf{x}_i] + [0, h_j]\mathbf{f}([\tilde{\mathbf{x}}_j]) \subseteq [\tilde{\mathbf{x}}_j] \tag{II.10}$$

En pratique les méthodes de vérification basées sur l'inclusion (II.10), appelées méthodes du premier ordre, génèrent des pas d'intégration d'une largeur très courte, ce qui représente un sérieux inconvénient. Cependant, il existe d'autres méthodes de vérification qui utilisent des encadrements polynomiaux ou le développement en série de Taylor intervalles d'ordre supérieur [19, 69, 72]. Ces méthodes sont capables de générer des pas d'intégration d'une largeur supérieure à celle obtenue avec les méthodes de premier ordre. Par exemple, Corliss et Rihm [19] ont proposé une méthode de vérification qui s'appuie sur l'inclusion suivante :

$$[\mathbf{\Phi}]([\tilde{\mathbf{x}}_j]) = [\mathbf{x}_j] + \sum_{i=1}^{k-1}[0, h_j^i]\mathbf{f}^{[i]}([\mathbf{x}_j]) + [0, h_j^k]\mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j]) \subseteq [\tilde{\mathbf{x}}_j] \tag{II.11}$$

où les $\mathbf{f}^{[i]}$ sont les coefficients du développement de Taylor qui peuvent être calculés soit par les méthodes de calcul symbolique ou par les techniques de différentiation automatique. De plus, Nedialkov [69, 72] a implémenté cette méthode dans son logiciel dédié à l'intégration numérique garantie VNODE [69, 71].

Ainsi, nous résumons dans l'algorithme ci-dessous le principe de cette première étape des méthodes d'intégration numérique garantie des EDO basées sur le calcul intervalle,

Algorithme I : (Etape de prédiction)

1. Estimer une solution *a priori* $[\tilde{\mathbf{x}}_j]_0$ et un pas d'intégration h_j ;
2. $[\tilde{\mathbf{x}}_j]_1 := [\mathbf{\Phi}]([\tilde{\mathbf{x}}_j]_0)$;
3. SI $[\tilde{\mathbf{x}}_j]_1 \subseteq [\tilde{\mathbf{x}}_j]_0$ FIN.
4. SINON Réessayer une autre fois en réduisant h_j ou en élargissant $[\tilde{\mathbf{x}}_j]_0$;
5. FIN.

Enfin, cette étape de vérification de l'existence et l'unicité de solutions est considérée comme étant l'étape de prédiction de solutions. En effet, la solution *a priori* $[\tilde{\mathbf{x}}_j]$ calculée

dans cette étape constitue une approximation conservative de la vraie solution à l'instant t_{j+1}. Pour améliorer cette estimation on utilisera un développement de Taylor intervalle dans l'étape de correction (*Algorithme* II) qui consiste à contracter $[\tilde{\mathbf{x}}_j]$ pour obtenir un encadrement ou une fonction d'inclusion la plus étroite possible contenant l'ensemble des solutions $[\mathbf{x}_{j+1}]$ du (II.1) à l'instant t_{j+1}.

II.3.2 Calcul de l'encadrement garanti de solutions : Etape de correction

L'idée principale des méthodes d'intégration numérique garantie consiste à progresser dans le processus d'intégration en utilisant le développement en série de Taylor intervalle. Celui-ci permet : (*i*) une meilleure maîtrise du pessimisme dû au phénomène de dépendance ; (*ii*) de fournir un encadrement garanti de l'erreur de troncature. Dans cette section, nous présentons quelques schémas d'intégration fondés sur le développement de Taylor intervalle.

II.3.2.1 Principe

A l'issu de l'exécution de l'*Algorithme* I (étape de prédiction), nous disposons d'une solution *a priori* $[\tilde{\mathbf{x}}_j]$ tel que :

$$\mathbf{x}(t; t_j, [\mathbf{x}_j]) \subseteq [\tilde{\mathbf{x}}_j],\ \forall \mathbf{x}_j \in [\mathbf{x}_j] \text{ et } \forall t \in [t_j, t_{j+1}].$$

Ainsi l'objectif de l'*Algorithme* II (étape de correction) consiste à calculer un pavé $[\mathbf{x}_{j+1}] \subseteq [\tilde{\mathbf{x}}_j]$ pour lequel l'inclusion ci-dessus doit être satisfaite,

$$\mathbf{x}(t_{j+1}; t_0, [\mathbf{x}_0]) \subseteq [\mathbf{x}_{j+1}].$$

Pour ce faire, nous considérons le développement en série de Taylor

$$\mathbf{x}_{j+1} = \mathbf{x}_j + \sum_{i=1}^{k-1} h_j^i \mathbf{f}^{[i]}(\mathbf{x}_j) + h_j^k \mathbf{f}^{[k]}(\mathbf{x}; t_j, t_{j+1}), \tag{II.12}$$

où $\mathbf{x}_j \in [\mathbf{x}_j]$. Notons par $\mathbf{f}^{[k]}(\mathbf{x}; t_j, t_{j+1})$ le terme reste du développement de Taylor évalué pour un certain instant $t_c \in [t_j, t_{j+1}]$ généralement inconnu. Par ailleurs, comme $t_c \in [t_j, t_{j+1}]$ alors le vecteur $\mathbf{x}(t_c) \in [\tilde{\mathbf{x}}_j]$. Ainsi, on peut obtenir facilement un encadrement garanti de ce terme reste en calculant sa fonction d'inclusion $\mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j])$. Pour généraliser la

relation (II.12) au cas de d'intervalles, c'est-à-dire pour tout $\mathbf{x}_j \in [\mathbf{x}_j]$, il suffit de remplacer $\mathbf{x}_j$ par son correspondant intervalle $[\mathbf{x}_j]$ et le terme reste par sa fonction d'inclusion,

$$[\mathbf{x}_{j+1}] = [\mathbf{x}_j] + \sum_{i=1}^{k-1} h_j^i \mathbf{f}^{[i]}([\mathbf{x}_j]) + h_j^k \mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j]). \tag{II.13}$$

En procédant de la sorte, on obtient effectivement des encadrements garantis de toutes les solutions possibles de (II.1) aux instants $t_{j+1},\ j \geq 0$. En revanche, la taille de ces encadrements est croissante en fonction de j, même pour les cas où le vrai ensemble solution de (II.1) est contractant en fonction de t. Ceci est mis en évidence par la relation suivante :

$$w([\mathbf{x}_{j+1}]) = w([\mathbf{x}_j]) + \sum_{i=1}^{k-1} h_j^i w(\mathbf{f}^{[i]}([\mathbf{x}_j])) + h_j^k w(\mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j])) \geq w([\mathbf{x}_j]).$$

En pratique, l'utilisation de la formule de la valeur moyenne (I.27) pour le calcul de fonctions d'inclusions des coefficients de Taylor $\mathbf{f}^{[i]}([\mathbf{x}_j]),\ i = 1, ..., k-1$, permet de réduire le pessimisme dans (II.13). Pour cela, on écrit au voisinage de $\hat{\mathbf{x}}_j \in [\mathbf{x}_j]$

$$\mathbf{f}^{[i]}(\mathbf{x}_j) = \mathbf{f}^{[i]}(\hat{\mathbf{x}}_j) + \mathbf{J}(\mathbf{f}^{[i]}; \mathbf{x}_j, \hat{\mathbf{x}}_j)(\mathbf{x}_j - \hat{\mathbf{x}}_j), \tag{II.14}$$

où $\mathbf{J}(\mathbf{f}^{[i]}; \mathbf{x}_j, \hat{\mathbf{x}}_j)$ représente le Jacobien de $\mathbf{f}^{[i]}$ dont les éléments sont évalués en certains points mal connus $\mathbf{x}_j + \theta_{il}(\hat{\mathbf{x}}_j - \mathbf{x}_j)$ où $\theta_{il} \in [0, 1]\ l = 1, ..., n$. Donc, à partir des équations (II.12) et (II.14) on obtient,

$$\mathbf{x}_{j+1} = \hat{\mathbf{x}}_j + \sum_{i=1}^{k-1} h_j^i \mathbf{f}^{[i]}(\hat{\mathbf{x}}_j) + h_j^k \mathbf{f}^{[k]}(\mathbf{x}; t_j, t_{j+1}) + \{\mathbf{I} + \sum_{i=1}^{k-1} h_j^i \mathbf{J}(\mathbf{f}^{[i]}; \mathbf{x}_j, \hat{\mathbf{x}}_j)\}(\mathbf{x}_j - \hat{\mathbf{x}}_j). \tag{II.15}$$

C'est sur cette formule (II.15) que s'appuient les différentes méthodes d'intégration numérique garantie développées par Moore [68], Eijgenraam [24], Lohner [59], et Rihm [84] pour élaborer l'*Algorithme* II de l'étape de correction. En effet, l'objectif principal de ces méthodes est de réduire la propagation du pessimisme dû à l'effet d'enveloppement qui, dans certains cas, conduit à des encadrements de tailles très importantes inexploitables en pratique.

II.3.2.2 Effet d'enveloppement

L'effet d'enveloppement représente une source supplémentaire de pessimisme, qui apparaît souvent dans les algorithmes itératifs de calcul intervalle vectoriel. Ce phénomène

est bien mis en évidence dans l'exemple ci-dessous donné par Moore [68] et repris par Nedialkov dans sa thèse [69].

Exemple : Soit le système d'équations différentielles de deux dimensions suivant :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -x_1 \end{cases} \tag{II.16}$$

dont la solution est calculée analytiquement et son expression en fonction de conditions initiales est $\mathbf{x}(t) = \mathbf{A}(t)\mathbf{x}_0$, où

$$\mathbf{A}(t) = \begin{pmatrix} \cos(t) & \sin(t) \\ -\sin(t) & \cos(t) \end{pmatrix}$$

et $\mathbf{x}_0 \in [\mathbf{x}_0]$. Selon cette expression, on voit bien que l'évolution du pavé de conditions initiales $[\mathbf{x}_0]$ est régi par une transformation ponctuelle du type rotation dans le plan par rapport à l'origine, où $\mathbf{A}(t)$ représente la matrice de rotation. Donc, dans ce cas, les ensembles définis par $\{\mathbf{A}(t)\mathbf{x}_0 \mid \mathbf{x}_0 \in [\mathbf{x}_0], t > t_0\}$ sont équi-surfaces. Par ailleurs, comme

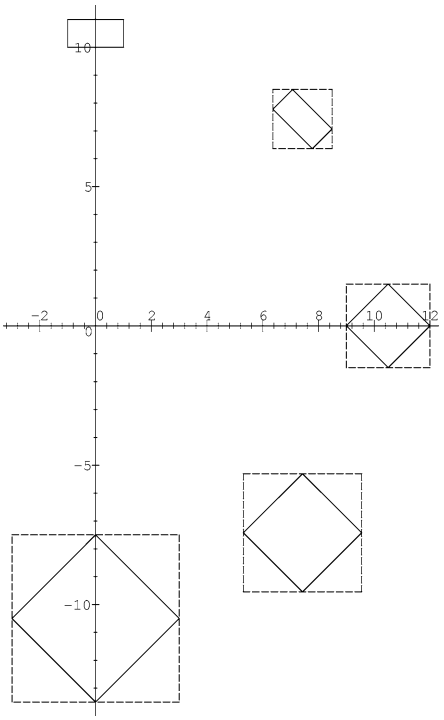


Fig. II.2 *: L'effet d'enveloppement*

le montre la figure II.2, si on calcul d'une manière itérative en utilisant l'arithmétique

d'intervalles les ensembles solutions, par exemple aux instants $t = \frac{\pi}{4}n$, où $n = 1, ..., 4$ pour $[\mathbf{x}_0] = [-1, 1] \times [10, 11]$, on introduit du pessimisme à chaque itération. Cela est dû à la sur-approximation des vrais ensembles solutions qui sont des parallélépipèdes par des pavés alignés avec les axes du repère original. Ainsi, l'accumulation des erreurs liées à cette sur-approximation mène à l'explosion de la taille de ces pavés qui provoque l'arrêt prématuré du processus du calcul.

II.3.2.3 Méthode directe

L'extension naturelle de l'expression (II.15) au cas intervalle, représente la méthode de correction directe des méthodes d'intégration numérique garantie. En effet, si on note par,

$$\begin{array}{l} \mathbf{J}(\mathbf{f}^{[i]}; [\mathbf{x}_j]) = \left\{ \frac{\partial \mathbf{f}^{[i]}(\mathbf{x})}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\mathbf{x}_j} \mid \mathbf{x}_j \in [\mathbf{x}_j] \right\}, \\ [\mathbf{S}_j] = \mathbf{I} + \sum_{i=1}^{k-1} h_j^i \mathbf{J}(\mathbf{f}^{[i]}; [\mathbf{x}_j]), \text{ et } \quad [\mathbf{z}_{j+1}] = h_j^k \mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j]) \end{array} \tag{II.17}$$

où les i^{ime} Jacobiens $\mathbf{J}(\mathbf{f}^{[i]}; [\mathbf{x}_j])$ sont soit calculés en différenciant les coefficients $\mathbf{f}^{[i]}([\mathbf{x}_j])$ du développement en série de Taylor [69], ou en calculant les coefficients de Taylor de l'équation variationnelle associé [69].

Ainsi en utilisant les notations données par (II.17), la version intervalles de l'expression (II.15) est :

$$[\mathbf{x}_{j+1}] = \hat{\mathbf{x}}_j + \sum_{i=1}^{k-1} h_j^i \mathbf{f}^{[i]}(\hat{\mathbf{x}}_j) + [\mathbf{z}_{j+1}] + [\mathbf{S}_j]([\mathbf{x}_j] - \hat{\mathbf{x}}_j). \tag{II.18}$$

De plus si on pose

$$[\mathbf{v}_{j+1}] = \hat{\mathbf{x}}_j + \sum_{i=1}^{k-1} h_j^i \mathbf{f}^{[i]}(\hat{\mathbf{x}}_j) + [\mathbf{z}_{j+1}]. \tag{II.19}$$

Alors l'équation (II.18) devient :

$$[\mathbf{x}_{j+1}] = [\mathbf{v}_{j+1}] + [\mathbf{S}_j]([\mathbf{x}_j] - \hat{\mathbf{x}}_j). \tag{II.20}$$

L'algorithme ci-dessous regroupe et organise les différents calculs nécessaires pour réaliser un pas d'intégration de cette méthode directe. Cet algorithme permet aussi de préparer les conditions initiales pour exécuter *l'Algorithme I* au prochain pas d'intégration.

Algorithme II :(Méthode directe)

1. Entrées : $[\tilde{\mathbf{x}}_j], h_j, [\mathbf{x}_j], \hat{\mathbf{x}}_j$;

2. $[\mathbf{z}_{j+1}] := h_j^k \mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j])$;
3. $[\mathbf{v}_{j+1}] := \hat{\mathbf{x}}_j + \sum_{i=1}^{k-1} h_j^i \mathbf{f}^{[i]}(\hat{\mathbf{x}}_j) + [\mathbf{z}_{j+1}]$;
4. $[\mathbf{S}_j] := \mathbf{I} + \sum_{i=1}^{k-1} h_j^i \mathbf{J}(\mathbf{f}^{[i]}; [\mathbf{x}_j])$;
5. $[\mathbf{x}_{j+1}] := [\mathbf{v}_{j+1}] + [\mathbf{S}_j]([\mathbf{x}_j] - \hat{\mathbf{x}}_j)$;
6. $\hat{\mathbf{x}}_{j+1} := m([\mathbf{v}_{j+1}])$ (le pavé $[\mathbf{S}_j]([\mathbf{x}_j] - \hat{\mathbf{x}}_j)$ est symétrique) ;
7. Sorties : $[\mathbf{x}_{j+1}], \hat{\mathbf{x}}_{j+1}$.

Dans certains cas, cette méthode directe peut calculer des pavés solutions de taille relativement raisonnable. Mais en général, le pavé $[\mathbf{S}_j]([\mathbf{x}_j] - \hat{\mathbf{x}}_j)$ représente une sur-approximation significative de l'ensemble,

$$\big\{\mathbf{S}_j(\mathbf{x}_j - \hat{\mathbf{x}}_j) \mid \mathbf{S}_j \in [\mathbf{S}_j],\ \mathbf{x}_j \in [\mathbf{x}_j]\big\}.$$

Par ailleurs, la propagation du pessimisme lié à cette sur-approximation (effet d'enveloppement) limite considérablement son utilisation. D'autres approches ont été développées pour mieux maîtriser cette propagation du pessimisme (Lohner [59], Rihm [84]) qui utilisent un pré-conditionnement matriciel.

II.3.2.4 La méthode de Lohner

Le principe fondamental de la méthode de Lohner consiste à réaliser, à chaque pas d'intégration, un changement de coordonnées orthogonales du système (II.1) pour mieux réduire la sur-approximation du vrai ensemble solution [59, 60]. L'idée exploitée est de faire propager dans le processus d'intégration des parallélépipèdes, qui encadrent l'ensemble solution à chaque pas d'intégration, au lieu des pavés alignés aux axes du système original. Il s'agit donc de définir, à chaque pas d'intégration, une matrice non singulière $\mathbf{A}_{j+1}$ et un vecteur intervalles $[\mathbf{r}_{j+1}]$ tel que :

$$\mathbf{x}(t_{j+1}; t_0, [\mathbf{x}_0]) \subseteq \{\hat{\mathbf{x}}_{j+1} + \mathbf{A}_{j+1}\mathbf{r}_{j+1} |\ \mathbf{r}_{j+1} \in [\mathbf{r}_{j+1}]\}. \tag{II.21}$$

Avant de présenter l'algorithme d'intégration de Lohner, nous allons montrer, à partir des deux premières itérations de (II.15), comment on peut caractériser les ensembles solutions par des parallélépipèdes. Pour cela, nous gardons les mêmes définitions de $[\mathbf{S}_j]$, $[\mathbf{z}_{j+1}]$, $[\mathbf{v}_{j+1}]$ et de $\hat{\mathbf{x}}_{j+1} = m([\mathbf{v}_{j+1}])$.

Soient :

$$\mathbf{A}_0 = \mathbf{I},\ \hat{\mathbf{x}}_0 = m([\mathbf{x}_0])\ \text{ et }\ [\mathbf{r}_0] = [\mathbf{x}_0] - \hat{\mathbf{x}}_0 \tag{II.22}$$

Ainsi, compte tenu de ces définitions, on peut écrire la première itération de (II.15) comme ci-dessous,

$$\begin{aligned}
\mathbf{x}_1 &= \hat{\mathbf{x}}_1 + \mathbf{S}_0(\mathbf{x}_0 - \hat{\mathbf{x}}_0) + \mathbf{z}_1 - \hat{\mathbf{z}}_1 \\
&\in \hat{\mathbf{x}}_1 + ([\mathbf{S}_0]\mathbf{A}_0)[\mathbf{r}_0] + [\mathbf{z}_1] - \hat{\mathbf{z}}_1 \\
&\equiv [\mathbf{x}_1],\ \text{autrement} \\
\mathbf{x}_1 &= \hat{\mathbf{x}}_1 + \mathbf{S}_0(\mathbf{x}_0 - \hat{\mathbf{x}}_0) + \mathbf{z}_1 - \hat{\mathbf{z}}_1 \\
&= \hat{\mathbf{x}}_1 + \mathbf{A}_1((\mathbf{A}_1^{-1}\mathbf{S}_0\mathbf{A}_0)\mathbf{r}_0 + \mathbf{A}_1^{-1}(\mathbf{z}_1 - \hat{\mathbf{z}}_1)) \\
&\in \{\hat{\mathbf{x}}_1 + \mathbf{A}_1\mathbf{r}_1 \mid \mathbf{r}_1 \in [\mathbf{r}_1]\} \subseteq [\mathbf{x}_1],
\end{aligned}$$

où $\mathbf{A}_1 \in \mathbb{R}^{n\times n}$ est une matrice non singulière et

$$[\mathbf{r}_1] = (\mathbf{A}_1^{-1}([\mathbf{S}_0]\mathbf{A}_0))[\mathbf{r}_0] + \mathbf{A}_1^{-1}([\mathbf{z}_1] - \hat{\mathbf{z}}_1).$$

De même, pour la deuxième itération on a

$$\begin{aligned}
\mathbf{x}_2 &= \hat{\mathbf{x}}_2 + \mathbf{S}_1(\mathbf{x}_1 - \hat{\mathbf{x}}_1) + \mathbf{z}_2 - \hat{\mathbf{z}}_2 \\
&\in \hat{\mathbf{x}}_2 + ([\mathbf{S}_1]\mathbf{A}_1)[\mathbf{r}_1] + [\mathbf{z}_2] - \hat{\mathbf{z}}_2 \\
&\equiv [\mathbf{x}_2],\ \text{autrement} \\
\mathbf{x}_2 &= \hat{\mathbf{x}}_2 + \mathbf{S}_1(\mathbf{x}_1 - \hat{\mathbf{x}}_1) + \mathbf{z}_2 - \hat{\mathbf{z}}_2 \\
&= \hat{\mathbf{x}}_2 + \mathbf{A}_2((\mathbf{A}_2^{-1}\mathbf{S}_1\mathbf{A}_1)\mathbf{r}_1 + \mathbf{A}_2^{-1}(\mathbf{z}_2 - \hat{\mathbf{z}}_2)) \\
&\in \{\hat{\mathbf{x}}_2 + \mathbf{A}_2\mathbf{r}_2 \mid \mathbf{r}_2 \in [\mathbf{r}_2]\} \subseteq [\mathbf{x}_2],
\end{aligned}$$

où $\mathbf{A}_2 \in \mathbb{R}^{n\times n}$ est aussi une matrice non singulière et

$$[\mathbf{r}_2] = (\mathbf{A}_2^{-1}([\mathbf{S}_1]\mathbf{A}_1))[\mathbf{r}_1] + \mathbf{A}_2^{-1}([\mathbf{z}_2] - \hat{\mathbf{z}}_2).$$

On s'aperçoit qu'à chaque pas d'intégration, l'ensemble solution est encadré par un parallélépipède défini par une matrice $\mathbf{A}_j$ et un vecteur intervalles $[\mathbf{r}_j]$. En effet, c'est sur cette représentation parallélépipédique que s'appuie l'algorithme de Lohner pour réduire le pessimisme. Cependant, le choix des matrices $\mathbf{A}_j$ $(j \geq 1)$ joue un rôle crucial pour faire réussir la méthode de Lohner.

Algorithme II : (Méthode de Lohner)

1. Entrées : $[\tilde{\mathbf{x}}_j], h_j, [\mathbf{x}_j], \hat{\mathbf{x}}_j, \mathbf{A}_j, [\mathbf{r}_j]$;
2. $[\mathbf{z}_{j+1}] := h_j^k \mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j])$;

3. $\hat{\mathbf{z}}_{j+} := m([\mathbf{z}_{j+1}])$;
4. $\hat{\mathbf{x}}_{j+1} := \hat{\mathbf{x}}_j + \sum_{i=1}^{k-1} h_j^i \mathbf{f}^{[i]}(\hat{\mathbf{x}}_j) + \hat{\mathbf{z}}_{j+1}$;
5. $[\mathbf{S}_j] := I + \sum_{i=1}^{k-1} h_j^i J(\mathbf{f}^{[i]}; [\mathbf{x}_j])$;
6. Faire un choix convenable de la matrice $\mathbf{A}_{j+1}$;
7. $[\mathbf{x}_{j+1}] := \hat{\mathbf{x}}_{j+1} + ([\mathbf{S}_j]\mathbf{A}_j)[\mathbf{r}_j] + [\mathbf{z}_{j+1}] - \hat{\mathbf{z}}_{j+1}$;
8. $[\mathbf{r}_{j+1}] = (\mathbf{A}_{j+1}^{-1}([\mathbf{S}_j]\mathbf{A}_j))[\mathbf{r}_j] + \mathbf{A}_{j+1}^{-1}([\mathbf{z}_{j+1}] - \hat{\mathbf{z}}_{j+1})$;
9. Sorties : $[\mathbf{x}_{j+1}], \hat{\mathbf{x}}_{j+1}, \mathbf{A}_{j+1}, [\mathbf{r}_{j+1}]$.

Il existe dans la littérature, deux méthodes pour le choix ou le calcul des matrices $\mathbf{A}_{j+1}$ $(j \geq 0)$: la première est dite méthode parallélépipédique et la seconde est dite méthode par factorisation **QR**. Nous présentons, dans la suite, ces deux méthodes.

II.3.2.4.1 Méthode parallélépipédique

En analysant la ligne 8 de l'algorithme de Lohner, nous constatons que la propagation du pessimisme est due essentiellement au terme $\mathbf{A}_{j+1}^{-1}([\mathbf{S}_j]\mathbf{A}_j)[\mathbf{r}_j]$. Pour mieux maîtriser ce phénomène, on doit trouver une matrice non singulière $\mathbf{A}_{j+1}$ tel que $\mathbf{A}_{j+1}^{-1}([\mathbf{S}_j]\mathbf{A}_j) \approx \mathbf{I}$. Pour cela, la méthode parallélépipédique propose de poser $\mathbf{A}_{j+1} = m([\mathbf{S}_j]\mathbf{A}_j)$. En général, cette manière de calculer $\mathbf{A}_{j+1}$ est insuffisante pour garantir le bon fonctionnement de l'algorithme de Lohner. En effet, si on pose $\hat{\mathbf{S}}_j = m([\mathbf{S}_j])$ et $[\mathbf{S}_j] = \hat{\mathbf{S}}_j + [\mathbf{E}_j]$ on obtient :

$$\begin{aligned} \mathbf{A}_{j+1} &= \hat{\mathbf{S}}_j\mathbf{A}_j \quad \text{et} \\ \mathbf{A}_{j+1}^{-1}([\mathbf{S}_j]\mathbf{A}_j) &= \mathbf{A}_j^{-1}\hat{\mathbf{S}}_j^{-1}(\hat{\mathbf{S}}_j\mathbf{A}_j + [\mathbf{E}_j]\mathbf{A}_j) \\ &= \mathbf{I} + \mathbf{A}_j^{-1}\hat{\mathbf{S}}_j^{-1}[\mathbf{E}_j]\mathbf{A}_j. \end{aligned}$$

De plus, on peut dire que,

$$||\mathbf{A}_j^{-1}\hat{\mathbf{S}}_j^{-1}[\mathbf{E}_j]\mathbf{A}_j|| \leq \text{cond}(\mathbf{A}_j)||\hat{\mathbf{S}}_j^{-1}[\mathbf{E}_j]||.$$

Dès lors, pour assurer que $\mathbf{A}_{j+1}^{-1}([\mathbf{S}_j]\mathbf{A}_j) \approx \mathbf{I}$, il faudrait que (i) $||\hat{\mathbf{S}}_j^{-1}[\mathbf{E}_j]||$ soit négligeable (de valeur très faible) et (ii) le nombre de conditionnement $\text{cond}(\mathbf{A}_j)$ ne doit pas être très important. Cependant, ces deux conditions et la non-singularité de la matrice $\mathbf{A}_{j+1}$ ne sont pas garanties par cette façon de calculer $\mathbf{A}_{j+1}$.

II.3.2.4.2 Factorisation QR

Afin d'éviter les inconvénients ou les insuffisances de la méthode parallélépipèdique, la méthode par factorisation **QR** propose un meilleur choix des matrices $\mathbf{A}_{j+1},\ j \geq 0$, qui s'appuie sur certaines propriétés du calcul matriciel.

Comme nous l'avons évoqué dans la sous-section précédente, la propagation du pessimisme dans l'algorithme de Lohner est due essentiellement aux encadrements $[\mathbf{r}_{j+1}]$ calculées dans la ligne 8. En effet, l'objectif est de calculer à chaque pas d'intégration un encadrement le plus étroit possible de l'ensemble

$$\{(\mathbf{A}_{j+1}^{-1}\mathbf{S}_j\mathbf{A}_j)\mathbf{r}_j + \mathbf{A}_{j+1}^{-1}(\mathbf{z}_{j+1} - \hat{\mathbf{z}}_{j+1}) \mid \mathbf{S}_j \in [\mathbf{S}_j],\ \mathbf{r}_j \in [\mathbf{r}_j],\ \mathbf{z}_{j+1} \in [\mathbf{z}_{j+}]\}. \tag{II.23}$$

Pour des raisons de simplicité, nous considérons tout d'abord l'ensemble

$$\{\mathbf{A}_{j+1}^{-1}(\mathbf{z}_{j+1} - \hat{\mathbf{z}}_{j+1}) \mid \mathbf{z}_{j+1} \in [\mathbf{z}_{j+}]\}. \tag{II.24}$$

En utilisant les propriétés des normes matricielles, on obtient l'inégalité suivante :

$$\left|\left|w(\mathbf{A}_{j+1}^{-1}([\mathbf{z}_{j+1}] - \hat{\mathbf{z}}_{j+1}))\right|\right| = \left|\left||\mathbf{A}_{j+1}^{-1}|w([\mathbf{z}_{j+1}])\right|\right| \leq \left|\left|\mathbf{A}_{j+1}^{-1}\right|\right|.\left|\left|w([\mathbf{z}_{j+1}])\right|\right| \tag{II.25}$$

Sachant que dans cette approche la matrice $\mathbf{A}_{j+1}^{-1}$ est orthogonale, alors $||\mathbf{A}_{j+1}^{-1}|| \leq \sqrt{n}$. Par ailleurs, en réduisant la largeur du pas d'intégration h_j ou en élevant l'ordre du développement en série de Taylor, on peut facilement obtenir une taille $w([\mathbf{z}_{j+1}])$ très petite du pavé $[\mathbf{z}_{j+1}]$. Par conséquent, nous pouvons encadrer l'ensemble (II.24) par le vecteur intervalles $\mathbf{A}_{j+1}^{-1}([\mathbf{z}_{j+1}] - \hat{\mathbf{z}}_{j+1})$ sans introduire de pessimisme.

Ainsi, l'ensemble qui nécessite réellement le plus de précautions lors de l'encadrement, pour éviter d'introduire excessivement du pessimisme est

$$\{(\mathbf{A}_{j+1}^{-1}\mathbf{S}_j\mathbf{A}_j)\mathbf{r}_j \mid \mathbf{S}_j \in [\mathbf{S}_j],\ \mathbf{r}_j \in [\mathbf{r}_j]\} \tag{II.26}$$

Soient $\tilde{\mathbf{A}}_{j+1} \in [\mathbf{S}_j]\mathbf{A}_j$ et $\hat{\mathbf{A}}_{j+1} = \tilde{\mathbf{A}}_{j+1}\mathbf{P}_{j+1}$, où $\mathbf{P}_{j+1}$ est une matrice de permutation qui sert à arranger les colonnes de la matrice $\hat{\mathbf{A}}_{j+1}$ de sorte à améliorer la qualité de l'encadrement de l'ensemble (II.26). Par ailleurs, nous effectuons une factorisation **QR** de la matrice $\hat{\mathbf{A}}_{j+1}$,

$$\hat{\mathbf{A}}_{j+1} = \mathbf{Q}_{j+1}\mathbf{R}_{j+1}$$

où $\mathbf{Q}_{j+1}$ est une matrice orthogonale et $\mathbf{R}_{j+1}$ une matrice triangulaire supérieure. Ainsi, la matrice $\mathbf{A}_{j+1}$ de l'algorithme de Lohner est égale à $\mathbf{Q}_{j+1}$. Dans la suite, nous allons montrer l'intérêt de ce choix.

Si la taille $w([\mathbf{S}_j])$ est petite, alors l'approximation ensembliste ci-dessous est valable.

$$\{(\mathbf{S}_j\mathbf{A}_j)\mathbf{r}_j \mid \mathbf{S}_j \in [\mathbf{S}_j],\ \mathbf{r}_j \in [\mathbf{r}_j]\} \approx \{\tilde{\mathbf{A}}_{j+1})\mathbf{r}_j \mid\ \mathbf{r}_j \in [\mathbf{r}_j]\}. \tag{II.27}$$

Ainsi, de (II.26) et (II.27) on peut montrer l'inclusion suivante,

$$\begin{aligned}
\{(\mathbf{A}_{j+1}^{-1}\mathbf{S}_j\mathbf{A}_j)\mathbf{r}_j \mid \mathbf{S}_j &\in [\mathbf{S}_j],\ \mathbf{r}_j \in [\mathbf{r}_j]\} \\
&= \{(\mathbf{Q}_{j+1}^{-1}(\mathbf{S}_j\mathbf{A}_j))\mathbf{r}_j \mid \mathbf{S}_j \in [\mathbf{S}_j],\ \mathbf{r}_j \in [\mathbf{r}_j]\} \\
&\approx \{(\mathbf{Q}_{j+1}^{-1}\tilde{\mathbf{A}}_{j+1})\mathbf{r}_j \mid \mathbf{r}_j \in [\mathbf{r}_j]\} \\
&\subseteq (\mathbf{Q}_{j+1}^{-1}\tilde{\mathbf{A}}_{j+1})[\mathbf{r}_j].
\end{aligned} \tag{II.28}$$

La matrice $\mathbf{Q}_{j+1}$ forme une nouvelle base orthogonale des coordonnées du système (II.1), dont l'axe qui correspond à sa première colonne est parallèle aux côtés du parallélépipède $\{\tilde{\mathbf{A}}_{j+1}\mathbf{r}_j \mid \mathbf{r}_j \in [\mathbf{r}_j]\}$ qui sont parallèles à la première colonne de $\hat{\mathbf{A}}_{j+1}$. Par conséquent, nous obtiendrons un meilleur encadrement (avec beaucoup moins de pessimisme) dans la base de coordonnées induite par $\mathbf{Q}_{j+1}$ que celui obtenu dans la base originale de coordonnées du système. De plus, la précision d'encadrement est d'autant meilleure si la première colonne de $\mathbf{Q}_{j+1}$ est parallèle au côté le plus large du parallélépipède $\{\tilde{\mathbf{A}}_{j+1}\mathbf{r}_j \mid \mathbf{r}_j \in [\mathbf{r}_j]\}$. C'est pour cette raison que nous devrons réarranger les colonnes de la matrice, $\tilde{\mathbf{A}}_{j+1}$ en la multipliant par la matrice de permutation $\mathbf{P}_{j+1}$, de telle sorte que la première colonne de $\hat{\mathbf{A}}_{j+1}$ corresponde au côté le plus large de $\{\tilde{\mathbf{A}}_{j+1}\mathbf{r}_j \mid \mathbf{r}_j \in [\mathbf{r}_j]\}$, la seconde colonne au deuxième côté le plus large et ainsi de suite.

Pour illustrer le principe de cette méthode de factorisation **QR** nous considérons l'exemple de Nedialkov [69] ci-dessous.

Exemple : Soit

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 2 & 1 \end{pmatrix} \quad \text{et} \quad [\mathbf{r}] = \begin{pmatrix} [1,2] \\ [1,4] \end{pmatrix}. \tag{II.29}$$

La factorisation **QR** de la matrice **A** est

$$\mathbf{A} = \Big(-\frac{1}{\sqrt{5}}\begin{pmatrix} 1 & 1 \\ 2 & -1 \end{pmatrix}\Big)\Big(-\frac{1}{\sqrt{5}}\begin{pmatrix} 5 & 3 \\ 0 & 1 \end{pmatrix}\Big) \equiv \mathbf{QR} \tag{II.30}$$

Considérons tout d'abord l'ensemble

$$\{\mathbf{Ar} \mid \mathbf{r} \in [\mathbf{r}]\}. \tag{II.31}$$

La figure II.3(a) montre la représentation graphique du pavé initial $[\mathbf{r}]$ dans le plan (x, y). Une application linéaire, définie par la matrice $\mathbf{A}$, le transforme en un parallélépipède tracé dans la figure II.3(b). Sur cette figure, le carré en ligne discontinue représente une sur-approximation de ce parallélépipède obtenu par un simple calcul intervalles $\mathbf{A}[\mathbf{r}]$. En revanche, un changement orthogonal de coordonnées, engendré par la matrice $\mathbf{Q}$, permet d'améliorer nettement la précision de l'encadrement du parallélépipède $\{\mathbf{Ar} \mid \mathbf{r} \in [\mathbf{r}]\}$ dans cette nouvelle base de coordonnées $(\mathbf{Q}x, \mathbf{Q}y)$ comme le montre la figure II.3(c).

Maintenant, considérons l'ensemble

$$\{\mathbf{Q}^{-1}\mathbf{Ar} \mid \mathbf{r} \in [\mathbf{r}]\}. \tag{II.32}$$

La matrice $\mathbf{Q}^{-1}$, calculée à partir de (II.30), transforme le parallélépipède (II.31) en un autre parallélépipède (II.32) dont le côté court est parallèle à l'axe original x. Son encadrement est $(\mathbf{Q}^{-1}\mathbf{A})[\mathbf{r}]$ représenté sur la figure II.3(d).

Par ailleurs, on peut obtenir un meilleur encadrement en réarrangeant les colonnes de la matrice $\mathbf{A}$. La factorisation $\mathbf{QR}$ de la nouvelle matrice $\hat{\mathbf{A}}$ est

$$\hat{\mathbf{A}} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} = \Big(-\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\Big)\Big(-\frac{1}{\sqrt{2}}\begin{pmatrix} 2 & 3 \\ 0 & -1 \end{pmatrix}\Big) \equiv \hat{\mathbf{Q}}\hat{\mathbf{R}}. \tag{II.33}$$

La figure II.3(e) montre que l'encadrement de l'ensemble (II.31) dans le plan $(\hat{\mathbf{Q}}^{-1}x, \hat{\mathbf{Q}}^{-1}y)$ est plus précis que celui obtenu dans le plan $(\mathbf{Q}^{-1}x, \mathbf{Q}^{-1}y)$. De même, le pavé $(\hat{\mathbf{Q}}^{-1}\mathbf{A})[\mathbf{r}]$ est de taille inférieure par rapport à celle du pavé $(\mathbf{Q}^{-1}\mathbf{A})[\mathbf{r}]$ comme l'illustre la figure II.3 (f). En effet, le côté le plus large du parallélépipède, $\{\mathbf{Ar} \mid \mathbf{r} \in [\mathbf{r}]\}$ transformé par $\mathbf{Q}^{-1}$, est parallèle à l'axe original x.

II.3.2.5 La méthode de Rihm

La méthode de Rihm [84], connue aussi sous le nom de la méthode de la valeur moyenne étendue, s'appuie également sur la formule (II.20). En effet, pour réduire la propagation du pessimisme induit par le terme,

$$[\mathbf{s}_{j+1}] = [\mathbf{S}_j]([\mathbf{x}_j] - \hat{\mathbf{x}}_j),$$

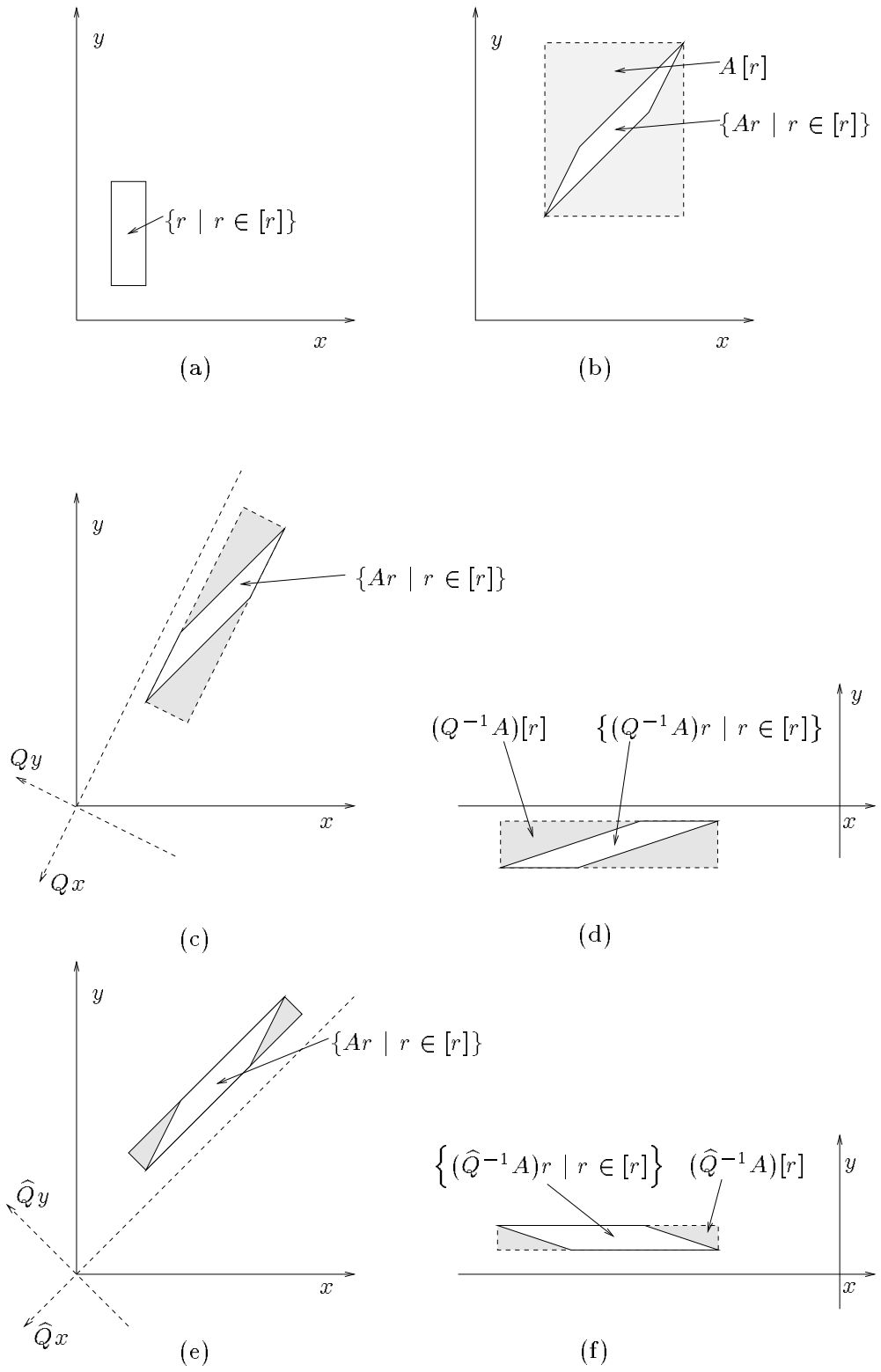


Fig. II.3 *: Factorisation* ***QR*** *[69].*

elle consiste à introduire, au fur et à mesure de l'intégration, des matrices régulières $\mathbf{A}_j$ de pré-conditionnement de l'espace de calcul de sorte à obtenir des encadrements plus précis des ensembles solutions aux instants t_j. Pour illustrer le principe de cette méthode, nous considérons d'abord le premier pas d'intégration où on a

$$[\mathbf{s}_1] = [\mathbf{S}_0]([\mathbf{x}_0] - \hat{\mathbf{x}}_0).$$

Soit le pavé $[\mathbf{p}_1]$ défini comme suit :

$$[\mathbf{p}_1] = (\mathbf{A}_1^{-1}[\mathbf{S}_0])([\mathbf{x}_0] - \hat{\mathbf{x}}_0).$$

Si la matrice $\mathbf{A}_1$ est calculée de telle sorte que $\mathbf{A_1} \in [\mathbf{S}_0]$, alors le produit $\mathbf{A}_1^{-1}[\mathbf{S}_0]$ inclut une matrice identité, et par conséquent, le pavé $[\mathbf{p}_1]$ n'introduit pas beaucoup de pessimisme. De plus, il est clair que le parallélépipède défini par le couple $(\mathbf{A}_1, [\mathbf{p}_1]) = \{\mathbf{A}_1\mathbf{p}_1 \mid \mathbf{p}_1 \in [\mathbf{p}_1]\}$ contient l'ensemble $\{\mathbf{S}_0\mathbf{x}_0 \mid \mathbf{S}_0 \in [\mathbf{S}_0], \mathbf{x}_0 \in [\mathbf{x}_0]\}$ qui est évidemment moins volumineux que le vecteur intervalles $[\mathbf{s}_1]$. En s'appuyant sur cette remarque, Rihm propose de calculer, à chaque pas d'intégration, le pavé $[\mathbf{s}_{j+1}]$ en fonction de $\mathbf{A}_j$ et $[\mathbf{p}_j]$ au lieu de le calculer en fonction de $[\mathbf{s}_j]$.

Par exemple, pour la première itération nous avons comme pavé solution l'expression suivante,

$$[\mathbf{x}_1] = [\mathbf{v}_1] + [\mathbf{S}_0]([\mathbf{x}_0] - \hat{\mathbf{x}}_0).$$

L'objectif est d'éviter de faire propager le vecteur intervalle $[\mathbf{s}_1]$ aux futurs pas d'intégration. Pour cela, partant de l'équation ci-dessous,

$$[\mathbf{s}_2] = [\mathbf{S}_1]([\mathbf{x}_1] - \hat{\mathbf{x}}_1) = [\mathbf{S}_1]([\mathbf{s}_1] + [\mathbf{v}_1] - \hat{\mathbf{x}}_1)$$

et en remplaçant $[\mathbf{s}_1]$ par la paire $(\mathbf{A}_1, [\mathbf{p}_1])$, on peut redéfinir $[\mathbf{s}_2]$ par :

$$[\mathbf{s}_2] = ([\mathbf{S}_1]\mathbf{A}_1)[\mathbf{p}_1] + [\mathbf{S}_1]([\mathbf{v}_1] - \hat{\mathbf{x}}_1).$$

Ainsi, l'encadrement de l'ensemble solution à la deuxième itération est donné par

$$[\mathbf{x}_2] = [\mathbf{v}_2] + [\mathbf{s}_2].$$

Finalement, pour préparer la prochaine itération, on pose

$$[\mathbf{p}_2] = (\mathbf{A}_2^{-1}([\mathbf{S}_2]\mathbf{A}_1))[\mathbf{p}_1] + (\mathbf{A}_2^{-1}[\mathbf{S}_2])([\mathbf{v}_1] - \hat{\mathbf{x}}_1)$$

où $\mathbf{A}_2$ est une nouvelle matrice régulière incluse dans $[\mathbf{S}_2]$.

Ainsi, en appliquant ce même raisonnement pour les futures itérations, nous obtiendrons l'algorithme de la valeur moyenne étendue de Rihm [85] présenté ci-dessous.

Algorithme II : (Méthode de la valeur moyenne étendue)

1. Entrées : $[\tilde{\mathbf{x}}_j], h_j, [\mathbf{v}_j], [\mathbf{x}_j], \hat{\mathbf{x}}_j$;
2. $[\mathbf{z}_{j+1}] := h_j^k \mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j])$;
3. $[\mathbf{v}_{j+1}] := \hat{\mathbf{x}}_j + \sum_{i=1}^{k-1} h_j^i \mathbf{f}^{[i]}(\hat{\mathbf{x}}_j) + [\mathbf{z}_{j+1}]$;
4. $[\mathbf{S}_j] := I + \sum_{i=1}^{k-1} h_j^i J(\mathbf{f}^{[i]}; [\mathbf{x}_j])$;
5. $[\mathbf{s}_{j+1}] = ([\mathbf{S}_j]\mathbf{A}_j)[\mathbf{p}_j] + [\mathbf{S}_j]([\mathbf{v}_j] - \hat{\mathbf{x}}_j)$
6. $[\mathbf{p}_{j+1}] = (\mathbf{A}_{j+1}^{-1}([\mathbf{S}_j]\mathbf{A}_j))[\mathbf{p}_j] + (\mathbf{A}_{j+1}^{-1}[\mathbf{S}_j])([\mathbf{v}_j] - \hat{\mathbf{x}}_j)$
7. $[\mathbf{x}_{j+1}] := [\mathbf{v}_{j+1}] + [\mathbf{s}_{j+1}]$;
8. $\hat{\mathbf{x}}_{j+1} := m([\mathbf{v}_{j+1}])$ (Le pavé $[\mathbf{S}_j]([\mathbf{x}_j] - \hat{\mathbf{x}}_j)$ est symétrique) ;
9. Sorties : $[\mathbf{v}_{j+1}], [\mathbf{x}_{j+1}], \hat{\mathbf{x}}_{j+1}$.

Pour l'itération initiale, on fixe $[\mathbf{v}_0] = [\mathbf{x}_0]$, $[\mathbf{p}_0] = 0$ et la matrice $[\mathbf{A}_0] = \mathbf{I}$.

Comme dans le cas de la méthode de Lohner, le choix des matrices $\mathbf{A}_j$ joue un rôle déterminant pour la réussite de l'algorithme de Rihm. Mais, en pratique, on ne sait pas quelles sont les matrices optimales $\mathbf{A}_j$ qui doivent être utilisées pour mieux réduire la propagation du pessimisme. Cependant, à ce jour, le pré-conditionnement matriciel le plus utilisé est celui de la factorisation **QR** de Lohner présentée dans la section précédente. En effet, les matrices $\mathbf{A}_j$ choisies sont non-singulières et bien conditionnées.

Remarque : Rihm [84] affirme que, si on effectue m pas d'intégration d'une équation différentielle donnée par la méthode de Lohner et par celle de la valeur moyenne étendue dans les mêmes conditions, c'est-à-dire conditions initiales, pas d'intégration, ordre du développement de Taylor et matrices de pré-conditionnement $\mathbf{A}_j, \forall j = 0, ..., m$, alors on obtient les inclusions suivantes :

$$[\mathbf{x}_j]_M \subseteq [\mathbf{x}_j]_L \text{ et } m([\mathbf{x}_j]_M) = m([\mathbf{x}_j]_L) \text{ pour } j = 1, 2, ..., m. \tag{II.34}$$

où $[\mathbf{x}_j]_M$ et $[\mathbf{x}_j]_L$ représentent respectivement les encadrements de solutions obtenus par la méthode de la valeur moyenne étendue et ceux calculés par la méthode de Lohner.

II.3.2.6 Cas des EDO où la factorisation QR n'est pas préconisé

Dans le cas des systèmes coopératifs, que nous allons définir dans le chapitre suivant, la factorisation **QR** n'est pas préférable [84]. En effet, pour ce genre d'EDO il est intéressant de fixer les matrices $\mathbf{A}_j = \mathbf{I}$, $\forall j \geq 0$. Il s'agit donc d'utiliser simplement la méthode directe. L'exemple ci-dessous montre bien l'avantage de la méthode directe par rapport à la factorisation **QR** dans le cas des EDO coopératives.

Exemple : Soit l'équation différentielle coopérative,

$$\dot{\mathbf{x}} = \begin{pmatrix} -10 & 14 & 7 \\ 0.5 & -5 & 2 \\ 0 & 0 & 0 \end{pmatrix} \mathbf{x}, \tag{II.35}$$

où $\mathbf{x}_0 \in [\mathbf{x}_0] = [2,3] \times [-3,-2] \times [1.8,2]$.

La résolution de cette équation différentielle sur une période de $2s$ avec la méthode directe et celle de Lohner par factorisation **QR**, permet d'obtenir les courbes représentées sur les figures II.4 et II.5.

Les courbes bleues en ligne continue représentent les sur-approximations de toutes les solutions possibles de la première et la deuxième composantes du vecteur $\mathbf{x}(t)$ solution de II.35. Ces encadrements sont obtenus par la méthode directe. Les courbes rouges en ligne discontinue montrent les sur-approximations de ces mêmes composantes du vecteur $\mathbf{x}(t)$ obtenues par la méthode de Lohner avec factorisation **QR** . D'après ces courbes on voit bien que $[x_{1_{bleue}}](t) \subset [x_{1_{rouge}}](t)$ et $[x_{2_{bleue}}](t) \subset [x_{2_{rouge}}](t)$.

II.3.2.7 Schéma d'intégration implicite

En pratique, dans le contexte classique, c'est-à-dire ponctuel, de la résolution d'équations différentielles, les méthodes implicites sont généralement plus performantes que les méthodes explicites, en particulier pour les systèmes raides. En fait, elles permettent de travailler avec des pas d'intégration relativement larges. De plus, elles sont connues pour leur stabilité numérique ce qui naturellement pourrait limiter la propagation du pessimisme dans le cas intervalles. Ceci a été la motivation principale de Rihm [85] pour développer une nouvelle stratégie d'encadrement de solutions EDO qui repose sur un schéma implicite pour le développement en série de Taylor intervalles.

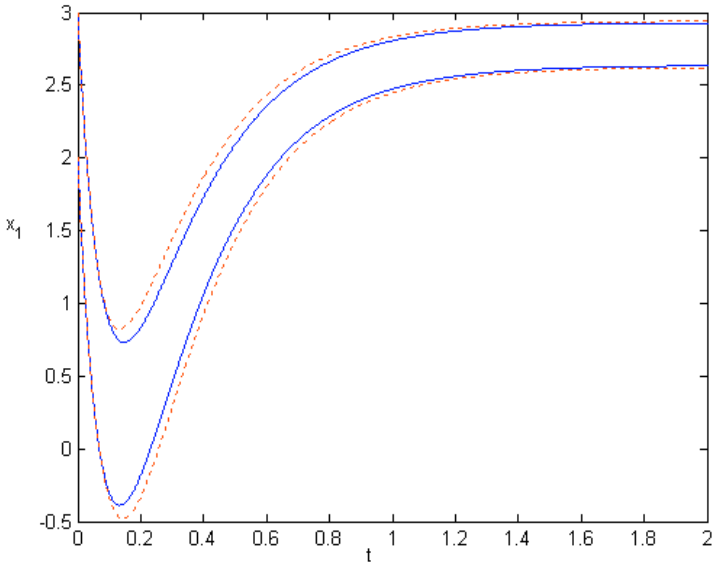


Fig. II.4 *: Comparaisons entre la méthode directe et la méthode de factorisation* ***QR*** *dans le cas des EDO coopératives, encadrement de la variable* $x_1(t)$.

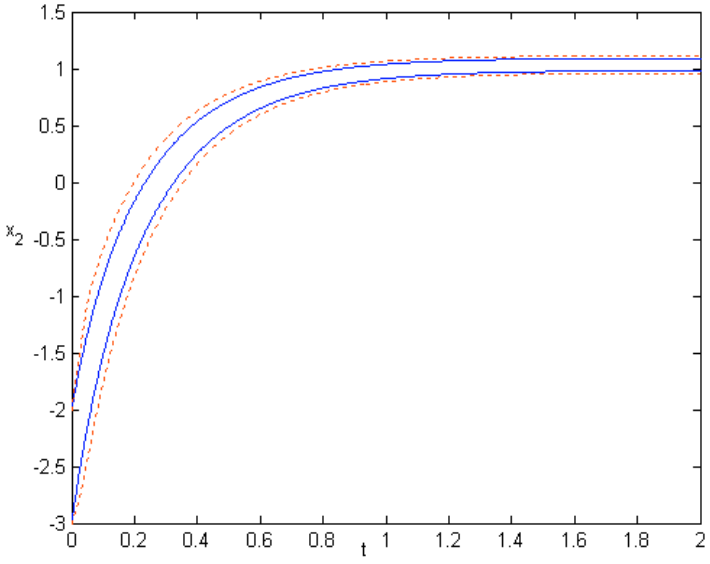


Fig. II.5 *: Comparaisons entre la méthode directe et la méthode de factorisation* ***QR*** *dans le cas des EDO coopératives, encadrement de la variable* $x_2(t)$.

Dans cette section, nous montrons le principe de cette méthode pour un pas d'intégration, par contre nous ne traitons pas le problème de la propagation du pessimisme. Supposons que nous disposons d'une solution *a priori* $[\tilde{\mathbf{x}}_j]$ qui englobe la vraie solution du (II.1) sur l'intervalle $[t_j, t_{j+1}]$. Alors il existe $\mathbf{x}_\xi \in [\tilde{\mathbf{x}}_j]$ tel que :

$$\mathbf{x}_j = \mathbf{x}_{j+1} + \sum_{i=1}^{k-1}(-h_j)^i \mathbf{f}^{[i]}(\mathbf{x}_{j+1}) + (-h_j)^k \mathbf{f}^{[k]}(\mathbf{x}_\xi), \tag{II.36}$$

où $h = t_{j+1} - t_j$.

Ainsi, on aperçoit clairement que la solution recherchée $\mathbf{x}_{j+1}$ apparaît implicitement dans II.36. De ce fait, $\mathbf{x}_{j+1}$ est la racine de la fonction vectorielle $\mathbf{g} : [\tilde{\mathbf{x}}_j] \to \mathbb{R}^n$ définie par,

$$\mathbf{g}(\mathbf{x}) = \mathbf{x} + \sum_{i=1}^{k-1}(-h_j)^i \mathbf{f}^{[i]}(\mathbf{x}) - \mathbf{x}_j + (-h_j)^k \mathbf{f}^{[k]}(\mathbf{x}_\xi). \tag{II.37}$$

Etant donné que $\mathbf{x}_\xi$ et $\mathbf{x}_j$ ne sont pas connues exactement, nous considérons alors la fonction d'inclusion suivante,

$$[\mathbf{g}](\mathbf{x}) = \mathbf{x} + \sum_{i=1}^{k-1}(-h_j)^i \mathbf{f}^{[i]}(\mathbf{x}) - [\mathbf{x}_j] + (-h_j)^k \mathbf{f}^{[k]}([\tilde{\mathbf{x}}_j]). \tag{II.38}$$

Pour résoudre ce problème du calcul de racines dans le contexte intervalles, nous pouvons utiliser une méthode de type Newton étendue aux intervalles [50] afin de déterminer un encadrement extérieur garanti $[\mathbf{x}_{j+1}]$ de l'ensemble $\{\mathbf{x} \in [\tilde{\mathbf{x}}_j] \mid 0 \in [\mathbf{g}](\mathbf{x})\}$, c'est-à-dire

$$[\mathbf{x}_{j+1}] \supseteq \{\mathbf{x} \in [\tilde{\mathbf{x}}_j] \mid 0 \in [\mathbf{g}](\mathbf{x})\}. \tag{II.39}$$

Le théorème ci-dessous [85] présente un algorithme pour déterminer $[\mathbf{x}_{j+1}]$ sous certaines conditions de régularité des matrices intervalles.

Soit la matrice intervalles régulière $[\tilde{\mathbf{S}}_j] = \mathbf{I} + \sum_{i=1}^{k-1}(-h_j)^i \mathbf{J}(\mathbf{f}^{[i]}; [\tilde{\mathbf{x}}_j])$ (c'est-à-dire que chaque matrice réelle $\mathbf{A} \in [\tilde{\mathbf{S}}_j]$ est régulière) et soit $[\tilde{\mathbf{S}}_j^{-1}]$ une matrice intervalles tel que $\mathbf{A}^{-1} \in [\tilde{\mathbf{S}}_j^{-1}]$ pour tout $\mathbf{A} \in [\tilde{\mathbf{S}}_j]$. Soit la suite $[\mathbf{y}_0], [\mathbf{y}_1], [\mathbf{y}_2], ...$ donnée par la formule récursive suivante :

Algorithme II : (Méthode implicite)

1. Entrées : $[\tilde{\mathbf{x}}_j]$, $[\mathbf{x}_j]$
2. $l = 0$
3. $[\mathbf{y}_0] = [\tilde{\mathbf{x}}_j]$

4. $[\mathbf{y}_1] = (m([\mathbf{y}_0]) - [\tilde{\mathbf{S}}_j^{-1}][\mathbf{g}](m([\mathbf{y}_0])) \cap [\mathbf{y}_0]$
5. **Tant que** $([\mathbf{y}_{l+1}] = [\mathbf{y}_l] \vee 0 \in [\mathbf{g}](m([\mathbf{y}_l]))$ **faire**
6. $[\mathbf{y}_{l+1}] = (m([\mathbf{y}_l]) - [\tilde{\mathbf{S}}_j^{-1}][\mathbf{g}](m([\mathbf{y}_l])) \cap [\mathbf{y}_l]$
7. **Fin**
8. Sortie : $[\mathbf{x}_{j+1}] = [\mathbf{y}_{l+1}]$.

Théorème 2 : *A chaque itération de cet algorithme, on a les deux inclusions suivantes :*

$$[\mathbf{y}_{l+1}] \subseteq [\mathbf{y}_l] \text{ et } [\mathbf{x}_{j+1}] \subseteq [\mathbf{y}_l] \ \forall l = 0, 1, 2, ...$$

Généralement, les conditions fixées par ce théorème sont très restrictives. En pratique, la solution *a priori* $[\tilde{\mathbf{x}}_j]$ est de taille considérable ce qui rend impossible l'inversion de $[\tilde{\mathbf{S}}_j]$. Cependant, il suffit d'utiliser un encadrement de $\mathbf{x}(t)$ à l'instant de t_{j+1} et de définir $[\tilde{\mathbf{S}}_j] = \mathbf{I} + \sum_{i=1}^{k-1} h_j^i \mathbf{J}(\mathbf{f}^{[i]}; [\mathbf{x}_{j+1}])$ et $[\mathbf{y}_0] = [\mathbf{x}_{j+1}]$ pour préserver la validité du théorème 2. Par ailleurs, le pavé $[\mathbf{x}_{j+1}]$ peut être calculé, par exemple, par une méthode d'intégration explicite. Dès lors, nous pouvons considérer la méthode implicite comme étant la phase de correction des méthodes explicites qui représentent la phase de prédiction.

Cette méthode implicite, telle qu'elle est conditionnée et présentée dans le théorème 2, ne prend pas en compte le problème d'enveloppement. Par conséquent, plusieurs variantes de cette méthode peuvent voir le jour pour améliorer ses performances.

II.3.2.8 Une comparaison simple entre la méthode de la valeur moyenne et la méthode implicite dans le cas des EDO linéaires

Pour montrer l'avantage de la méthode implicite sur la méthode explicite, en terme de capacité à réduire le pessimisme sur un pas d'intégration, nous allons traiter, pour des raisons de simplicité, le cas des EDO linéaires.

Considérons le système d'équations différentielles linéaires suivant :

$$\begin{cases} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}, \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \in [\mathbf{x}_0] \end{cases} \tag{II.40}$$

où $\mathbf{A} \in \mathbb{R}^{n \times n}$. Soit $\mathbf{I}$ une matrice identité et en supposant que nous disposons d'une solution *a priori* $[\tilde{\mathbf{x}}_0]$ de (II.40) sur l'intervalle temporel $[0, h]$, où h est le pas d'intégration.

Dans ce contexte, la méthode de la valeur moyenne (II.18) donne,

$$\mathbf{x}(h) \in [\mathbf{x}_1] = \mathbf{B}\hat{\mathbf{x}}_0 + \mathbf{B}([\mathbf{x}_0] - \hat{\mathbf{x}}_0) + [\mathbf{z}_1]$$

où $\mathbf{B} = \mathbf{I} + \sum_{i=1}^{k} \frac{1}{i!}(h\mathbf{A})^i$ est une matrice réelle et $[\mathbf{z}_1] = \frac{1}{k!}(h\mathbf{A})^{k+1}[\tilde{\mathbf{x}}_0]$ représente un encadrement garanti du terme reste. De ce fait, la taille du pavé $[\mathbf{x}_1]$ est

$$w_{ex}([\mathbf{x}_1]) = |\mathbf{B}| w([\mathbf{x}_0]) + w([\mathbf{z}_1]). \tag{II.41}$$

En revanche, pour le cas de la méthode implicite, en itérant une seule fois l'algorithme II présenté dans le théorème 2, on obtient :

$$\begin{aligned} \mathbf{x}(h) \in [\mathbf{x}_1] &= (\hat{\tilde{\mathbf{x}}}_0 - \mathbf{C}^{-1}[\mathbf{g}](\hat{\tilde{\mathbf{x}}}_0)) \cap [\tilde{\mathbf{x}}_0] \\ &= (\hat{\tilde{\mathbf{x}}}_0 - \mathbf{C}^{-1}(\mathbf{C}\hat{\tilde{\mathbf{x}}}_0 - [\mathbf{x}_0] + [\mathbf{z}'_1])) \cap [\tilde{\mathbf{x}}_0] \\ &= \mathbf{C}^{-1}([\mathbf{x}_0] - [\mathbf{z}'_1]) \cap [\tilde{\mathbf{x}}_0]. \end{aligned}$$

Où $\mathbf{C} = \mathbf{I} + \sum_{i=1}^{k} \frac{1}{i!}(-h\mathbf{A})^i$ est aussi une matrice réelle inversible et

$$[\mathbf{z}'_1] = \begin{cases} \quad [\mathbf{z}_1] & \text{si } k \text{ est impaire,} \\ - \quad [\mathbf{z}_1] & \text{si } k \text{ est paire.} \end{cases}$$

Donc, il est évident que $w([\mathbf{z}'_1]) = w([\mathbf{z}_1])$. Par conséquent la taille de $[\mathbf{x}_1]$ est donnée par,

$$w_{im}([\mathbf{x}_1]) = |\mathbf{C}^{-1}|(w([\mathbf{x}_0]) + w([\mathbf{z}_1])). \tag{II.42}$$

En analysant (II.41) et (II.42), on voit clairement que la méthode explicite engendre toujours un encadrement $[\mathbf{x}_1]$ de taille $w_{ex}([\mathbf{x}_1]) \geq w([\mathbf{z}_1])$ ce qui n'est pas le cas pour la méthode implicite. En effet, un choix approprié du pas d'intégration h permet d'obtenir une matrice $\mathbf{C}^{-1}$ tel que

$$w_{im}([\mathbf{x}_1]) = |\mathbf{C}^{-1}|(w([\mathbf{x}_0]) + w([\mathbf{z}_1])) \leq w([\mathbf{z}_1]).$$

Exemple [85] : Soit l'équation différentielle autonome définie par

$$\mathbf{A} = \begin{pmatrix} -50.5 & 49.5 \\ -49.5 & -05.5 \end{pmatrix}, \ [\mathbf{x}_0] = \begin{pmatrix} [0,1] \\ [0,1] \end{pmatrix}, \ [\tilde{\mathbf{x}}_0] = \begin{pmatrix} [-2,2] \\ [-2,2] \end{pmatrix}.$$

La réalisation d'un pas d'intégration de ce problème avec la méthode de la valeur moyenne et avec la méthode implicite, pour différents réglages de l'ordre du développement de

h	0.1	0.3	0.5	0.5
k	30	50	90	100
w_{ex}	$7.7\mathbf{e}^{-3}$	$1.7\mathbf{e}^{+2}$	$3.3\mathbf{e}^{+1}$	$1.1\mathbf{e}^{-2}$
w_{im}	$7.7\mathbf{e}^{-3}$	$5.4\mathbf{e}^{-5}$	$3.9\mathbf{e}^{-10}$	$1.4\mathbf{e}^{-11}$

Tab. II.1 *: Comparaison entre les Tailles des deux méthodes.*

Taylor k et différents pas d'intégration h, donne les résultats présentés dans le tableau ci-dessous.

Récemment, Nedialkov [70] a proposé un nouveau schéma d'intégration garanti dans lequel on trouve à la fois les aspects explicite et implicite de l'intégration. Ce schéma fait l'objet de l'**Annexe A** et une brève description dans la section suivante.

II.4 Modèle de Hermite-Obreschkoff

Jusqu'à présent, nous avons présenté des méthodes d'intégration numérique garantie basées sur l'expansion de Taylor. Cependant, dans la littérature, il existe un autre polynôme qui permet d'intégrer numériquement des fonctions différentiables et de fournir une expression explicite de l'erreur de troncature. Ce polynôme est celui de Hermite-Obreschkoff [32, 33, 101, 100] développé initialement pour remédier aux problèmes de l'instabilité dans le cadre de la résolution de systèmes d'équations différentielles raides. Nedialkov [70, 69] a proposé une version intervalle de ce polynôme en introduisant quelques modifications pour contrôler, bien évidemment, la propagation du pessimisme dû au phénomène d'enveloppement.

En pratique, les schémas d'intégration intervalle basés sur ce polynôme sont plus stables que ceux basés sur le développement en série de Taylor, ce qui les favorise pour résoudre une large classe des EDO non linéaires. En effet, généralement, les encadrements de solutions obtenus avec la méthode de Hermite-Obreschkoff sont de meilleure précision que ceux obtenus avec les méthodes d'intégration basées sur le modèle de Taylor.

Dans l'**Annexe A**, nous allons aborder le polynôme de Hermite-Obreschkoff dans le cas ponctuel d'intégration d'EDO. Ensuite, nous présentons son extension aux intervalles réalisée par Nedialkov [70]. Nous présentons également les algorithmes permettant la mise

en œuvre informatique de cette méthode.

II.5 Limite des méthodes d'intégration numérique garantie

Malgré l'utilisation de la représentation parallélépipèdique pour faire propager les encadrements de solutions et les différents schémas d'intégration, Taylor explicite, Taylor implicite et le polynôme de Hermite-Obreschkoff, l'effet d'enveloppement persiste et provoque généralement l'explosion de la taille de ces encadrements si le pavé de conditions initiales est de taille relativement large.

Exemple : Soit le système d'équations différentielles ci-dessous

$$\begin{cases} \dot{x_1} &= x_2 \\ \dot{x_2} &= 5(1-x_1^2)x_2 - x_1. \end{cases} \tag{II.43}$$

Avec $\mathbf{x}_0 \in [\mathbf{x}_0]$.

L'intégration numérique garantie de ce système, par la méthode de Hermite-Obreschkoff avec un ordre 38, a permis d'obtenir les courbes représentatives des solutions de (II.43) sur une période $t \in [0, 10]$. Les courbes rouges en ligne continue sur les figures II.6 et II.7, correspondent à l'encadrement de solutions calculé pour un pavé de contions initiales $[\mathbf{x}_0] = [2.0001, 2.0005] \times [0.0001, 0.0005]$. Les courbes en bleu quant à elles, sont obtenues pour un pavé de conditions initiales $[\mathbf{x}_0] = [1.99998, 2.0006] \times [-0.0002, 0.0006]$. Sur ces figures, on voit bien que les méthodes d'intégration numérique garantie divergent sous l'effet d'enveloppement au bout de quelques pas d'intégration même pour des pavés de conditions initiales de petites tailles. Pour surmonter ce problème, on doit avoir recours à la bissection de l'espace de variables, ce qui augmente considérablement le temps du calcul et limite ainsi l'utilisation de ces méthodes pour résoudre les problèmes de grande dimension.

Par ailleurs, les auteurs de [21] ont proposé d'utiliser les techniques basées sur le principe de contraction des pavés par l'analyse de consistance pour réduire l'effet d'enveloppement.

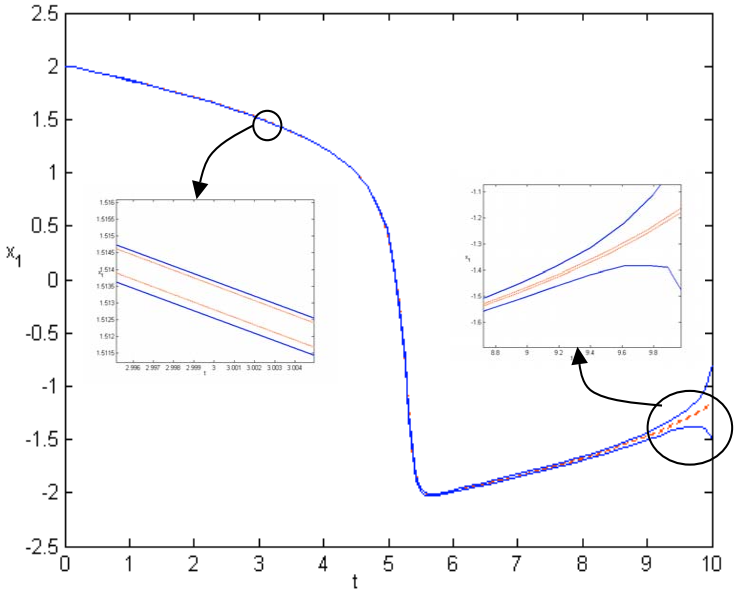


Fig. II.6 *: Exemple d'intégration avec la méthode de Hermite-Obreschkoff : Première composante du vecteur solution* $\mathbf{x}(t)$.

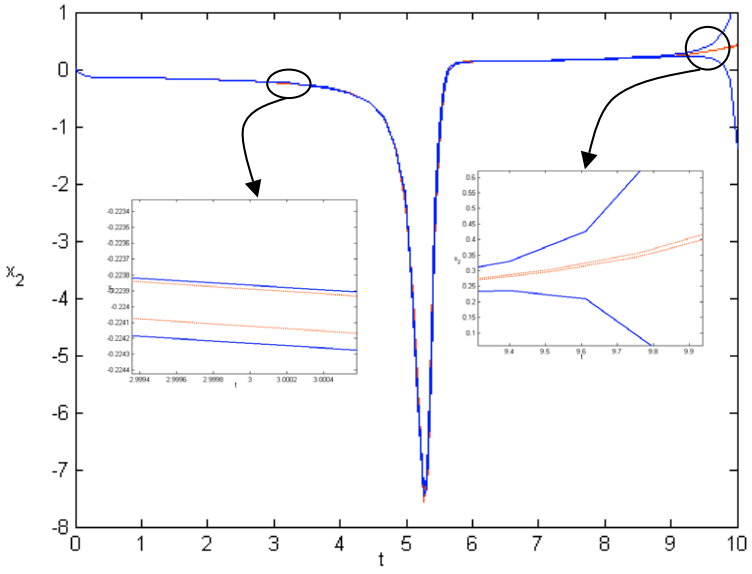


Fig. II.7 *: Exemple d'intégration avec la méthode de Hermite-Obreschkoff : Deuxième composante du vecteur solution* $\mathbf{x}(t)$.

II.6 Conclusion

Ce deuxième chapitre a été consacré aux méthodes d'intégration numérique garantie des équations différentielles ordinaires. Nous avons présenté en premier lieu leur principe basé sur le développement en série de Taylor intervalles. Par la suite, nous avons introduit les techniques qui permettent de réduire la propagation du pessimisme dû à l'effet d'enveloppement. Par ailleurs, nous avons évoqué leur limite due à cet effet d'enveloppement qui conduit généralement à l'explosion de la taille des encadrements solutions. Dans le chapitre suivant nous allons présenter une autre approche d'intégration numérique garantie qui s'appuie sur la théorie de comparaison des inégalités différentielles.

Chapitre III

Théorie de comparaison des inégalités différentielles

III.1 Introduction

Nous introduisons dans ce chapitre une autre approche d'encadrement garanti de solutions d'équations différentielles incertaines. Cette approche utilise la théorie de comparaison des inégalités différentielles développée conjointement pour des systèmes d'équations différentielles monotones [46, 88, 45] et dans le contexte fixé par le théorème de Müller [66, 99, 94, 63, 53].

L'idée fondamentale de cette approche d'intégration numérique garantie consiste à transformer le problème incertain (*intervalle*) en deux problèmes déterministes (*ponctuels*). La résolution de ces deux nouveaux problèmes déterministes donnera deux solutions, minimale et maximale, qui encadreront de manière garantie toutes les solutions possibles du problème original incertain. Comme les deux systèmes d'équations décrivant les systèmes englobants ne contiennent plus d'incertitudes au sens physique, les méthodes d'intégration numériques à base de modèles de Taylor présentées dans le chapitre précédent pourront être utilisées et réussiront à fournir des résultats exploitables.

Nous consacrons la première partie de ce chapitre aux définitions et propriétés liées à cette théorie qui s'avèreront utiles pour la compréhension des méthodes proposées par la suite. Puis nous abordons dans la seconde partie, la résolution des EDO incertaines en utilisant la théorie des inégalités différentielles. Nous verrons en particulier que pour obte-

nir les deux systèmes englobants, il peut être nécessaire d'accéder aux dérivées partielles du vecteur $\mathbf{f}$ par rapport à toutes les composantes du vecteur de variables $\mathbf{x}$ et celles du vecteur de paramètres $\mathbf{p}$.

III.2 Définitions et propriétés

III.2.1 Systèmes monotones

Par définition, les équations différentielles monotones sont des systèmes dynamiques générant des trajectoires d'état qui respectent, sur un domaine de définition donné, la relation d'ordre supérieur "$\geq$" définie par rapport aux conditions initiales [46, 45, 88, 55, 3]. Autrement dit :

Soient $\mathbf{z}(t, \mathbf{z}_0, \mathbf{u}_1)$ et $\mathbf{y}(t, \mathbf{y}_0, \mathbf{u}_2)$ deux trajectoires solutions d'un système dynamique monotone défini par

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \tag{III.1}$$

respectivement pour $\mathbf{x}(t_0) = \mathbf{z}_0$, $\mathbf{u}(t) = \mathbf{u}_1$ et pour $\mathbf{x}(t_0) = \mathbf{y}_0$ et $\mathbf{u}(t) = \mathbf{u}_2$. Alors,

$$\mathbf{y}_0 \geq \mathbf{z}_0 \text{ et } \mathbf{u}_2 \geq \mathbf{u}_1 \Rightarrow \mathbf{y}(t, \mathbf{y}_0, \mathbf{u}_2) \geq \mathbf{z}(t, \mathbf{z}_0, \mathbf{u}_1) \ \forall t \geq t_0. \tag{III.2}$$

Ici la relation d'ordre $\geq$ doit être respectée composante par composante, c'est-à-dire que $y_{0,i} \geq z_{0,i} \, \forall i \in \{1, ..., n\}$ et $u_{2,j} \geq u_{1,j} \, \forall j \in \{1, ..., m\}$, où n et m sont respectivement les dimensions des vecteurs $\mathbf{x}$ et $\mathbf{u}$.

III.2.2 Préservation d'ordre

Par ailleurs, une généralisation de la notion de monotonie des EDO est la préservation d'ordre [55, 3]. En effet, on dit qu'un système d'équations différentielles est préservateur d'ordre, par rapport à une relation d'ordre quelconque, s'il existe deux matrices diagonales de dimensions respectivement $n \times n$ et $m \times m$,

$$\begin{gathered} \mathbf{D} = diag[(-1)^{\varepsilon_1}, ..., [(-1)^{\varepsilon_n}], \varepsilon_i \in \{0, 1\} \\ \text{et} \\ \mathbf{P} = diag[(-1)^{\varepsilon_1}, ..., [(-1)^{\varepsilon_m}], \varepsilon_j \in \{0, 1\} \end{gathered} \tag{III.3}$$

tel que,

$$\mathbf{Dy}_0 \geq \mathbf{Dz}_0 \text{ et } \mathbf{Pu}_2 \geq \mathbf{Pu}_1 \Rightarrow \mathbf{Dy}(t, \mathbf{y}_0, \mathbf{u}_2) \geq \mathbf{Dz}(t, \mathbf{z}_0, \mathbf{u}_1) \ \forall t \geq t_0. \tag{III.4}$$

Là aussi, la relation d'ordre $\geq$ doit être respectée composante par composante.

Dans la section suivante, nous allons faire un rappel sur une approche graphique qui permet de vérifier cette propriété de préservation d'ordre des EDO [55, 4, 90]. Notons que cette méthode n'est applicable que pour le cas des équations différentielles dans lesquelles les signes des dérivées partielles,

$$\begin{array}{l} \frac{\partial f_i}{\partial x_j}(.,.) \text{ pour } i \neq j \, \forall i \in \{1, ..., n\}, \, \forall j \in \{1, ..., n\} \text{ et} \\ \frac{\partial f_i}{\partial u_k}(.,.) \ \forall i \in \{1, ..., n\}, \, \forall k \in \{1, ..., m\}, \end{array}$$

demeurent constants sur toute la durée d'étude.

III.2.3 Condition graphique pour la préservation d'ordre

Considérons le système d'équations différentielles (III.1). Le graphe d'incidence associé à ce système est composé de nœuds qui représentent les variables $x_i, ..., x_n, u_1, ..., u_m$ et des arcs signés $e^x_{i,j}$ et $e^u_{i,j}$ qui représentent les liaisons entre deux nœuds x_i et x_j ou entre x_i et u_k. La construction de ce graphe est réalisée en respectant les règles ci-dessous :

• **Arcs entre les nœuds** x_i :

1. Si $\frac{\partial f_i}{\partial x_j}(.,.) > 0 \ \forall (\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}$, nous traçons un arc positif $e^x_{i,j}$ orienté du nœud x_j vers le nœud x_i.
2. Si $\frac{\partial f_i}{\partial x_j}(.,.) < 0 \ \forall (\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}$, nous traçons un arc négatif $e^x_{i,j}$ orienté du nœud x_j vers le nœud x_i.
3. Sinon, $\frac{\partial f_i}{\partial x_j}(.,.) = 0 \ \forall \mathbf{x}, \mathbf{u} \in \mathbb{X} \times \mathbb{U}$. Par conséquent, il n'aura aucun arc entre les nœuds x_i et x_j.

$\mathbb{X}$ et $\mathbb{U}$ représentent respectivement les domaines d'évolutions des vecteurs $\mathbf{x}$ et $\mathbf{u}$.

• **Arcs entre les nœuds** x_i **et** u_k :

1. Si $\frac{\partial f_i}{\partial u_k}(.,.) > 0 \ \forall (\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}$, nous traçons un arc positif $e^u_{i,j}$ orienté du nœud u_k vers le nœud x_i.
2. Si $\frac{\partial f_i}{\partial u_k}(.,.) < 0 \ \forall \mathbf{x}, \mathbf{u} \in \mathbb{X} \times \mathbb{U}$. Alors, nous traçons un arc négatif $e^u_{i,j}$ orienté du nœud u_k vers le nœud x_i.

3. Sinon, $\frac{\partial f_i}{\partial u_k}(.,.) = 0 \ \forall \mathbf{x}, \mathbf{u} \in \mathbb{X} \times \mathbb{U}$. Par conséquent, il n'aura aucun arc entre les nœuds x_i et u_k.

Ainsi, nous définissons par chemin orienté $\mathfrak{C}$ de longueur l toute séquence finie de nœuds $v_{n_0}, v_{n_1}...v_{n_l}$, dans laquelle chaque nœud $v_{n_i}\, i = 0, ..., l$ est apparu au plus une seule fois et pour chaque deux nœuds successifs il existe un arc $e_{i,j}$ qui les relie. Par ailleurs, un cycle de ce graphe, pas nécessairement orienté, est aussi une séquence finie de nœuds $v_{n_0}, v_{n_1}...v_{n_l}$ tel que $v_{n_0} = v_{n_l}$ et pour chaque deux nœuds successifs il existe un arc $e_{i,j}$ qui les relie. Dès lors, nous définissons le signe d'un cycle comme étant le produit entre les signes de ses arcs.

Proposition 1 : *Un système d'équations différentielles du type* (III.1)*, qui admet un graphe d'incidence conçu en respectant l'ensemble des règles présentées ci-dessus, est préservateur d'ordre si et seulement si son graphe d'incidence ne contient aucun cycle de signe négatif [55, 4].*

Dans la suite de ce chapitre, nous allons appeler les systèmes d'équations différentielles préservateurs d'ordre par systèmes monotones, c'est-à-dire monotones par rapport à une relation d'ordre donnée.

Exemple 1 : Soit le système d'équations différentielles

$$\begin{cases} \dot{x}_1 &= x_1(-x_1 + x_2) \\ \dot{x}_2 &= 3x_2(-x_1 + u) \end{cases} \tag{III.5}$$

évoluant dans l'espace $\mathbb{R}^2_{\geq 0}$. Son graphe d'incidence est représenté sur la figure III.1.

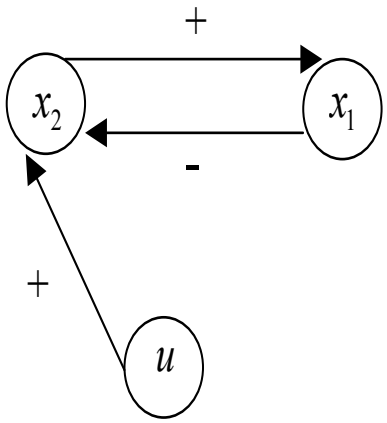


Fig. III.1 *: Graphe d'incidence d'un système non préservateur d'ordre.*

Il est clair que ce graphe contient un seul cycle qui est de signe négatif

$$sign(e_{1,2}).sign(e_{2,1}) < 0.$$

Par conséquent, ce système d'équations différentielles n'est pas monotone (préservateur d'ordre).

Exemple 2 : Considérons maintenant le système ci-dessous

$$\begin{cases} \dot{x}_1 &= -u\theta_1(x_1) + \theta_2(1 - x_1 - x_2) \\ \dot{x}_2 &= u\theta_3(1 - x_1 - x_2) - \theta_4(x_2) \end{cases} \tag{III.6}$$

où les fonctions $\theta_i(.)$ ont des dérivées strictement positives et satisfont $\theta_i(0) = 0$. Le domaine d'évolution de ce système est défini par l'ensemble

$$\mathbb{X} = \{x_1 \geq 0, x_2 \geq 0, x_1 + x_2 \leq 1\}.$$

La figure III.2 montre son graphe d'incidence. On voit bien que ce graphe contient trois

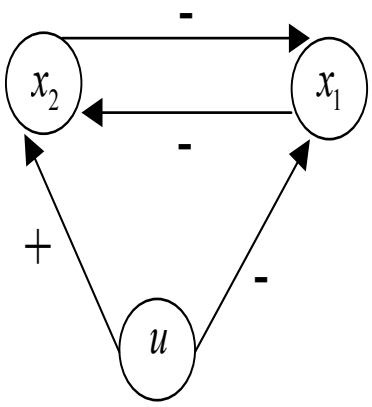


Fig. III.2 *: Graphe d'incidence d'un système préservateur d'ordre*

cycles tous de signe positif.

1. cycle 1 x_1, x_2, x_1 de signe positif $sign(e^x_{1,2}).sign(e^x_{2,1}) > 0$
2. cycle 2 x_1, x_2, u, x_1 de signe positif $sign(e^x_{1,2}).sign(e^u_{1,2}).sign(e^u_{1,1}) > 0$
3. cycle 3 x_1, x_2, u, x_1 de signe positif $sign(e^x_{2,1}).sign(e^u_{1,2}).sign(e^u_{1,1}) > 0$

Donc, ce système est préservateur d'ordre, c'est-à-dire monotone par rapport à une certaine relation d'ordre définie par rapport à ses conditions initiales.

Remarque : Généralement, on peut transformer un système préservateur d'ordre en un système monotone en effectuant les changements de variables suivants :

$$\mathbf{y} = \mathbf{D}\mathbf{x} \text{ et } \mathbf{w} = \mathbf{P}\mathbf{u} \tag{III.7}$$

où les matrices $\mathbf{D}$ et $\mathbf{P}$ sont définies par (III.3).

III.2.4 Systèmes coopératifs

Les EDO coopératives sont des systèmes dynamiques dans lesquels les variables d'état agissent positivement les unes sur les autres. En fait, la propriété de *coopérativité* est vérifiée par un grand nombre de systèmes réels. Les systèmes de diffusion-réaction sont coopératifs, une grande partie des modèles écrits en biologie, en chimie ou en économie le sont ou peuvent être réécrits pour vérifier cette propriété [3, 4, 46, 88].

Définition 1 :

Un système d'équations différentielles du type (III.1) est dit quasi-monotone s'il satisfait la condition de Kamke [93, 73, 88], c'est-à-dire si toutes ses composantes $f_i(\mathbf{x}, \mathbf{u}(t))$ ($i \in \{1, 2, ..., n\}$) sont monotones par rapport à chaque variable x_k du vecteur $\mathbf{x}$ avec $k \neq i$,

$$\forall(x_{1_j} \leq x_{2_j}\ j \neq i,\ x_{1i} = x_{2i}),\ \text{et}\ u_{1_k} \leq u_{2_k} \Rightarrow f_i(\mathbf{x}_2, \mathbf{u}_2) - f_i(\mathbf{x}_1, \mathbf{u}_1) \geq 0. \tag{III.8}$$

Dans [3, 4] les auteurs ont réécrit cette inégalité (III.8) sous forme de conditions sur les signes des dérivées partielles du champs de vecteurs $\mathbf{f}$.

Proposition 2 : *Supposons que* $\mathbf{f}$ *est continûment différentiable, alors le système* (III.1) *est coopératif si est seulement si les conditions suivantes sont satisfaites :*

$$\begin{array}{l} \frac{\partial f_i}{\partial x_j}(\mathbf{x}, \mathbf{u}) \geq 0\ \forall \mathbf{x} \in \mathbb{X},\ \forall \mathbf{u} \in \mathbb{U},\ \forall i \neq j \\ \frac{\partial f_i}{\partial u_j}(\mathbf{x}, \mathbf{u}) \geq 0\ \forall \mathbf{x} \in \mathbb{X},\ \forall \mathbf{u} \in \mathbb{U} \end{array} \tag{III.9}$$

pour tout $i \in \{1, 2, ..., n\}$ *et pour tout* $j \in \{1, 2, ..., m\}$.

Remarque : Les systèmes coopératifs sont par définition des systèmes monotones.

Une des propriétés des systèmes dynamiques coopératifs, qui s'avère très intéressante pour l'intégration numérique garantie des EDO incertaines, est le principe de comparaison d'opérateurs différentiels [99]. Dans ce contexte ce principe est énoncé comme suit :

Propriété 1 : (Principe de comparaison dans le cas des systèmes coopératifs [89])

Soient deux systèmes coopératifs

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)) \\ \dot{\mathbf{y}}(t) &= \mathbf{g}(\mathbf{y}(t), \mathbf{w}(t)),\end{aligned} \tag{III.10}$$

où $\mathbf{x}, \mathbf{y} \in \mathbb{X}$ et $\mathbf{u}, \mathbf{w} \in \mathbb{U}$. Alors,

$$\begin{aligned}(h_i(\mathbf{x}, \mathbf{u}) \leq g_i(\mathbf{x}, \mathbf{u})) \wedge (x_{0_i} \leq y_{0_i})\, \forall i \in \{1, ...n\},\, \forall \mathbf{x} \in \mathbb{X},\, \forall \mathbf{u} \in \mathbb{U} \\ \Rightarrow \forall i \in \{1, ...n\},\, x_i(t) \leq y_i(t)\, \forall t \geq t_0.\end{aligned} \tag{III.11}$$

Une autre propriété séduisante des systèmes coopératifs est la positivité. Cette propriété se traduit par le fait que l'espace atteignable d'un système dynamique positif initialisé dans $\mathbb{R}^n_{\geq 0}$, demeure invariant vis-à-vis de celui-ci. Dans [3] les auteurs ont montré qu'un système linéaire est coopératif si et seulement si il est positif.

Propriété 2 : (Positivité)

Soit le système d'équations différentielles linéaires positif ci-dessous,

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t). \tag{III.12}$$

où $\mathbf{A}(t)$ est une matrice coopérative, c'est-à-dire que ses éléments non diagonaux sont tous non négatifs $a_{i,j}(t) \geq 0,\, \forall i \neq j,\, \forall t \geq t_0$. De même pour les éléments de la matrice $\mathbf{B}(t)$, $b_{i,j}(t) \geq 0,\ \forall (i,j),\ \forall t \geq t_0$. Alors

$$\forall\, (i,j)\, u_{i,j}(t) \geq 0 \text{ et } \forall i\ x_{0_i} \geq 0\ \Rightarrow\ x_i(t) \geq 0\ \forall t > t_0,\ \forall i. \tag{III.13}$$

Par ailleurs, on dit qu'un système de type (III.1) est incrémentalement positif ou variationnellement positif si, pour chaque solution $\mathbf{x}(t)$ de (III.1), son linéarisé

$$\dot{\mathbf{z}}(t) = \mathbf{L}(t)\mathbf{z}(t) + \mathbf{E}(t)\mathbf{v}(t), \tag{III.14}$$

est positif. Par analogie avec le cas linéaire, D. Angeli et D. Sontag ont montré dans [3] que si le champ de vecteurs $\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$ est continûment différentiable, alors le système (III.1) est coopératif si et seulement si il est incrémentalement positif.

Propriété 3 :

Si l'EDO (III.1) est coopérative, alors pour toute entrée positive $\forall t \geq 0$, $\mathbf{u}(t) \geq 0$ nous avons la propriété suivante

$$\forall \mathbf{x}(t_0) \in \mathbb{X},\ \mathbf{f}(\mathbf{x}(t_0), \mathbf{u}(t_0)) \geq 0 \Rightarrow \forall t \geq t_0\ \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \geq 0. \tag{III.15}$$

La démonstration de cette propriété est donnée dans [88].

III.3 Intégration numérique garantie des ODE monotones

L'intégration numérique garantie des EDO monotones en présence d'incertitudes, conjointement sur les conditions initiales et sur le vecteur de paramètres $\mathbf{p}$, est réalisée en deux étapes. La première étape consiste à transformer l'EDO monotone incertaine en deux EDO monotones certaines minimale et maximale. La résolution de ces deux ODE fournit des approximations inférieure et supérieure des solutions minimale et maximale de l'EDO incertaine. Par la suite, la deuxième étape consiste à utiliser les méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalles pour calculer ces approximations de solutions. Notons que la transformation réalisée dans la première étape permet de contourner efficacement le problème de la propagation du pessimisme dû simultanément à l'effet d'enveloppement et au phénomène de dépendance.

Sachant qu'on peut réécrire un système monotone en un système coopératif en utilisant le changement de variables défini par (III.3) et (III.7), nous considérons dans la suite de cette section le cas des systèmes d'équations différentielles coopératifs présentés par,

$$\begin{cases} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}, \mathbf{p}, \mathbf{u}(t)) \\ \mathbf{x}_0 &\in [\mathbf{x}_0] \subseteq \mathbb{X} \\ \mathbf{p} &\in [\mathbf{p}] \\ \mathbf{u}(t) &\in \mathbb{U} \end{cases} \tag{III.16}$$

où $\mathbf{p}$ représente le vecteur de paramètres incertains de dimension r.

Le théorème que nous présentons ci-dessous, est fondé sur la Propriété 1 des systèmes dynamiques coopératifs. Ce théorème met en avant les conditions que doivent vérifier les EDO certaines pour que leur résolution fournisse un encadrement garanti de toutes les solutions possibles de (III.16) [99, 94, 53].

Théorème 1 : *Soient les vecteurs* $\underline{\Theta}$ *et* $\overline{\Theta}$ *de dimension inférieure ou égale à* $r \times n$ *telle que*

$$\begin{gathered} \forall i \in \{1, ..., n\} \\ f_i(\mathbf{x}, \underline{\Theta}, .) \leq f_i(\mathbf{x}, \mathbf{p}, .) \leq f_i(\mathbf{x}, \overline{\Theta}, .) \end{gathered} \tag{III.17}$$

et soient $\underline{\mathbf{x}}(t)$ *et* $\overline{\mathbf{x}}(t)$ *les solutions des deux EDO coopératives certaines* (III.18) *et* (III.19)

conçues à partir du système incertain (III.16)

$$\dot{\underline{\mathbf{x}}}(t) = \mathbf{f}(\underline{\mathbf{x}}, \underline{\Theta}, \mathbf{u}_1(t)) \tag{III.18}$$

$$\dot{\overline{\mathbf{x}}}(t) = \mathbf{f}(\overline{\mathbf{x}}, \overline{\Theta}, \mathbf{u}_2(t)) \tag{III.19}$$

Alors, compte tenu de la Propriété 1 des systèmes coopératifs et si pour tout $i \in \{1, ..., n\}$, $\underline{x}_i(t_0) \leq \overline{x}_i(t_0)$, *on affirme que :*

$$\begin{gathered} \forall i, \forall t \geq t_0, \forall \mathbf{p} \in [\mathbf{p}], \forall \mathbf{u}(t) \in [\mathbf{u}_1(t), \mathbf{u}_2(t)] \ \textit{et} \ \forall \mathbf{x}(t_0) \in [\underline{\mathbf{x}}(t_0), \overline{\mathbf{x}}(t_0)] \\ \underline{x}_i(t) \leq x_i(t) \leq \overline{x}_i(t) \end{gathered} \tag{III.20}$$

Par ailleurs, pour chaque $\mathbf{x}_0 \in [\mathbf{x}_0], \mathbf{p} \in [\mathbf{p}]$ *et* $\mathbf{u}(t) \in \mathbb{U}$ *l'unicité de la solution est garantie si la fonction* $\mathbf{f}$ *est Lipschitzienne par rapport à* $\mathbf{x}$.

Preuve : Voir Propriété 1 des systèmes coopératifs, la double inégalité (III.17) et les références [88, 99].

Ainsi, en terme de l'arithmétique par intervalles, nous obtenons une fonction d'inclusion de solutions de (III.16) donnée par la relation suivante :

$$[\mathbf{x}](t) = [\underline{\mathbf{x}}(t), \overline{\mathbf{x}}(t)]. \tag{III.21}$$

où $\underline{\mathbf{x}}(t)$ et $\overline{\mathbf{x}}(t)$ sont respectivement les solutions des équations différentielles inférieure (III.18) et supérieure (III.19). Par ailleurs, pour garantir numériquement la relation (III.21) nous pouvons utiliser une des méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalles pour calculer $\underline{\mathbf{x}}(t)$ et $\overline{\mathbf{x}}(t)$.

III.3.1 Obtention de solutions minimale et maximale

Le choix des vecteurs $\underline{\Theta}$ et $\overline{\Theta}$ est très important pour construire les EDO englobantes (III.18) et (III.19). En effet, c'est ces deux vecteurs qui assurent le calcul d'un encadrement garanti de toutes les solutions possibles de (III.16) par le biais de l'intégration numérique garantie de (III.18) et de (III.19).

Nous montrons, dans cette section, une règle de sélection [53, 54] de ces deux vecteurs en fonction des bornes du pavé de paramètres $[\mathbf{p}]$. Cette méthode s'applique aux systèmes dont les signes des dérivées partielles de $\mathbf{f}$ par rapport aux vecteur $\mathbf{p}$ sont constants.

Règle 1 : (Analyse de signes des dérivées partielles $\frac{\partial f_i}{\partial p_k}$)
Pour tout $i \in \{1, ..., n\}$ et pour tout $k \in \{1, ..., r\}$, nous définissons les éléments $\overline{\theta}_i^k$ par

$$\overline{\theta}_i^k = \begin{cases} \overline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(.,.,.) \geq 0 \ \forall \mathbf{x} \in \mathbb{X}, \forall \mathbf{p} \in [\mathbf{p}], \forall \mathbf{u}(t) \in \mathbb{U} \\ \underline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(.,.,.) < 0 \ \forall \mathbf{x} \in \mathbb{X}, \forall \mathbf{p} \in [\mathbf{p}], \forall \mathbf{u}(t) \in \mathbb{U} \end{cases} \tag{III.22}$$

ainsi le vecteur $\overline{\Theta} = [\overline{\theta}_1^1, ..., \overline{\theta}_n^1, ..., \overline{\theta}_n^r]^T$. De la même façon, nous obtiendrons les éléments $\underline{\theta}_i^k$ par

$$\underline{\theta}_i^k = \begin{cases} \underline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(.,.,.) \geq 0 \ \forall \mathbf{x} \in \mathbb{X}, \forall \mathbf{p} \in [\mathbf{p}], \forall \mathbf{u}(t) \in \mathbb{U} \\ \overline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(.,.,.) < 0 \ \forall \mathbf{x} \in \mathbb{X}, \forall \mathbf{p} \in [\mathbf{p}], \forall \mathbf{u}(t) \in \mathbb{U} \end{cases} \tag{III.23}$$

d'où le vecteur $\underline{\Theta} = [\underline{\theta}_1^1, ..., \underline{\theta}_n^1, ..., \underline{\theta}_n^r]^T$.

Ainsi, cette règle permet de calculer le minimum $\underline{\mathbf{f}}$ et le maximum $\overline{\mathbf{f}}$ de $\mathbf{f}$ sur le domaine de paramètre $[\mathbf{p}]$. Donc, elle satisfait les doubles inégalités (III.17).

En pratique cette règle permet d'éviter de construire des EDO englobantes instables, c'est-à-dire la taille de l'encadrement (III.21) obtenue dans le contexte de cette règle ne diverge pas. Généralement, cette instabilité est due à l'apparition de la borne inférieure et de la borne supérieure du même paramètre dans l'expression d'une des composantes des champs de vecteurs $\underline{\mathbf{f}}$ ou $\overline{\mathbf{f}}$ [79]. Dans le chapitre suivant, nous allons étendre l'application de cette règle pour le cas des EDO où au moins une de ces dérivées partielles change de signe en fonction de t.

III.3.2 Exemples illustratifs

Les exemples suivants illustrent la mise en application de la Règle 1 pour la construction des EDO englobantes. De plus, ils montrent les avantages de l'intégration numérique garantie basée sur le principe de comparaison des inégalités différentielles par rapport à celle fondée sur les modèles de Taylor intervalles.

Exemple 1 : Soit l'EDO incertaine coopérative qui modélise un système de stockage de matière à deux compartiments

$$\begin{cases} \dot{x}_1 = -p_4 x_1 - \frac{p_1 x_1}{1+p_2 x_2} + p_3 x_2 + u \\ \dot{x}_2 = p_4 x_1 - p_3 x_2 \end{cases} \tag{III.24}$$

où le vecteur de conditions initiales $\mathbf{x}_0$ et celui de paramètres $\mathbf{p}$ sont mal connus et supposés appartenir aux pavés : $\mathbf{x}_0 \in [1, 1.4] \times [0, 0.4]$ et $\mathbf{p} \in [0.9, 1.1] \times [1.1, 1.3] \times [0.45, 0.55] \times [0.2, 0.3]$. L'excitation de ce système est considérée constante $u(t) = 0.1$ avec $t \in [0, 20]$.

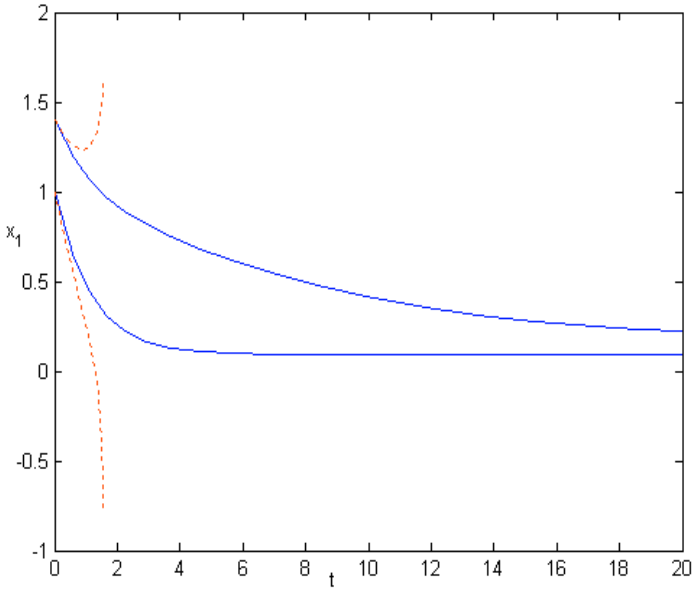


FIG. III.3 *: Encadrement de toutes les valeurs possibles de $x_1(t)$. Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillées méthode de Hermite-Obreschkoff intervalle avec pas d'intégration variable.*

L'application de la Règle 1 permet d'obtenir l'EDO inférieure définie par

$$\begin{cases} \dot{\underline{x}}_1 = -\overline{p}_4\underline{x}_1 - \frac{\overline{p}_1\underline{x}_1}{1+\underline{p}_2\underline{x}_2} + \underline{p}_3\underline{x}_2 + u \\ \dot{\underline{x}}_2 = \underline{p}_4\underline{x}_1 - \overline{p}_3\underline{x}_2. \end{cases} \tag{III.25}$$

Où le vecteur $\underline{\Theta} = [\underline{\theta}_1^1, \underline{\theta}_2^1, \underline{\theta}_3^1, \underline{\theta}_4^1, \underline{\theta}_3^2, \underline{\theta}_4^2]^T = [\overline{p}_1, \underline{p}_2, \underline{p}_3, \overline{p}_4, \overline{p}_3, \underline{p}_4]^T$. Et l'EDO supérieure donnée par

$$\begin{cases} \dot{\overline{x}}_1 = -\underline{p}_4\overline{x}_1 - \frac{\underline{p}_1\overline{x}_1}{1+\overline{p}_2\overline{x}_2} + \overline{p}_3\overline{x}_2 + u \\ \dot{\overline{x}}_2 = \overline{p}_4\overline{x}_1 - \underline{p}_3\overline{x}_2. \end{cases} \tag{III.26}$$

Où le vecteur $\overline{\Theta} = [\overline{\theta}_1^1, \overline{\theta}_2^1, \overline{\theta}_3^1, \overline{\theta}_4^1, \overline{\theta}_3^2, \overline{\theta}_4^2]^T = [\underline{p}_1, \overline{p}_2, \overline{p}_3, \underline{p}_4, \underline{p}_3, \overline{p}_4]^T$.

Donc, l'intégration numérique de ces deux EDO coopératives, (III.25) et (III.26), permet d'obtenir un encadrement garanti de toutes les solutions possibles de (III.24).

En effet, sur les figures III.3 et III.4 les courbes en bleu montrent respectivement l'encadrement obtenu pour la première composante $x_1(t)$ et celui de la deuxième composante $x_2(t)$. En revanche, les courbes pointillées en rouges montrent la limite des méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalles. En effet, on constate sur ces figures que l'encadrement de solutions de (III.24) obtenu avec ces méthodes diverge au bout de quelques pas d'intégration.

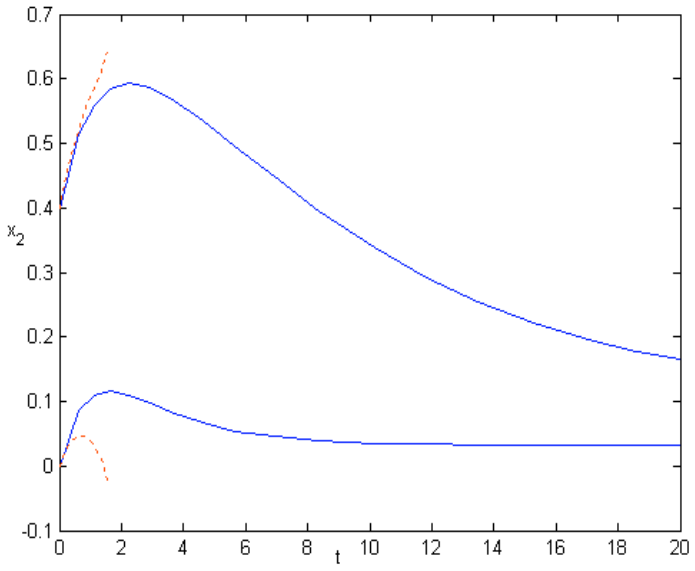


Fig. III.4 *: Encadrement de toutes les valeurs possibles de $x_2(t)$. Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle avec pas d'intégration variable*

Exemple 2 : Considérons l'EDO incertaine coopérative ci dessous

$$\begin{cases} \dot{x}_1 = -p_1 x_1^2 + p_3 x_2 + u \\ \dot{x}_2 = -(p_2 + p_3) x_2 + p_1 x_1^2 \\ \dot{x}_3 = p_2 x_2. \end{cases} \tag{III.27}$$

Où le vecteur des paramètres incertains $\mathbf{p} = [p_1, p_2, p_3]^T \in [1, 1.2] \times [2.5, 2.7] \times [0.3, 0.5]$. Les conditions initiales sont aussi considérées incertaines mais bornées de bornes connues *a priori* $\mathbf{x}_0 \in [0.5, 0.8] \times [0.8, 1.2] \times [0.1, 0.4]$. Nous supposons que l'excitation extérieure $u(t) \in [u_1(t), u_2(t)]$ avec

$$\begin{aligned} u_1(t) &= 1.5(1 + \sin(0.2t)) \\ u_2(t) &= 1.7(1 + \sin(0.2t)). \end{aligned}$$

Par ailleurs, comme ce système d'équations différentielles est coopératif il est donc positif. Alors, pour tout $\mathbf{x}_0 \in [\mathbf{x}_0]$, nous avons :

$$\forall t \geq 0 \begin{cases} x_1(t) \geq 0 \\ x_2(t) \geq 0 \\ x_3(t) \geq 0. \end{cases}$$

Donc, l'application de la Règle 1 pour le système (III.27) permet de construire les deux EDO englobantes suivantes :

1. EDO inférieure

$$\begin{cases} \dot{\underline{x}}_1 = -\overline{p}_1\underline{x}_1^2 + \underline{p}_3\underline{x}_2 + u_1 \\ \dot{\underline{x}}_2 = -(\overline{p}_2 + \overline{p}_3)\underline{x}_2 + \underline{p}_1\underline{x}_1^2 \\ \dot{\underline{x}}_3 = \underline{p}_2\underline{x}_2. \end{cases} \tag{III.28}$$

2. EDO Supérieure

$$\begin{cases} \dot{\overline{x}}_1 = -\underline{p}_1\overline{x}_1^2 + \overline{p}_3\overline{x}_2 + u_2 \\ \dot{\overline{x}}_2 = -(\underline{p}_2 + \underline{p}_3)\overline{x}_2 + \overline{p}_1\overline{x}_1^2 \\ \dot{\overline{x}}_3 = \overline{p}_2\overline{x}_2. \end{cases} \tag{III.29}$$

Dès lors, l'intégration numérique garantie de (III.28) et (III.29) séparément, sur une période $t \in [0, 5]$, permet de caractériser de manière garantie l'ensemble de toutes les solutions possibles de (III.27). Sur les figures III.5, III.6 et III.7, nous avons représenté en lignes bleues continues les encadrements obtenus pour chaque variable $x_i(t), i = 1, 2, 3$. Par ailleurs, les courbes en rouge pointillé montrent la limite des méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalles si les tailles de grandeurs incertaines dépassent un certain seuil.

Exemple 3 : Dans cet exemple, nous allons montrer comment réécrire une EDO non coopérative mais monotone par rapport à une certaine relation d'ordre en une EDO coopérative. Pour ce faire, nous considérons le système d'équations différentielles ci-dessous

$$\begin{cases} \dot{x}_1(t) &= -2x_1 + u(t) \\ \dot{x}_2(t) &= 2x_1 - x_3 \\ \dot{x}_3(t) &= -2x_1 - x_2. \end{cases} \tag{III.30}$$

Il est facile de vérifier que ce système (III.30) n'est pas coopératif. En effet, les signes des dérivées partielles $\frac{\partial f_2}{\partial x_3}$, $\frac{\partial f_3}{\partial x_1}$ et $\frac{\partial f_3}{\partial x_2}$ sont tous négatifs. En revanche, son graphe d'incidence présenté dans la figure III.8, montre que ce système est conservateur d'ordre. Effectivement, les signes des trois cycles qui constituent ce graphe sont tous positifs :

1. cycle 1 x_2, x_3, x_2 de signe positif $sign(e^x_{2,3}).sign(e^x_{3,2}) > 0$
2. cycle 2 x_1, x_2, x_3, x_1 de signe positif $sign(e^x_{1,2}).sign(e^x_{2,3}).sign(e^x_{3,1}) > 0$
3. cycle 3 x_1, x_3, x_2, x_1 de signe positif $sign(e^x_{1,3}).sign(e^x_{3,2}).sign(e^x_{2,1}) > 0$

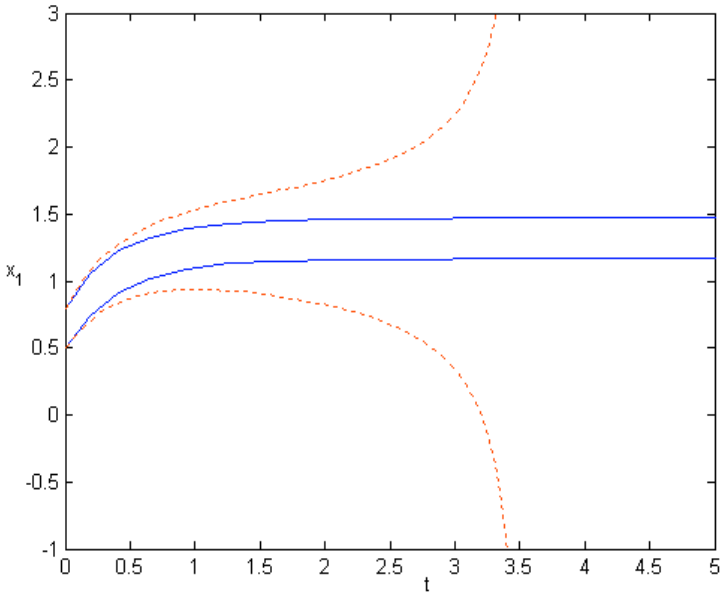


Fig. III.5 : *Encadrement de toutes les valeurs possibles de $x_1(t)$.Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle.*

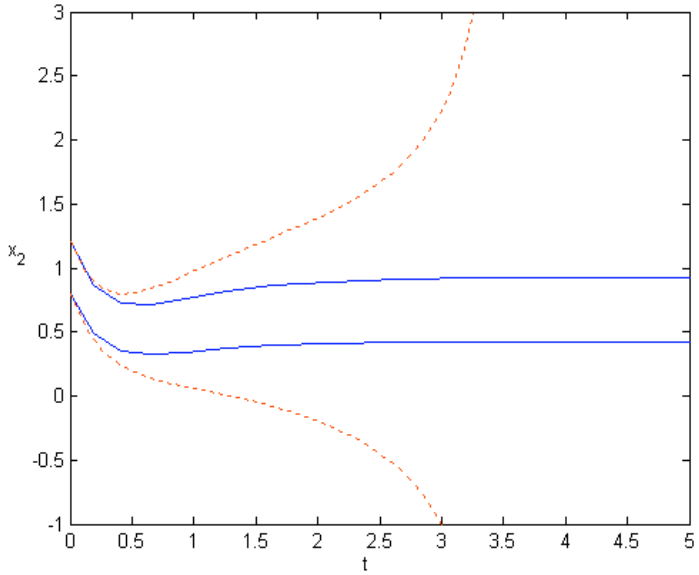


Fig. III.6 : *Encadrement de toutes les valeurs possibles de $x_2(t)$. Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle.*

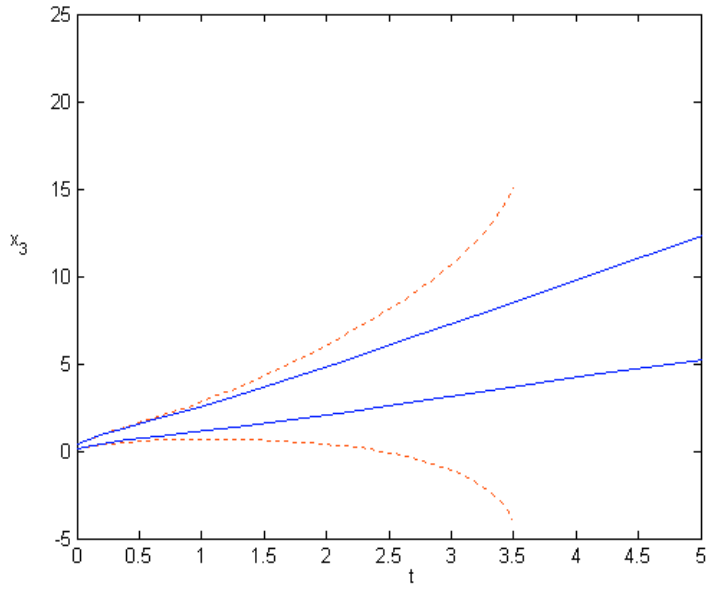


Fig. III.7 *: Encadrement de toutes les valeurs possibles de $x_3(t)$. Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle.*

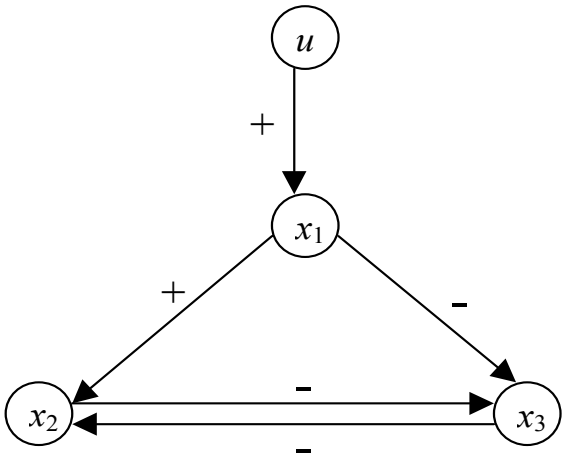


Fig. III.8 *: Graphe d'incidence de* (III.30)

Par ailleurs, en utilisant une matrice $\mathbf{D}$ de changement de base définie par

$$\mathbf{D} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}.$$

Le nouveau système (III.31), défini dans la nouvelle base de variables où $\mathbf{y} = \mathbf{D}\mathbf{x}$, est coopératif.

$$\begin{cases} \dot{y}_1(t) &= -2y_1 + u(t) \\ \dot{y}_2(t) &= 2y_1 + y_3 \\ \dot{y}_3(t) &= 2y_1 + y_2. \end{cases} \tag{III.31}$$

Il est évident que toutes les dérivées partielles $\frac{\partial f_i}{\partial y_j}, i \neq j$ sont de signes positifs. Evidemment, il est donc intéressant de résoudre numériquement l'EDO (III.30) dans cette nouvelle base de coordonnées en intégrant le système (III.31). Pour obtenir les solutions dans la base originale, nous utilisons la relation suivante

$$\mathbf{x}(t) = \mathbf{D}^{-1}\mathbf{y}(t).$$

Après avoir présenté comment réaliser l'intégration numérique garantie des EDO incertaines dans le cas des systèmes qui respectent une certaine relation d'ordre définie par rapport aux conditions initiales, nous consacrons la section suivante au cas général. Il s'agit donc de traiter le cas des EDO qui ne préservent aucune relation d'ordre. Dans la suite, nous appelons ces EDO, EDO non monotones.

III.4 Systèmes non monotones

Tout d'abord, avant de montrer comment construire les EDO englobantes dans le cas général, nous présentons le théorème de Müller [66, 99, 53, 54] qui définit les conditions permettant d'obtenir un encadrement garanti de toutes les solutions possibles d'une EDO incertaine non monotone. Dans cette section, nous ne considérons que des EDO avec des entrées certaines. Cela n'est pas une restriction de la méthode. En effet, nous pouvons ramener le problème d'entrées incertaines au problème de paramètres incertains.

III.4.1 Théorème de Müller

Contrairement au cas des EDO coopératives, pour lesquels les EDO englobantes inférieure et supérieure sont établies séparément grâce au principe de comparaison (Propriété 1), le cas des EDO non monotones nécessite la construction simultanée de ces deux EDO englobantes. Pour des raisons de simplicité, nous supposons que le vecteur d'excitation $\mathbf{u}(t)$ est connu sans incertitude. Dans la suite, nous donnons le théorème de Müller tel que formulé dans [53, 54, 95].

Théorème 2 [66, 54, 99] : *Considérons le système d'équations différentielles défini par*

$$\begin{cases} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}, \mathbf{p}, t) \\ \mathbf{x}_0 &\in [\mathbf{x}_0] \subseteq \mathbb{X} \\ \mathbf{p} &\in [\mathbf{p}] \end{cases} \tag{III.32}$$

où le champ de vecteur $\mathbf{f}$ *est continu sur un domaine* $\mathbb{T}$ *défini comme suite :*

$$\mathbb{T} : \begin{cases} \omega(t) \leq \mathbf{x}(t) \leq \Omega(t) \\ \underline{\mathbf{p}} \leq \mathbf{p} \leq \overline{\mathbf{p}} \\ t_0 \leq t \leq t_{nT}. \end{cases} \tag{III.33}$$

Les fonctions $\omega_i(t)$ *et* $\Omega_i(t)$ $i = 1, ..., n$ *sont continues sur l'intervalle* $[t_0, t_{nT}]$ *et satisfont les propriétés suivantes :*

1. $\omega(t_0) = \underline{\mathbf{x}}_0$ *et* $\Omega(t_0) = \overline{\mathbf{x}}_0$
2. *Les dérivées à gauche* $D^-\omega_i(t)$ *et* $D^-\Omega_i(t)$ *et les dérivées à droite* $D^+\omega_i(t)$ *et* $D^+\Omega_i(t)$ *de* $\omega_i(t)$ *et* $\Omega_i(t)$ *sont telles que*

$$\forall i,\ D^{\pm}\omega_i(t) \leq \min_{\underline{\mathbb{T}}} f_i(\mathbf{x}, \mathbf{p}, t) \tag{III.34}$$

$$\forall i,\ D^{\pm}\Omega_i(t) \leq \max_{\overline{\mathbb{T}}} f_i(\mathbf{x}, \mathbf{p}, t) \tag{III.35}$$

où $\underline{\mathbb{T}}$ *est un sous-domaine de* $\mathbb{T}$ *défini par*

$$\underline{\mathbb{T}}_i : \begin{cases} x_i = \omega_i(t) \\ \omega_j(t) \leq x_j \leq \Omega_j(t),\ j \neq i \\ \underline{\mathbf{p}} \leq \mathbf{p} \leq \overline{\mathbf{p}} \end{cases} \tag{III.36}$$

et $\overline{\mathbb{T}}$ *est aussi un sous-domaine de* $\mathbb{T}$ *défini par*

$$\overline{\mathbb{T}}_i : \begin{cases} x_i = \Omega_i(t) \\ \omega_j(t) \leq x_j \leq \Omega_j(t),\ j \neq i \\ \underline{\mathbf{p}} \leq \mathbf{p} \leq \overline{\mathbf{p}} \end{cases} \tag{III.37}$$

Alors, pour tous $\mathbf{x}_0 \in [\underline{\mathbf{x}}_0, \overline{\mathbf{x}}_0]$ *et* $\mathbf{p} \in [\underline{\mathbf{p}}, \overline{\mathbf{p}}]$, *le système* (III.32) *admet une solution* $\mathbf{x}(t)$ *incluse dans le domaine ci-dessous*

$$\mathbb{X} : \begin{cases} t_0 \leq t \leq t_{nT} \\ \omega(t) \leq x(t) \leq \Omega(t). \end{cases} \tag{III.38}$$

Comme dans le cas des systèmes quasi-monotones, l'unicité de la solution est assurée si la fonction $\mathbf{f}$ *est Lipschitzienne par rapport à* $\mathbf{x}$ *sur* $\mathbb{X}$ *[66, 63, 99].*

Preuve : Voir les références [98, 86].

III.4.2 Intégration numérique garantie dans le cas général

Comme nous l'avons présenté précédemment, l'intégration numérique garantie dans le cadre de la théorie de comparaison des inégalités différentielles consiste, en premier lieu, à transformer l'EDO initiale incertaines en deux EDO certaines englobantes. En second lieu, par le biais des méthodes de Taylor intervalles elle résout numériquement ces deux EDO certaines. Nous présentons dans la section suivante une règle qui permet de construire ces deux EDO englobantes dans le cas général. Pour cela, il est nécessaire de faire un couplage entre les deux EDO englobantes.

III.4.2.1 Approximations de solutions minimale et maximale

En pratique, la principale difficulté rencontrée dans le contexte du Théorème 2 est comment construire, dans le cas général, les deux EDO qui génèrent les approximations inférieure et supérieure respectivement des solutions minimale $\omega(t)$ et maximale $\Omega(t)$. Cependant, dans le cas où toutes les composantes du champs de vecteurs $\mathbf{f}$ sont monotones par rapport aux paramètres incertains et par rapport à toutes les composantes du vecteur de variables $\mathbf{x}$, nous réécrivons ci-dessous la règle de sélection des bornes de variables et paramètres incertains introduite par [53, 54]. En effet, cette règle est l'extension logique de la Règle 1 aux variables $x_i, i = 1, ..., n$.

Règle 2 : (Analyse de signes des dérivées partielles $\frac{\partial f_i}{\partial p_k}$ et $\frac{\partial f_i}{\partial x_j}$)

1. Pour tout $i \in \{1, ..., n\}$ et pour tout $k \in \{1, ..., r\}$, nous définissons les éléments $\overline{\theta}_i^k$ par :

$$\overline{\theta}_i^k = \begin{cases} \overline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(.,.,.) \geq 0 \ \forall \mathbf{x} \in \mathbb{X}, \forall \mathbf{p} \in [\mathbf{p}] \\ \underline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(.,.,.) < 0 \ \forall \mathbf{x} \in \mathbb{X}, \forall \mathbf{p} \in [\mathbf{p}] \end{cases} \tag{III.39}$$

ainsi le vecteur $\overline{\Theta} = [\overline{\theta}_1^1, ..., \overline{\theta}_n^1, ..., \overline{\theta}_n^r]^T$. De la même façon, nous obtenons les éléments $\underline{\theta}_i^k$ par

$$\underline{\theta}_i^k = \begin{cases} \underline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(.,.,.) \geq 0 \ \forall \mathbf{x} \in \mathbb{X}, \ \forall \mathbf{p} \in [\mathbf{p}] \\ \overline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(.,.,.) < 0 \ \forall \mathbf{x} \in \mathbb{X}, \ \forall \mathbf{p} \in [\mathbf{p}] \end{cases} \tag{III.40}$$

d'où le vecteur $\underline{\Theta} = [\underline{\theta}_1^1, ..., \underline{\theta}_n^1, ..., \underline{\theta}_n^r]^T$.

2. Pour tout $i \in \{1, ..., n\}$ et pour tout $j \in \{1, ..., n\}$, nous définissons les éléments $\overline{\gamma}_i^j$ du vecteur $\overline{\Gamma}$, de dimension inférieure ou égale à $n \times n$, par :

$$\overline{\gamma}_i^j = \begin{cases} \Omega_i \text{ si } i = j \\ \Omega_i \text{ si } (i \neq j) \wedge \frac{\partial f_i}{\partial x_j}(.,.,.) \geq 0 \ \forall \mathbf{x} \in \mathbb{X}, \ \forall \mathbf{p} \in [\mathbf{p}] \\ \omega_i \text{ si } (i \neq j) \wedge \frac{\partial f_i}{\partial x_j}(.,.,.) < 0 \ \forall \mathbf{x} \in \mathbb{X}, \ \forall \mathbf{p} \in [\mathbf{p}] \end{cases} \tag{III.41}$$

ainsi le vecteur $\overline{\Gamma} = [\overline{\gamma}_1^1, ..., \overline{\gamma}_1^n, ..., \overline{\gamma}_n^n]^T$. De manière similaire, nous construisons le vecteur $\underline{\Gamma}$ d'éléments $\underline{\gamma}_i^j$,

$$\underline{\gamma}_i^j = \begin{cases} \omega_i \text{ si } i = j \\ \omega_i \text{ si } \ (i \neq j) \wedge \frac{\partial f_i}{\partial x_j}(.,.,.) \geq 0 \ \forall \mathbf{x} \in \mathbb{X}, \ \forall \mathbf{p} \in [\mathbf{p}] \\ \Omega_j \text{ si } (i \neq j) \wedge \frac{\partial f_i}{\partial x_j}(.,.,.) < 0 \ \forall \mathbf{x} \in \mathbb{X}, \ \forall \mathbf{p} \in [\mathbf{p}] \end{cases} \tag{III.42}$$

d'où le vecteur $\underline{\Gamma} = [\underline{\gamma}_1^1, ..., \underline{\gamma}_1^n, ..., \underline{\gamma}_n^n]^T$.

En conséquence, les équations différentielles qui satisfont les conditions du théorème de Müller, sont données par :

$$i = 1, ..., n, \ \begin{cases} \dot{\omega}_i(t) &= \underline{f}_i(\underline{\Gamma}, \underline{\Theta}, t) \\ \dot{\Omega}_i(t) &= \overline{f}_i(\overline{\Gamma}, \overline{\Theta}, t). \end{cases} \tag{III.43}$$

Notons

$$\begin{aligned} \underline{f}_i(\omega, \Omega, \underline{\mathbf{p}}, \overline{\mathbf{p}}, t) &= \underline{f}_i(\underline{\Gamma}, \underline{\Theta}, t) \\ \overline{f}_i(\omega, \Omega, \underline{\mathbf{p}}, \overline{\mathbf{p}}, t) &= \overline{f}_i(\overline{\Gamma}, \overline{\Theta}, t). \end{aligned} \tag{III.44}$$

Ainsi, les fonctions $\omega(t)$ et $\Omega(t)$ sont les solutions du système d'équations différentielles couplées suivant :

$$\begin{cases} \dot{\omega}(t) = \underline{\mathbf{f}}(\omega, \Omega, \underline{\mathbf{p}}, \overline{\mathbf{p}}, t) \\ \dot{\Omega}(t) = \overline{\mathbf{f}}(\omega, \Omega, \underline{\mathbf{p}}, \overline{\mathbf{p}}, t) \\ \omega(t_0) = \underline{\mathbf{x}}_0 \\ \Omega(t_0) = \overline{\mathbf{x}}_0 \end{cases} \tag{III.45}$$

Ce système ne met en jeu que des variables et des paramètres sans incertitudes. Dès lors, l'utilisation des méthodes d'intégration garantie basées sur le développement en série de Taylor intervalles pour résoudre (III.45) permet de contrôler l'effet d'enveloppement et de fournir un encadrement garanti de toutes les solutions possibles de (III.32).

Remarques : La Règle 2 représente une généralisation de la Règle 1. Elle est donc applicable aussi pour le cas des équations différentielles monotones.

III.4.2.2 Exemples illustratifs

Pour illustrer la mise en œuvre de cette méthode d'intégration numérique garantie dans le cas des systèmes non monotones, nous considérons dans cette section deux exemples dans lesquels nous montrons comment construire les EDO englobantes et nous présentons les résultats de leur intégration numérique.

Exemple 1 : Soit le système dynamique

$$\begin{cases} \dot{x}_1 &= p_1 x_2 - p_2 u(t) \\ \dot{x}_2 &= -x_2 - 2p_1 x_2 \exp(x_1). \end{cases} \tag{III.46}$$

Où le premier paramètre est incertain $p_1 \in [0.4, 0.5]$, et le second est connu $p_2 = 0.45$. Le pavé de conditions initiales est $[\mathbf{x}_0] = [-4, -3] \times [12, 14]$ et l'entrée $u(t)$ est donnée par

$$u(t) = 4\sin(3t).$$

Ce système n'est pas préservateur d'ordre comme le montre son graphe d'incidence présenté dans la figure III.9. En effet, ce graphe contient un cycle de signe négatif.

Donc pour résoudre ce problème, nous utilisons la Règle 2 qui satisfait les conditions de Théorème 2 pour construire les EDO englobantes couplées suivantes :

$$\begin{cases} \dot{\underline{x}}_1 &= \underline{p}_1 \underline{x}_2 - p_2 u(t) \\ \dot{\underline{x}}_2 &= -\underline{x}_2 - 2\overline{p}_1 \underline{x}_2 \exp(\overline{x}_1) \\ \dot{\overline{x}}_1 &= \overline{p}_1 \overline{x}_2 - p_2 u(t) \\ \dot{\overline{x}}_2 &= -\overline{x}_2 - 2\underline{p}_1 \overline{x}_2 \exp(\underline{x}_1). \end{cases} \tag{III.47}$$

Par conséquent, l'intégration numérique garantie du système couplé (III.47), par le biais des méthodes basées sur les modèles de Taylor, permet d'obtenir simultanément des approximations inférieure et supérieure des solutions minimale et maximale de (III.46). De

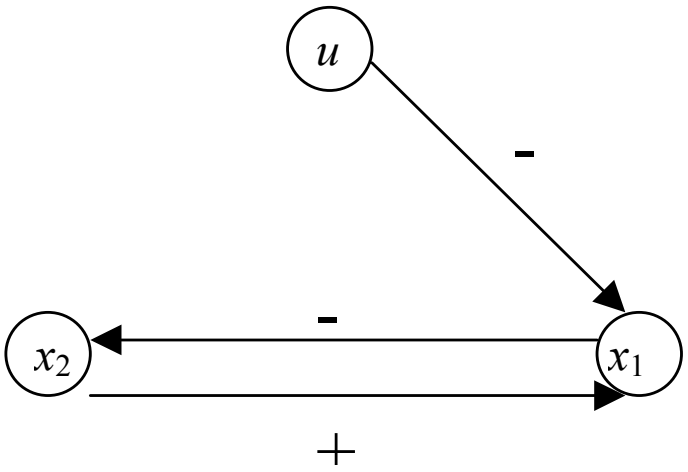


Fig. III.9 *: Graphe d'incidence de* (III.46)

plus, comme (III.47) ne met en jeu que des variables et des paramètres certains, la propagation du pessimisme due à l'effet d'enveloppement dans les modèles de Taylor intervalles est négligeable. Sur les figures III.10 et III.11 nous avons tracé en lignes bleues continues les encadrements de toutes les solutions possibles de (III.46) obtenus par l'application du théorème de Müller. Par ailleurs, pour montrer l'incapacité des méthodes d'intégration basées sur les modèles de Taylor intervalles appliquées directement au système (III.46), nous présentons sur ces mêmes figures en lignes rouges pointillés les encadrements de solutions obtenus en appliquant la méthode de Hermite-Obreschkoff intervalle à l'EDO incertaine (III.46). Ainsi, on voit bien que les courbes en rouges pointillés divergent sous l'effet d'enveloppement au bout de quelques pas d'intégration.

Dans l'exemple suivant, nous allons présenter un cas d'EDO où les méthodes d'intégration basées sur les modèles de Taylor intervalles et les méthodes basées sur le théorème de Müller donnent des résultats similaires.

Exemple 2 : Considérons le système d'équations différentielles linéaires avec un vecteur d'entrée incertain suivant :

$$\begin{cases} \dot{x}_1 &= -x_1 + u_1(t) \\ \dot{x}_2 &= -x_1 - x_2 + x_3 - u_2(t) \\ \dot{x}_3 &= -x_1 + x_2 - x_3, \end{cases} \tag{III.48}$$

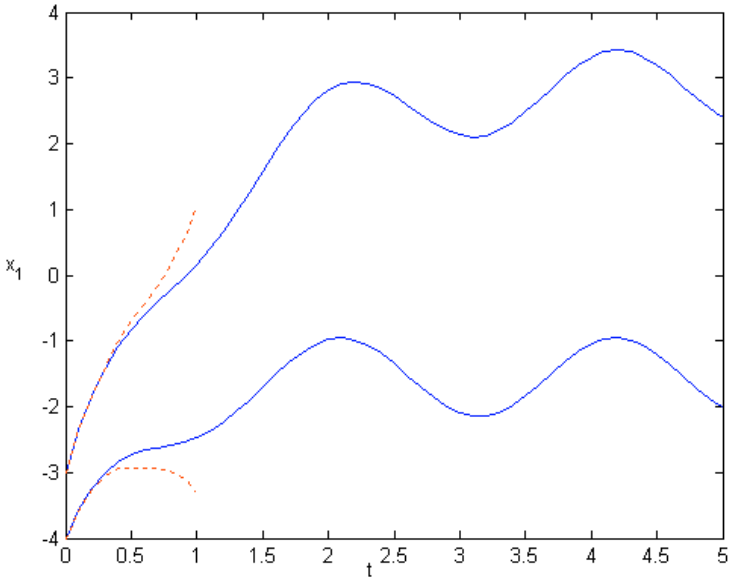


Fig. III.10 : *Encadrement de toutes les valeurs possibles de $x_1(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle.*

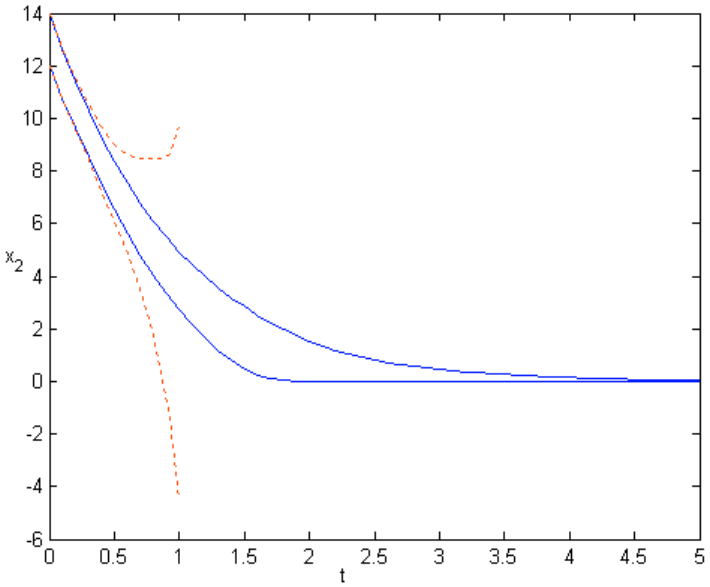


Fig. III.11 : *Encadrement de toutes les valeurs possibles de $x_2(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle.*

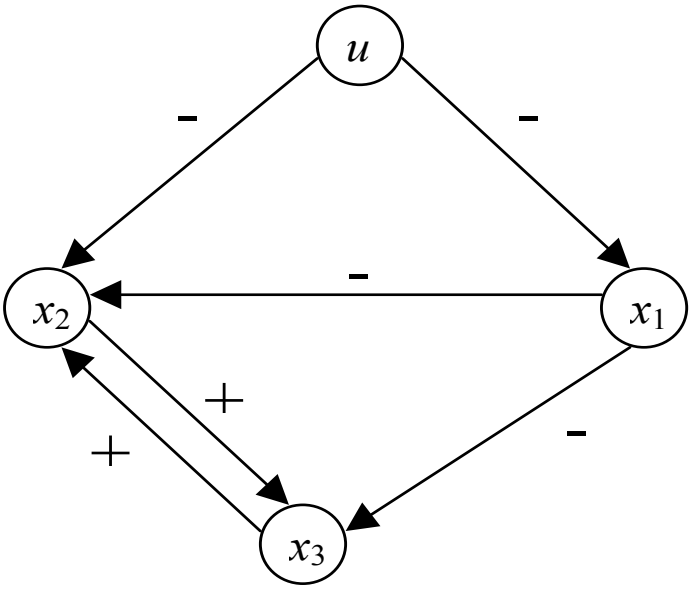


Fig. III.12 *: Graphe d'incidence de* (III.48)

où les conditions initiales sont incertaines mais bornées et de bornes connues *a priori* $[\mathbf{x}_0] = [-3, 1] \times [1, 2] \times [2, 3]$. De même, le vecteur d'entrée $\mathbf{u}(t)$ est borné entre

$$\begin{array}{rclcccl} \underline{u}_1(t) & = & \frac{\underline{p}_1}{3+\sin(t)} & \leq & u_1(t) & \leq & \frac{\overline{p}_1}{3+\sin(t)} & = & \overline{u}_1(t) \\ \underline{u}_2(t) & = & \underline{p}_2(2+\sin(t)) & \leq & u_2(t) & \leq & \overline{p}_2(2+\sin(t)) & = & \overline{u}_2(t) \end{array} \tag{III.49}$$

où $[p_1] = [1.4, 1.7]$ et $[p_2] = [2.4, 2.6]$.

D'après le graphe d'incidence de III.48 présenté dans la figure III.12, ce système n'est pas préservateur d'ordre. Donc, il n'existe pas de matrice de changement de base permettant de réécrire III.48 sous forme coopérative dans la nouvelle base de coordonnées. Cependant, en utilisant la Règle 2, nous pouvons construire deux systèmes englobants couplés (III.50) qui vérifieront les conditions imposées par le Théorème 2.

$$\left\{ \begin{array}{rcl} \dot{\underline{x}}_1 & = & -\underline{x}_1 + \underline{u}_1(t) \\ \dot{\underline{x}}_2 & = & -\overline{x}_1 - \underline{x}_2 + \underline{x}_3 - \overline{u}_2(t) \\ \dot{\underline{x}}_3 & = & -\overline{x}_1 + \underline{x}_2 - \underline{x}_3, \\ \dot{\overline{x}}_1 & = & -\overline{x}_1 + \overline{u}_1(t) \\ \dot{\overline{x}}_2 & = & -\underline{x}_1 - \overline{x}_2 + \overline{x}_3 - \underline{u}_2(t) \\ \dot{\overline{x}}_3 & = & -\underline{x}_1 + \overline{x}_2 - \overline{x}_3 \end{array} \right. \tag{III.50}$$

En simulant le système (III.48) sur une durée $t \in [0, 10]$ une fois par le biais de l'intégration numérique garantie de (III.50) et une autre fois en l'intégrant directement

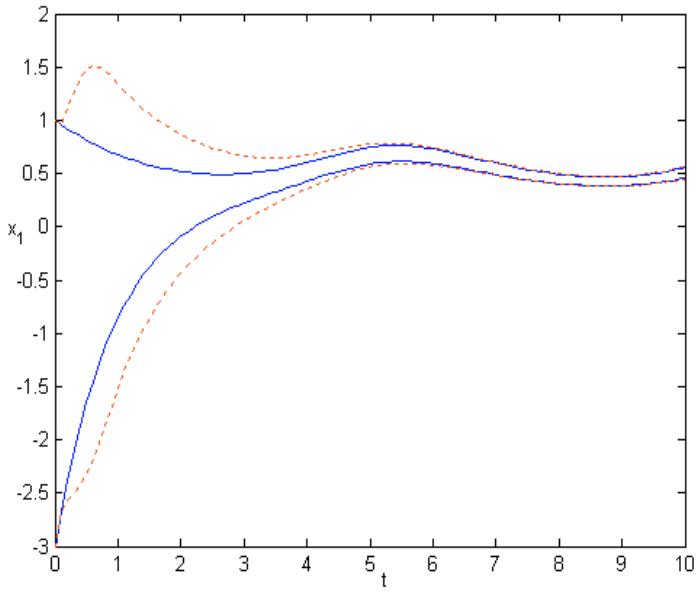


Fig. III.13 *: Encadrement de toutes les valeurs possibles de $x_1(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle.*

par la méthode de Hermite-Obroschkoff intervalle, nous caractérisons dans chaque cas l'ensemble de toutes ses solutions possibles sur cette durée. Effectivement, dans les figures III.13, III.14 et III.15 les courbes en bleu continu représentent respectivement les intervalles solutions $[x_1]_M(t)$, $[x_2]_M(t)$ et $[x_3]_M(t)$ obtenus en résolvant le système couplé (III.50). Cependant, les courbes en rouge discontinu correspondent respectivement aux intervalles solutions $[x_1]_T(t)$, $[x_2]_T(t)$ et $[x_3]_T(t)$ calculés par la méthode de Hermite-Obreschkoff intervalle appliquée à (III.48). Pour cet exemple, l'inclusion ci-dessous est toujours satisfaite :

$$\forall t \in [0,10] \begin{cases} [x_1]_M(t) & \subseteq & [x_1]_T(t) \\ [x_2]_M(t) & \subseteq & [x_2]_T(t) \\ [x_3]_M(t) & \subseteq & [x_3]_T(t). \end{cases}$$

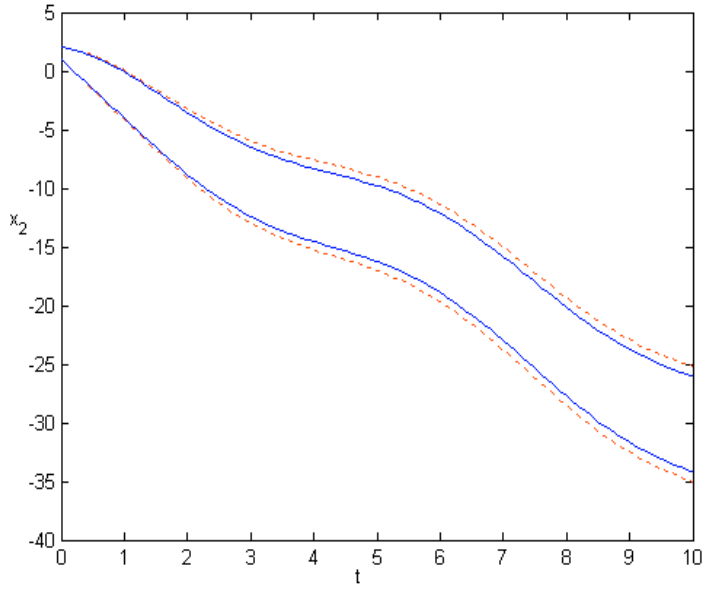


Fig. III.14 *: Encadrement de toutes les valeurs possibles de $x_2(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle.*

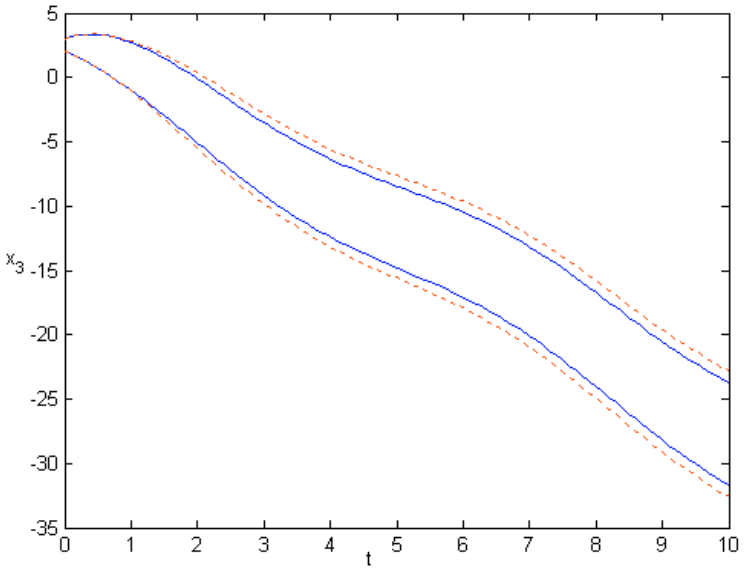


Fig. III.15 *: Encadrement de toutes les valeurs possibles de $x_3(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle.*

III.5 Quelques remarques concernant la précision des encadrements de solutions

Comme nous l'avons montré, les méthodes d'intégration numérique garanties des EDO incertaines par le biais des théorèmes de comparaison des inégalités différentielles présentées dans les sections III.3 et III.4, permettent de calculer des encadrements garanti de toutes les solutions possibles des EDO incertaines. Cependant, la précision de ces encadrements de solutions dépend de la nature des EDO étudiées. En effet, nous regroupons ces EDO dans trois cas :

1. Cas des EDO coopératives dans lesquels toutes les dérivées partielles par rapport à chaque paramètre sont de même signe : Dans ce contexte, les EDO englobantes (III.18) et (III.19) correspondent à des EDO réalisables. Donc il n'y a pas de pessimisme. Par conséquent, nous obtenons un encadrement exact de l'ensemble de toutes les solutions possibles.

2. Cas des EDO coopératives où au moins une des dérivées partielles par rapport à un paramètre n'est pas de même signe que les autres : Dans ce cas, les EDO représentées par (III.18) et (III.19) ne sont pas nécessairement réalisables, mais correspondent bien à deux EDO parmi toutes celles obtenues en effectuant le calcul par les méthodes de Taylor intervalles. Donc, nous affirmons que $\forall[\mathbf{x}]$, $\forall[\mathbf{p}]$ l'inclusion ci-dessous est vraie

$$[\mathbf{x}_{Cop}](t) \subseteq [\mathbf{x}_T](t)\ \forall t \geq t_0,$$

où $[\mathbf{x}_{Cop}](t)$ représente l'encadrement des solutions de (III.16) obtenu par l'utilisation du Théorème 1 et $[\mathbf{x}_T](t)$ représente l'encadrement de ces même solutions obtenu par les méthodes basées sur les modèles de Taylor intervalles ou le polynôme de Hermite-Obreschkoff intervalle. Les résultats obtenus par l'application du Théorème 1 seront au moins aussi bons que ceux obtenus par les méthodes basées sur les modèles de Taylor intervalles.

3. Cas des EDO non monotones : Dans ce contexte, pour certaines EDO le couplage entre les dynamiques minimale et maximale conduit à des EDO couplés instables même si l'EDO initiale est stable. Donc, les méthodes basées sur les modèles de Taylor intervalles avec des stratégies de bissection de l'espace d'état et celui des

paramètres peuvent donner de meilleurs résultats. Dans le cas général, les résultats fournis par le théorème de Müller et ceux obtenus par les méthodes de Taylor intervalle peuvent être utilisés de manière collaborative.

III.6 Conclusion

Pour pouvoir remédier à la propagation du pessimisme dû à l'effet d'enveloppement dans les schémas d'intégration numérique garantie des EDO, nous avons présenté dans ce chapitre des approches d'intégration garantie fondées sur la théorie de comparaison des inégalités différentielles monotones et non monotones. Dans ce contexte, nous avons présenté selon le cas deux règles de construction des EDO déterministes, inférieure et supérieure, dont les solutions englobent toutes les solutions possibles de l'équation différentielle incertaine initiale. Cependant, l'application de ces deux règles telles qu'elles sont présentées dans ce chapitre est limitée aux cas des équations différentielles dont toutes les dérivées partielles de $\mathbf{f}$ par rapport à chaque paramètre $\frac{\partial f_i}{\partial p_k}$ et par rapport à chaque variable $\frac{\partial f_i}{\partial x_j}$ sont de signe constant sur toute la période d'intégration. Ceci a pour effet de réduire fortement leur capacité à résoudre ce problème d'encadrement dans la majorité des cas pratiques. Pour généraliser l'application de ces deux règles nous proposons dans le chapitre suivant une approche hybride d'encadrement de solutions d'équations différentielles incertaines mettant en jeu un automate hybride.

Chapitre IV

Approche hybride pour l'encadrement des solutions d'EDO incertaines

IV.1 Introduction

Ce quatrième chapitre représente la première contribution de la thèse à l'intégration numérique garantie des équations différentielles incertaines. Généralement, dans le contexte de l'intégration numérique garantie par le biais de la théorie de comparaison des inégalités différentielles, on est souvent confrontés à des situations où les conditions exigées par les règles **Règle 1** et **Règle 2** présentées dans le chapitre III, ne sont pas satisfaites sur toute la période de simulation. Mais, ces conditions sont satisfaites localement sur plusieurs intervalles de temps qui constituent la période totale de simulation. Dès lors, nous proposons de modéliser les systèmes englobants sous la forme d'automates hybrides continu-discret [2, 1]. Il s'agit alors de transformer l'équation différentielle incertaine en un ensemble d'équations différentielles certaines gouvernées par un automate d'état fini et dont l'exécution génère les trajectoires solutions minimale et maximale. Dans la suite, nous présentons dans le détail cette approche hybride d'encadrement en montrant comment construire les modes de fonctionnement continu de l'automate hybride et les conditions de garde définissant les domaines invariants puis comment franchir les frontières de ces domaines sans perdre la garantie de l'encadrement. Nous présenterons également des algorithmes qui résument cette approche.

Nous commençons tout d'abord par présenter l'automate hybride d'encadrement dans

le cas général, c'est-à-dire le cas des équations différentielles qui ne possèdent pas de propriétés particulières (monotonie . . .). Puis, nous montrons comment cet automate peut aussi traiter le cas des EDO monotones.

IV.2 L'automate hybride d'encadrement

Nous définissons l'automate hybride dont l'exécution engendre un encadrement garanti de toutes les solutions possibles de (III.32) comme étant un Sextuple

$$\mathcal{H} = (\mathbb{Q}, \mathbb{X}, \mathbb{P}, \mathbb{F}, \mathbb{T}, \mathcal{RI}) \tag{IV.1}$$

où ;

1. $\mathbb{Q}$ est un ensemble discret de modes ou de situations. Pour chaque mode correspond un système d'équations différentielles couplées de la forme (III.45) qui génère localement les solutions minimale et maximale de (III.32) sur des intervalles de temps bien définis. Ces EDO couplées sont établies par l'utilisation de la Règle 2.
2. $\mathbb{X}$ est l'espace des variables de (III.32) inclus dans $\mathbb{R}^n$.
3. $\mathbb{P} = [\mathbf{p}] = [\underline{\mathbf{p}}, \overline{\mathbf{p}}]$ représente le domaine admissible borné de paramètres de (III.32).
4. $\mathbb{F} = \{(\underline{\mathbf{f}}_q, \overline{\mathbf{f}}_q), q \in \mathbb{Q}, \mathbf{f}\}$ est la collection du champs de vecteurs qui encadrent toutes les dynamiques possibles de (III.32)

$$\forall q \in \mathbb{Q}, \quad \left\{ \begin{array}{lcl} \underline{\mathbf{f}}_q & : & \mathbb{X}^2 \times \mathbb{P}^2 \longrightarrow \mathbb{R}^n \\ \overline{\mathbf{f}}_q & : & \mathbb{X}^2 \times \mathbb{P}^2 \longrightarrow \mathbb{R}^n \end{array} \right. \tag{IV.2}$$

5. $\mathbb{T} = \{t_e, e \in \mathbb{E}\}$ est la collection des instants de transitions, où $\mathbb{E} \subseteq \mathbb{Q} \times \mathbb{Q}$ est l'ensemble des transitions. Il contient toutes les commutations possibles entre les EDO couplées qui englobent localement (III.32). On définit $g_{i,l}(.) = \frac{\partial f_i}{\partial p_l}(.)$ et $h_{i,k}(.) = \frac{\partial f_i}{\partial x_k}(.)$ où $i \in \{1, ..., n\},\ l \in \{1, ..., r\}$ et $k \in \{1, ..., n\}$. L'ensemble $\mathbb{T}$ est défini par

$$\mathbb{T} = \left\{ \begin{array}{c} t_e \in [t_0, t_{nT}] \mid \exists i,\ \exists k,\ \exists l, \mid \\ (0 \in [g]_{i,l}([\mathbf{x}](t_e), [\mathbf{p}], t_e)) \vee (0 \in [h]_{i,k}([\mathbf{x}](t_e), [\mathbf{p}], t_e)) \end{array} \right\} \tag{IV.3}$$

Ceci signifie que si la monotonie de $\mathbf{f}$ par rapport à un paramètre ou par rapport à une variable change de sens à l'instant t_e, une transition $e = (q, q') \in \mathbb{E}$ a lieu et provoque un changement de l'EDO couplée englobante.

6. $\mathcal{RI} = \{\mathbf{R}_e, e \in \mathbb{E}\}$ est la collection des fonctions de mise à jour. Elles permettent d'initialiser les champs de vecteurs $\overline{\mathbf{f}}_{q'}$ (resp. $\underline{\mathbf{f}}_{q'}$) après que la transition $e = (q, q')$ ait lieu :

$$\{\underline{\mathbf{x}}_{q'}(t_e), \overline{\mathbf{x}}_{q'}(t_e)\} = \mathbf{R}_e(\underline{\mathbf{x}}_q(t_e), \overline{\mathbf{x}}_q(t_e)).$$

Donc, pour étendre l'application de la Règle 2, nous décomposons la durée de la simulation globale $[t_0, t_{nT}]$ en une suite d'intervalles $[t_j, t_{j+1}]$, avec $t_{j+1} = t_j + h_j$, où le pas d'intégration h_j est choisi *a priori* ou adapté au cours de l'intégration en utilisant par exemple une stratégie de pas d'intégration variable [69, 72].

Notons par $\mathbb{I}_M$ l'ensemble des intervalles de temps $[t_j, t_{j+1}]$ sur lesquels il n'y a pas de transitions, c'est-à-dire que les fonctions $g_{i,l}(.)$ et $h_{i,k}(.)$ ne s'annulent pas, et donc toutes les composantes du champ de vecteurs $\mathbf{f}$ de (III.32) sont monotones par rapport à chaque paramètre et par rapport à chaque variable. Cet ensemble est donné par la relation suivante :

$$\mathbb{I}_M = \{[t_j,\, t_{j+1}] \subseteq [0, t_{nT}] \,|\, \forall e \in \mathbb{E},\, t_e \notin [t_j,\, t_{j+1}]\} \tag{IV.4}$$

De plus, puisque par définition la solution *a priori* $[\tilde{\mathbf{x}}_j]$ encadre toutes les trajectoires solutions de (III.32) sur l'intervalle $[t_j, t_{j+1}]$, alors nous pouvons écrire un encadrement *intérieur* de l'ensemble (IV.4) sans perdre la garantie de la manière suivante :

$$\underline{\mathbb{I}}_M \equiv \left\{ \begin{array}{c} [t_j,\, t_{j+1}] \subseteq [0, t_{nT}] \,| \\ \forall i,\ \forall l,\ \forall k, \\ ((0 \notin [g]_{i,l}([\tilde{\mathbf{x}}_j], [\mathbf{p}], [t_j, t_{j+1}]) \vee (0 \notin [h]_{i,k}([\tilde{\mathbf{x}}_j], [\mathbf{p}], [t_j, t_{j+1}])) \end{array} \right\} \subseteq \mathbb{I}_M \tag{IV.5}$$

De même, nous définissons l'ensemble $\mathbb{I}_S$ comme étant l'ensemble de tous les intervalles $[t_j, t_{j+1}]$ de taille égale à la largeur du pas d'intégration courant, où au moins une condition de garde est satisfaite, c'est-à-dire

$$\mathbb{I}_S = \{[t_j,\, t_{j+1}] \subseteq [0, t_{nT}] \,|\, \exists e \in \mathbb{E},\, t_e \in [t_j,\, t_{j+1}]\}\,. \tag{IV.6}$$

Par ailleurs, comme la propriété suivante est vérifiée

$$[t_0, t_{nT}] = \mathbb{I}_M \cup \mathbb{I}_S \tag{IV.7}$$

alors, nous pouvons en déduire que :

$$\overline{\mathbb{I}}_S = [t_0, t_{nT}] \setminus \underline{\mathbb{I}}_M. \tag{IV.8}$$

Après cette partition de la durée globale de la simulation entre $\underline{\mathbb{I}}_M$ et $\overline{\mathbb{I}}_S$, nous pouvons appliquer la Règle 2 sur chaque intervalle $[I_m] \in \underline{\mathbb{I}}_M$ pour construire les champs de vecteurs couplés $\underline{\mathbf{f}}_m$ et $\overline{\mathbf{f}}_m$ encadrants localement toutes les dynamiques possibles de l'EDO incertaine (III.32)

$$\begin{gathered} \forall [I_m] \in \underline{\mathbb{I}}_M,\ \forall m \in \mathbb{Q},\ \forall \mathbf{p} \in [\mathbf{p}],\ \forall \mathbf{x} \in \mathbb{X}, \\ \forall t \in [I_m],\ \ \underline{\mathbf{f}}_m(\omega, \Omega, \underline{\mathbf{p}}, \overline{\mathbf{p}}, t) \leq \mathbf{f}(\mathbf{x}, \mathbf{p}, t) \leq \overline{\mathbf{f}}_m(\omega, \Omega, \underline{\mathbf{p}}, \overline{\mathbf{p}}, t) \end{gathered} \tag{IV.9}$$

avec $(\underline{\mathbf{f}}_m, \overline{\mathbf{f}}_m) \in \mathbb{F}$.

Cependant, sur les intervalles $[I_s] \in \overline{\mathbb{I}}_S$ la **Règle 2** n'est pas applicable pour les deux raisons suivantes : (i) les méthodes d'intégration que nous utilisons sont numériques, alors les instants de commutation ne sont pas connus exactement. En revanche, il est facile de déterminer les pas d'intégration $[t_j, t_j + h_{j+1}] \in \overline{\mathbb{I}}_S$ qui les contiennent ; (ii) comme les incertitudes dans (III.32) sont représentées par des pavés, alors il existe des intervalles de temps $[I_s] \in \overline{\mathbb{I}}_S$ dans lesquels au moins un des signes de dérivées partielles $h_{i,k}(.)$ ou $g_{i,l}(.)$ est ambigu, c'est-à-dire $0 \in [h]_{i,k}(.)$ ou $0 \in [g]_{i,l}(.)$. Pour lever cette ambiguïté et franchir les conditions de garde qui provoquent les transitions sans perdre la garantie de l'encadrement, l'automate hybride $\mathcal{H}$ réalise directement l'intégration de l'EDO (III.32) sur ces intervalles par une des méthodes d'intégration garantie basées sur les modèles de Taylor intervalles ou le polynôme de Hermite-Obreschkoff intervalle.

IV.3 Exemple illustratif

Considérons un système d'équations différentielles de dimension 2 avec un paramètre incertain p_1 donnée par :

$$\begin{cases} \dot{x}_1 = f_1(x_1, x_2, p_1, t) \\ \dot{x}_2 = f_2(x_1, x_2, t) \\ \mathbf{x}(t_0) \in [\underline{x}_1, \overline{x}_1](t_0) \times [\underline{x}_2, \overline{x}_2](t_0) \\ p_1 \in [\underline{p}_1, \overline{p}_1]. \end{cases} \tag{IV.10}$$

Supposons que les courbes présentées sur la figure IV.1 montrent toutes les évolutions temporelles possibles de $g_{1,1}(.) = \frac{\partial f_1}{\partial p_1}(.)$, $h_{1,2}(.) = \frac{\partial f_1}{\partial x_2}(.)$ et de $h_{2,1}(.) = \frac{\partial f_2}{\partial x_1}(.)$ sur une période de temps finie. Selon les signes de ces dérivées partielles l'EDO (IV.10) admet

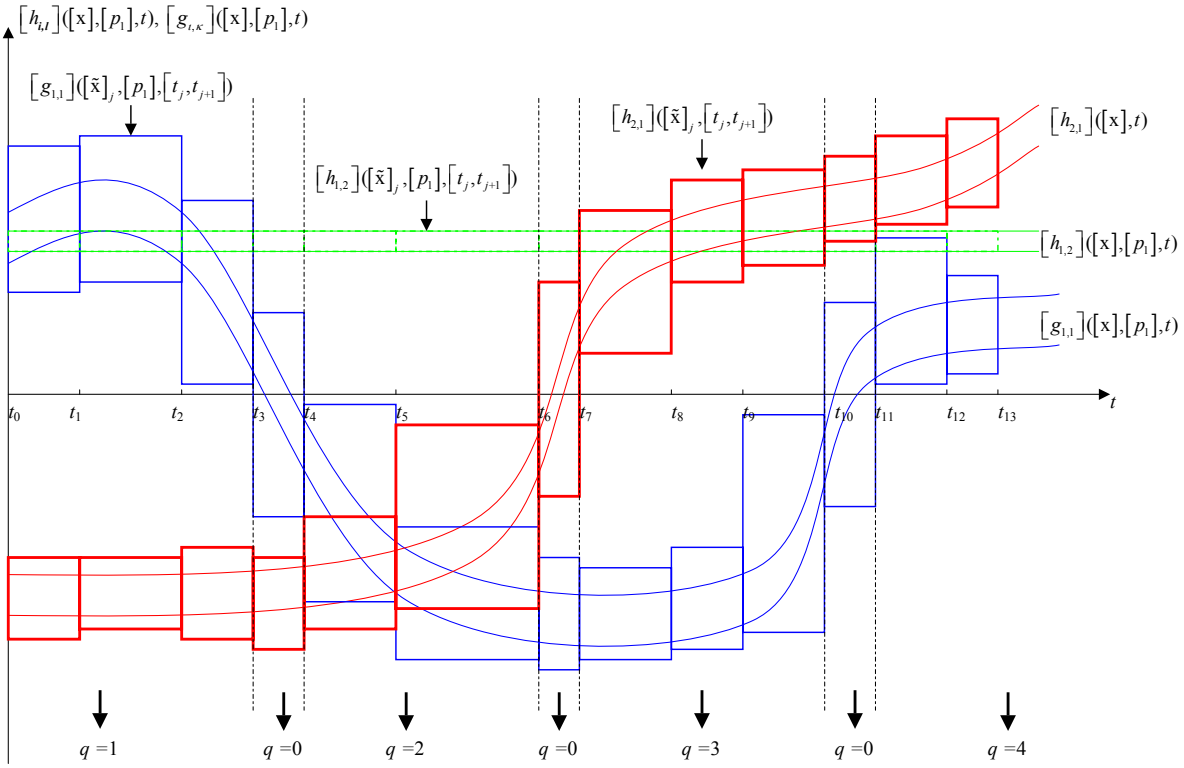


Fig. IV.1 *: Les différents modes d'encadrement possibles de* (IV.10)

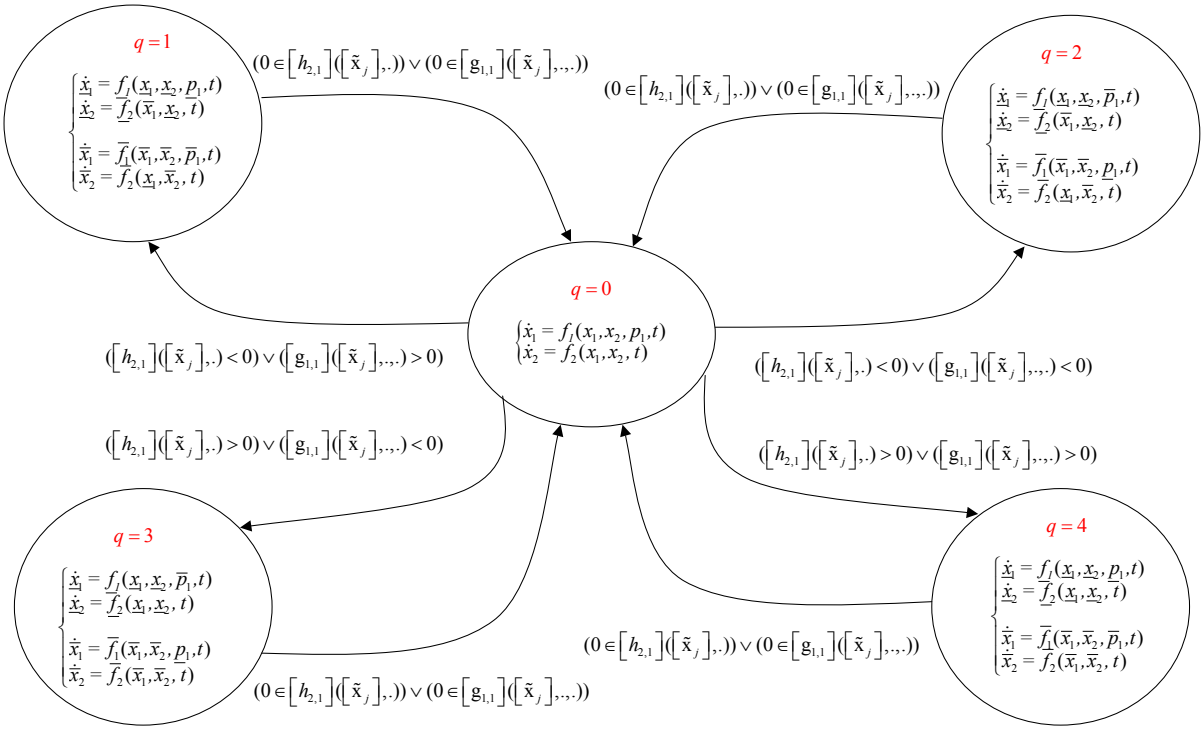


Fig. IV.2 *: Automate hybride*

globalement quatre EDO englobantes sans aucune ambiguïté qui représentent les modes de fonctionnement continu de l'automate hybride $\mathcal{H}$:

$$q=1\begin{cases}\dot{\underline{x}}_1=\underline{f}_1(\underline{x}_1,\underline{x}_2,\underline{p}_1,t)\\ \dot{\underline{x}}_2=\underline{f}_2(\overline{x}_1,\underline{x}_2,t)\\ \dot{\overline{x}}_1=\overline{f}_1(\overline{x}_1,\overline{x}_2,\overline{p}_1,t)\\ \dot{\overline{x}}_2=\overline{f}_2(\underline{x}_1,\overline{x}_2,t)\end{cases}\qquad q=2\begin{cases}\dot{\underline{x}}_1=\underline{f}_1(\underline{x}_1,\underline{x}_2,\overline{p}_1,t)\\ \dot{\underline{x}}_2=\underline{f}_2(\overline{x}_1,\underline{x}_2,t)\\ \dot{\overline{x}}_1=\underline{f}_1(\overline{x}_1,\overline{x}_2,\overline{p}_1,t)\\ \dot{\overline{x}}_2=\overline{f}_2(\underline{x}_1,\overline{x}_2,t)\end{cases}$$

$$q=3\begin{cases}\dot{\underline{x}}_1=\underline{f}_1(\underline{x}_1,\underline{x}_2,\overline{p}_1,t)\\ \dot{\underline{x}}_2=\underline{f}_2(\underline{x}_1,\underline{x}_2,t)\\ \dot{\overline{x}}_1=\overline{f}_1(\overline{x}_1,\overline{x}_2,\underline{p}_1,t)\\ \dot{\overline{x}}_2=\overline{f}_2(\overline{x}_1,\overline{x}_2,t)\end{cases}\qquad q=4\begin{cases}\dot{\underline{x}}_1=\underline{f}_1(\underline{x}_1,\underline{x}_2,\underline{p}_1,t)\\ \dot{\underline{x}}_2=\underline{f}_2(\underline{x}_1,\underline{x}_2,t)\\ \dot{\overline{x}}_1=\overline{f}_1(\overline{x}_1,\overline{x}_2,\overline{p}_1,t)\\ \dot{\overline{x}}_2=\overline{f}_2(\overline{x}_1,\overline{x}_2,t).\end{cases}\tag{IV.11}$$

Ces EDO sont construites par l'application locale de la **Règle 2** et leurs solutions encadrent localement de manière garantie toutes les solutions possibles de (IV.10). Par exemple, l'EDO englobante couplée qui correspond au mode $q=2$ n'est valable que sur l'intervalle de temps $[t_4,t_6]$.

Par contre, on voit sur la figure IV.1 qu'il existe des intervalles de temps où ces dérivées partielles changent de signe. Il s'agit des intervalles où au moins l'une des fonctions d'inclusion $[h_{2,1}]([\tilde{\mathbf{x}}_j],.)$ ou $[g_{1,1}]([\tilde{\mathbf{x}}_j],.,.)$ contient 0, comme par exemple sur l'intervalle $[t_6,t_7]$ on a $0\in[h_{2,1}]([\tilde{\mathbf{x}}_j],[t_6,t_7])$. Donc, pour ne pas perdre la garantie de l'encadrement sur ces intervalles, nous intégrons directement l'EDO incertaine (IV.10) sur ces intervalles par une des méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalles. Ainsi, nous introduisons un mode intermédiaire $q=0$ dans l'automate hybride dont le rôle est d'assurer le franchissement garanti des conditions de garde.

Finalement, la décomposition de la période totale $[T]=[t_0,t_{13}]$ de simulation de l'EDO (IV.10) en des intervalles $[I_m]$ et $[I_s]$ appartenants respectivement aux ensembles $\underline{\mathbb{I}}_M$ et $\overline{\mathbb{I}}_S$ est donnée par

$$\underline{\mathbb{I}}_M=\{[t_0,t_3],[t_4,t_6],[t_7,t_{10}],[t_{11},t_{13}]\}$$
$$\overline{\mathbb{I}}_S=\{[t_3,t_4],[t_6,t_7],[t_{10},t_{11}]\}.$$

Avec $[T]=\underline{\mathbb{I}}_M\cup\overline{\mathbb{I}}_S$.

Comme conclusion, l'automate hybride qui génère un encadrement garanti de toutes les solutions possibles du (IV.10) est constitué de 1+4 modes : le mode 0 correspond

à l'utilisation de méthodes de Taylor intervalles avec le modèle incertain original et les modes de 1 à 4 correspondent aux EDO englobantes résolues à l'aide des méthodes de Taylor intervalles. Donc, l'ensemble des états discrets de $\mathcal{H}$ est $\mathbb{Q} = \{0, 1, 2, 3, 4\}$. Cet automate est représenté dans la figure IV.2 qui indique les modes et les conditions de commutation.

Remarque : Dans le cas général, le nombre d'états discrets de l'automate hybride d'encadrement $\mathcal{H}$ est lié au nombre de dérivées partielles qui changent de signe au cours de la simulation garantie du système incertain. En effet, pour m dérivées partielles qui changent de signe, nous avons $2^m + 1$ états discrets possibles.

Dans la suite, nous désignerons toujours par $q = 0$ l'EDO incertaine initiale et par $q > 0$ chaque une des EDO englobantes.

IV.4 Déroulement de l'automate hybride

L'évolution discrète de l'automate hybride $\mathcal{H}$ est gouvernée par les instants de transitions $t_e \in \mathbb{T}$. En pratique, comme nous l'avons déjà mentionné, ces instants ne sont pas connus exactement. Mais généralement, il est facile de détecter les pas d'intégration qui les contiennent. Par conséquent, pour ne pas perdre la garantie du calcul, le passage d'un mode $q \neq 0$ à un autre mode $q' \neq 0$ se réalise par l'intermédiaire du mode $q = 0$. Les deux propositions ci-dessous montrent comment détecter et franchir les commutations entre les modes $q \neq 0 \mapsto q' = 0$ et $q = 0 \mapsto q' \neq 0$.

Proposition 1 : *(Commutation $q \neq 0 \mapsto q' = 0$) :*

$$\begin{aligned} Si((q \neq 0) \wedge (\exists i,\, \exists l,\, \exists k,\, (0 \in [g]_{i,k}([\tilde{\mathbf{x}}_j], [\mathbf{p}], [t_j,\, t_{j+1}])) \vee (0 \in [h]_{i,l}([\tilde{\mathbf{x}}_j], [\mathbf{p}], [t_j,\, t_{j+1]}])))) \\ \Rightarrow ((e = (q, q') \wedge (q' = 0)) \wedge ([\tilde{\mathbf{x}}_j]\ \textit{doit être recalculée}) \end{aligned} \tag{IV.12}$$

Preuve :

Le mode courant est $q \neq 0$. Lorsque l'une des dérivées partielles $g_{i,k}(.)$ ou $h_{i,l}(.)$ change de signe à un instant $t_e \in [t_j, t_{j+1}]$, une transition apparaît et le nouveau mode est nécessairement $q' = 0$. En effet, la dérivée partielle qui a provoqué la transition change de signe sur l'intervalle $[t_j, t_{j+1}]$. Etant donné que la solution *a priori* $[\tilde{\mathbf{x}}_j]$ est calculée

par le biais de l'EDO englobante locale, les solutions $\omega(t)$ et $\Omega(t)$ ne sont valables que sur l'intervalle $[t_j, t_e]$, et $[\tilde{\mathbf{x}}_j]$ ne contient pas l'encadrement de la solution $[\mathbf{x}](t)$ pour $t \in]t_e, t_{j+1}]$. C'est pour cette raison que $[\tilde{\mathbf{x}}_j]$ doit être recalculée en intégrant directement l'EDO incertaine originale par l'une des méthodes de Taylor intervalles.

Proposition 2 : *(Commutation $q = 0 \mapsto q' \neq 0$)*

$$Si\,((q=0) \wedge (\forall i,\, \forall l,\, \forall r,\, (0 \notin [g]_{i,r}([\tilde{\mathbf{x}}_j], [\mathbf{p}], [t_j,\, t_{j+1}])) \wedge (0 \notin [h]_{i,l}([\tilde{\mathbf{x}}_j], [\mathbf{p}], [t_j,\, t_{j+1}]))))$$
$$\Rightarrow (e = (q, q') \wedge (q' \neq 0)) \tag{IV.13}$$

Preuve :

Dans ce cas, le mode courant est $q = 0$. Ce mode est caractérisé par le fait qu'au moins une des dérivées partielles $g_{i,k}(.)$ ou $h_{i,l}(.)$ change de signe. Par ailleurs, si pour tout $t \in [t_j, t_{j+1}]$ ces dérivées partielles ne changent plus de signe, alors une transition apparaît et le nouveau mode est nécessairement $q' \neq 0$. Etant donné que la solution *a priori* $[\tilde{\mathbf{x}}_j]$ est calculée par le biais de l'EDO incertaine originale alors nous pouvons continuer l'intégration à partir de t_{j+1} avec le nouveau mode $q' \neq 0$.

Finalement, l'algorithme qui permet d'implémenter l'évolution de l'automate hybride $\mathcal{H}$, pour fournir un encadrement garanti de toutes les solutions d'une EDO du type (III.32), est présenté ci-dessous

Algorithme Hybrid-Bounding(

$in : t_0, t_{n_T}, \mathbb{F}, [\mathbf{x}_0], [\mathbf{p}]; out : [\tilde{\mathbf{x}}_0], [\tilde{\mathbf{x}}_1], \ldots, [\tilde{\mathbf{x}}_{n_T}], [\mathbf{x}_1], \ldots, [\mathbf{x}_{n_T}])$

1. $j := 0$;
2. $q :=$ **Initialize**$(\mathbf{f}, [\mathbf{x}_0], [\mathbf{p}])$;
3. while $(j < n_T)$ do
4. $\quad \{h_j, [\mathbf{x}_{j+1}], [\tilde{\mathbf{x}}_j]\} :=$ **Integrate-one-step-ahead**$(q, \{\mathbf{f}\}, \mathbb{F}, t_j, [\mathbf{x}_j], [\mathbf{p}])$;
5. $\quad \{_jump_, q'\} :=$ **Check-Switching**$(q, \{\mathbf{f}\}, [\tilde{\mathbf{x}}_j])$;
6. $\quad$ if $(_jump_)$ then
7. $\quad\quad$ if $(q = 0)$ then
8. $\quad\quad\quad q := q'$;
9. $\quad\quad\quad j := j + 1$;

10. else
11. $q := 0$;
12. endif
13. else
14. $j := j + 1$;
15. endif
16. end.

Par ailleurs, l'algorithme **Integrate-one-step-ahead**, détaillé ci-dessous, calcule sur un pas d'intégration, un encadrement garanti de solutions d'équation différentielles incertaines.

Algorithme Integrate-one-step-ahead $(in : q, \mathbb{F}, t_j, [\mathbf{x}_j], [\mathbf{p}]; out : h_j, [\mathbf{x}_{j+1}], [\tilde{\mathbf{x}}_j])$

1. if $q := 0$ then
2. $\{h_j, [\mathbf{x}_{j+1}], [\tilde{\mathbf{x}}_j]\} :=$ **Interval-Integrate**$(\mathbf{f}, [\mathbf{x}_j], [\mathbf{p}], t_j)$;
3. else
4. $(\underline{\mathbf{f}}_q, \overline{\mathbf{f}}_q) :=$ **Select-Boundings**$(q, \mathbb{F})$;
5. $[\omega_j] := [\underline{\mathbf{x}}_j]$;
6. $[\Omega_j] := [\overline{\mathbf{x}}_j]$;
7. $\{h_j, [\omega_{j+1}], [\Omega_{j+1}], [\tilde{\omega}_j], [\tilde{\Omega}_j])\} :=$ **Interval-Integrate**$(\underline{\mathbf{f}}_q, \overline{\mathbf{f}}_q, [\omega_j], [\Omega_j], [\underline{\mathbf{p}}], [\overline{\mathbf{p}}], t_j)$;
8. $[\tilde{\mathbf{x}}_j] := [\tilde{\omega}_j, \tilde{\Omega}_j]$;
9. $[\mathbf{x}_{j+1}] := [\omega_{j+1}, \Omega_{j+1}]$;
10. end

L'algorithme **Hybrid-Bounding** cherche le mode discret initial $q \in \mathbb{Q}$ dans la ligne 2 en exécutant la fonction **Initialize**. En effet, cette fonction calcule la valeur de q selon les signes de toutes les dérivées partielles $g_{i,k}(.)$ et $h_{i,l}(.)$. S'il n'est pas possible d'instancier le signe de toutes les dérivées partielles $g_{i,k}(.)$ et de $h_{i,l}(.)$, alors le mode est $q = 0$. Tant que le temps courant d'intégration est inférieur à t_{nT}, l'algorithme **Integrate-one-step-ahead** réalise un pas d'intégration de t à $t + h$ (ligne 4) et la fonction **Check-Switching** vérifie si une commutation $e \in \mathbb{E}$ entre les modes est activée sur l'intervalle $[t_j, t_{j+1}]$ (ligne 5), ce

qui se produit si une des dérivées partielles $g_{i,k}(.)$ ou $h_{i,l}(.)$ a changé de signe. Si c'est le cas, la variable boolienne *_jump_* est égale à *true*, sinon elle est égale à *false*. L'action à suivre dans le cas de commutation dépendra du mode courant. En effet, si le mode courant est $q = 0$ il suffit de commuter vers le nouveau mode $q' \neq 0$ et de continuer l'intégration (lignes 8-9) selon la Proposition 2. Par contre, si le mode courant est différent de 0, alors nous devrons, selon la Proposition 1, refaire l'intégration sur le pas d'intégration courant avec le mode 0 afin de traverser la condition de commutation sans perdre la garantie de l'encadrement (ligne 11).

Dans l'algorithme **Integrate-one-step-ahead**, pour le cas du mode $q = 0$ l'intégration numérique de l'équation différentielle incertaine est réalisée par le biais d'une des méthodes basées sur le développement de Taylor intervalles (ligne 2). En revanche, pour les modes $q \neq 0$ nous sélectionnons tout d'abord l'EDO englobante couplée appropriée à la ligne 4 et les conditions initiales à l'instant t_j ($\omega(t_j)$, $\Omega(t_j)$) sont fixées aux lignes 5 et 6. Ensuite, l'intégration numérique de cette EDO englobante est exécutée à la ligne 7. Enfin, pour garantir numériquement les résultats obtenues, nous utilisons, comme dans la ligne 2, les méthodes d'intégration basées sur les modèles de Taylor intervalles pour résoudre les EDO englobantes du type (III.45).

Exemple : Considérons l'EDO non monotone ci-dessous :

$$\begin{cases} \dot{x}_1 &= -x_1 + \frac{1}{1+u(t)} \\ \dot{x}_2 &= -x_1 - x_2 + p_1 x_3 - u(t) \\ \dot{x}_3 &= -x_1 + p_2 x_2 - x_3 \end{cases} \tag{IV.14}$$

où le pavé d'état initial est $[\mathbf{x}_0]^T = [-1, 0] \times [1, 3] \times [2, 4]$. Le vecteur de paramètres incertains est $[\mathbf{p}]^T = [1, 1.2] \times [0.8, 1]$ et l'entrée est une rampe incertaine $[u](t) = [1, 1.2]t$. Donc, selon les signes des variables $x_3 = \frac{\partial \dot{x}_2}{\partial p_1}$ et $x_2 = \frac{\partial \dot{x}_3}{\partial p_2}$, l'EDO (IV.14) admet quatre EDO englobantes couplées locales. Ces EDO englobantes sont obtenues par l'utilisation

de la **Règle 2**

$$q=1\begin{cases}\dot{\underline{x}}_1 &= -\underline{x}_1+\frac{1}{1+\overline{u}(t)}\\ \dot{\underline{x}}_2 &= -\overline{x}_1-\underline{x}_2+\underline{p}_1\underline{x}_3-\overline{u}(t)\\ \dot{\underline{x}}_3 &= -\overline{x}_1+\underline{p}_2\underline{x}_2-\underline{x}_3\\ \dot{\overline{x}}_1 &= -\overline{x}_1+\frac{1}{1+\underline{u}(t)}\\ \dot{\overline{x}}_2 &= -\underline{x}_1-\overline{x}_2+\overline{p}_1\overline{x}_3-\underline{u}(t)\\ \dot{\overline{x}}_3 &= -\underline{x}_1+\overline{p}_2\overline{x}_2-\overline{x}_3\end{cases}\quad q=2\begin{cases}\dot{\underline{x}}_1 &= -\underline{x}_1+\frac{1}{1+\overline{u}(t)}\\ \dot{\underline{x}}_2 &= -\overline{x}_1-\underline{x}_2+\underline{p}_1\underline{x}_3-\overline{u}(t)\\ \dot{\underline{x}}_3 &= -\overline{x}_1+\overline{p}_2\underline{x}_2-\underline{x}_3\\ \dot{\overline{x}}_1 &= -\overline{x}_1+\frac{1}{1+\underline{u}(t)}\\ \dot{\overline{x}}_2 &= -\underline{x}_1-\overline{x}_2+\overline{p}_1\overline{x}_3-\underline{u}(t)\\ \dot{\overline{x}}_3 &= -\underline{x}_1+\underline{p}_2\overline{x}_2-\overline{x}_3\end{cases}\tag{IV.15}$$

$$q=3\begin{cases}\dot{\underline{x}}_1 &= -\underline{x}_1+\frac{1}{1+\overline{u}(t)}\\ \dot{\underline{x}}_2 &= -\overline{x}_1-\underline{x}_2+\overline{p}_1\underline{x}_3-\overline{u}(t)\\ \dot{\underline{x}}_3 &= -\overline{x}_1+\underline{p}_2\underline{x}_2-\underline{x}_3\\ \dot{\overline{x}}_1 &= -\overline{x}_1+\frac{1}{1+\underline{u}(t)}\\ \dot{\overline{x}}_2 &= -\underline{x}_1-\overline{x}_2+\underline{p}_1\overline{x}_3-\underline{u}(t)\\ \dot{\overline{x}}_3 &= -\underline{x}_1+\overline{p}_2\overline{x}_2-\overline{x}_3\end{cases}\quad q=4\begin{cases}\dot{\underline{x}}_1 &= -\underline{x}_1+\frac{1}{1+\overline{u}(t)}\\ \dot{\underline{x}}_2 &= -\overline{x}_1-\underline{x}_2+\overline{p}_1\underline{x}_3-\overline{u}(t)\\ \dot{\underline{x}}_3 &= -\overline{x}_1+\overline{p}_2\underline{x}_2-\underline{x}_3\\ \dot{\overline{x}}_1 &= -\overline{x}_1+\frac{1}{1+\underline{u}(t)}\\ \dot{\overline{x}}_2 &= -\underline{x}_1-\overline{x}_2+\underline{p}_1\overline{x}_3-\underline{u}(t)\\ \dot{\overline{x}}_3 &= -\underline{x}_1+\underline{p}_2\overline{x}_2-\overline{x}_3.\end{cases}\tag{IV.16}$$

Ainsi, l'exécution de l'algorithme **Hybrid-Bounding** sur une période $T=[0,10]s$, permet d'encadrer toutes les solutions possibles de (IV.14). Dans les figures IV.3, IV.4 et IV.5, nous comparons les encadrements de solutions obtenus par notre méthode hybride d'intégration garantie et ceux obtenus par la méthode de Hermite-Obreschkoff intervalle. En effet, les courbes bleues en ligne continue représentent les encadrements de solutions donnés par la méthode hybride et les courbes rouges en ligne pointillée montrent ceux calculés par la méthode de Hermite-Obreschkoff intervalle. Par ailleurs, la figure IV.6 montre les différents modes d'encadrement utilisés sur cette période d'intégration garantie.

IV.5 L'approche hybride interfacée avec un algorithme de bissection

Généralement, la capacité de l'algorithme **Hybrid-Bounding** à résoudre des problèmes pratiques avec succès dépend de sa capacité à déduire sans ambiguïté les signes des dérivées $g_{i,k}(.)$ et $h_{i,l}(.)$ sur la quasi-totalité de la période de temps d'intégration.

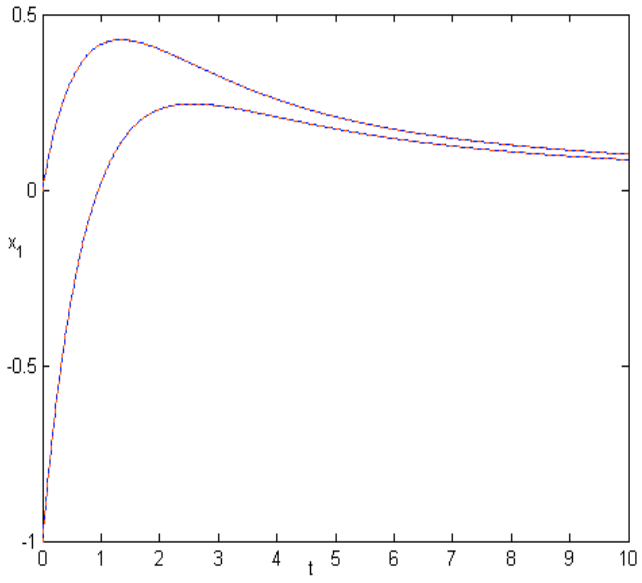


Fig. IV.3 *: Encadrement de toutes les solutions possibles de $x_1(t)$: courbes rouges pointillées méthode de Hermite-Obreschkoff ; courbes bleues continues méthode hybride d'encadrement.*

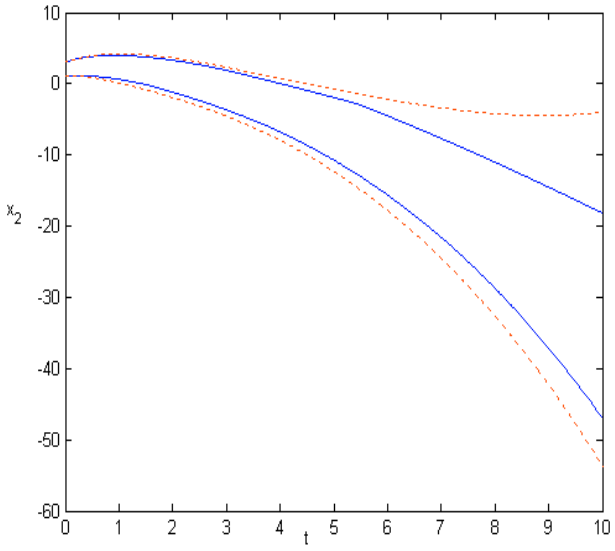


Fig. IV.4 *: Encadrement de toutes les solutions possibles de $x_2(t)$: courbes rouges pointillées méthode de Hermite-Obreschkoff ; courbes bleues continues méthode hybride d'encadrement.*

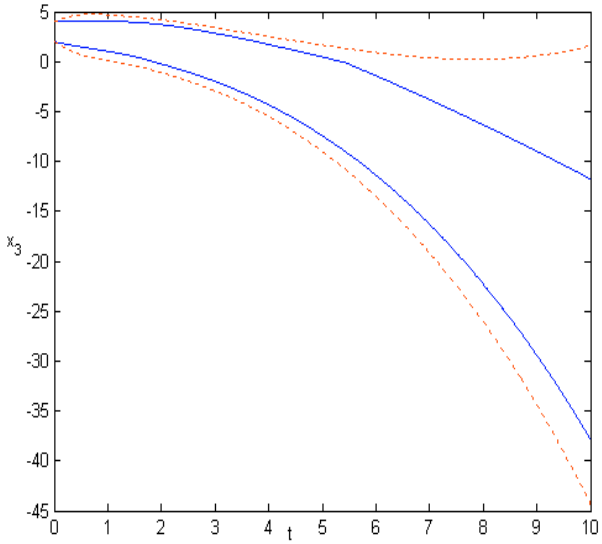


FIG. IV.5 *: Encadrement de toutes les solutions possibles de $x_3(t)$: courbes rouges pointillées méthode de Hermite-Obreschkoff; courbes bleues continues méthode hybride d'encadrement.*

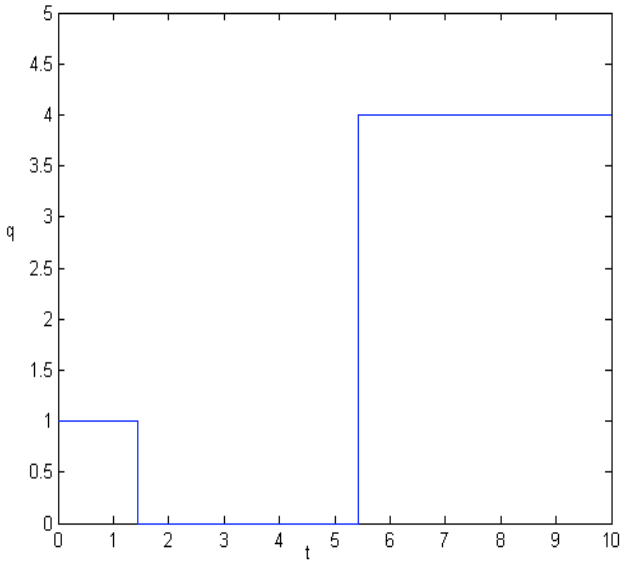


FIG. IV.6 *: Les différents modes d'encadrement utilisés au cours de l'intégration garantie* (IV.14).

Cependant, si la taille du pavé de conditions initiales et/ou celle du pavé de paramètres sont importantes, il est difficile de déterminer les signes de ces dérivées partielles. Ceci contraint l'algorithme à utiliser le mode $q = 0$, c'est-à-dire l'EDO incertaine initiale le long de la période d'intégration. Une manière simple de contourner ce problème consiste à bissecter le pavé des variables et celui des paramètres de sorte que la détermination des signes des dérivées partielles soit possible. Généralement, cette procédure de bissection est applicable à l'initialisation de l'algorithme et aux traversées de conditions de garde qui engendrent les commutations entre les modes de l'automate hybride $\mathcal{H}$.

Pour des raisons de simplicité, nous utilisons dans la suite un vecteur de variables étendu $\mathbf{z}(t) = (\mathbf{x}(t), \mathbf{p})$. Ainsi, à chaque instant d'intégration t_j nous aurons comme pavé de variables $[\mathbf{z}_j] = ([\mathbf{x}_j], [\mathbf{p}])$. Par ailleurs, pour bien gérer le traitement de tous les sous-pavés issus de la bissection, le nouveau algorithme est doté de plusieurs listes.

Avant de donner l'algorithme global **Hybrid-Bounding-with-Partition** qui combine l'approche hybride d'encadrement des solutions d'EDO avec une bissection de pavés intervalles, nous présentons tout d'abord l'algorithme **List-Initialize**. Cet algorithme utilise une fonction **Partition**, qui bissecte un pavé donné $[\mathbf{z}]$ en deux sous-pavés $L[\mathbf{z}]$ et $R[\mathbf{z}]$. Cette bissection est réalisée dans le but de construire une liste de sous-pavés dans lesquels les signes des dérivées partielles seront connus sans ambiguïté. De plus, cette liste contient aussi les sous-pavés de taille intérieure à ϵ pour lesquels le mode $q = 0$ doit être être utilisé. En général, ϵ est réglé de telle sorte que l'intégration avec les méthodes basées sur les modèles de Taylor intervalle peut être utilisée avec succès.

Algorithme List-Initialize $(in : \mathbf{f}, [\mathbf{z}_0], , \epsilon; out : \mathcal{L}_0^F)$

1. $\mathcal{L}_{init} \leftarrow [\mathbf{z}_0]$;
2. while $(\mathcal{L}_{init} \neq \emptyset)$ do
3. $\quad [\mathbf{z}_0] := pop(\mathcal{L}_{init})$;
4. $\quad$ if $(\forall i, \forall l, \forall k, (0 \notin [g]_{i,k}([\mathbf{z}_0], t_0)) \wedge (0 \notin [h]_{i,l}([\mathbf{z}_0], t_0))))$ then
5. $\quad\quad$ set q ;
6. $\quad\quad \mathcal{L}_0^F \leftarrow \{q, [\mathbf{z}_0]\}$;
7. $\quad$ else
8. $\quad\quad$ if $(w([\mathbf{z}_0]) \leq \epsilon)$ then

9. $\mathcal{L}_0^F \leftarrow \{(0, [\mathbf{z}_0])\}$
10. else
11. $\{[^r\mathbf{z}_0], [^l\mathbf{z}_0]\} := \textbf{Partition}([\mathbf{z}_0])$;
12. $\mathcal{L}_{init} \leftarrow {}^r[\mathbf{z}_0]$;
13. $\mathcal{L}_{init} \leftarrow {}^l[\mathbf{z}_0]$;
14. endif ;
15. endif ;
16. end while ;

Ainsi, les différences principales entre ce nouveau algorithme **Hybrid-Bounding-with-Partition** et l'algorithme **Hybrid-Bounding** sont : l'utilisation de la fonction de bissection **Partition** et l'introduction de listes dont les éléments sont les couples (mode, encadrement de solutions). $\mathcal{L}$ est la liste courante, $\mathcal{L}_j^F$ la liste des encadrements solutions aux instants t_j et $\mathcal{L}_j^R$ la liste des solutions a priori $[\tilde{\mathbf{z}}_j]$ qui contiennent toutes les trajectoire solutions sur les intervalles $[t_j, t_{j+1}]$.

Algorithme Hybrid-Bounding-with-Partition(
$in : t_0, t_{n_T}, \mathbb{F}, [\mathbf{z}_0], \epsilon; out : \mathcal{L}_0^R, \mathcal{L}_1^R, \ldots, \mathcal{L}_{n_T}^R, \mathcal{L}_1^F, \ldots, \mathcal{L}_{n_T}^F$)

1. $\mathcal{L}_0^F := \textbf{List-Initialize}(\mathbf{f}, [\mathbf{z}_0], \epsilon)$;
2. for $j := 0$ to n_T do
3. $\mathcal{L} := \mathcal{L}_j^F$;
4. while $(\mathcal{L} \neq \emptyset)$ do
5. $\{(q, [\mathbf{z}_j])\} := pop(\mathcal{L})$;
6. $\{h_j, [\mathbf{z}_{j+1}], [\tilde{\mathbf{z}}_j]\} := \textbf{Integrate-one-step-ahead}(q, \{\mathbf{f}\}, \mathbb{F}, t_j, [\mathbf{z}_j])$;
7. $\{_jump_, q'\} := \textbf{Check-Switching}(q, \{\mathbf{f}\}, [\tilde{\mathbf{z}}_j])$;
8. if $(_jump_)$ then
9. if $(q = 0)$ then
10. $\mathcal{L}_{j+1}^F \leftarrow (q', [\mathbf{z}_{j+1}])$;
11. $\mathcal{L}_j^R \leftarrow (0, [\tilde{\mathbf{z}}_j])$;
12. else

13. if $(w([\mathbf{z}_j]) \leq \epsilon)$ then
14. $\mathcal{L} \leftarrow (0, [\mathbf{z}_j])$
15. else
16. $\{[^r\mathbf{z}_j], [^l\mathbf{z}_j]\} := \textbf{Partition}([\mathbf{z}_j])$
17. $\mathcal{L} \leftarrow (q, {}^r[\mathbf{z}_j])$;
18. $\mathcal{L} \leftarrow (q, {}^l[\mathbf{z}_j])$;
19. endif
20. else
21. $\mathcal{L}^F_{j+1} \leftarrow (q, [\mathbf{z}_{j+1}])$;
22. $\mathcal{L}^R_j \leftarrow (q, [\tilde{\mathbf{z}}_j])$;
23. endif
24. end while
25. **Regularize**$(\mathcal{L}^F_{j+1})$;
26. end for

Finalement, l'utilisation de l'algorithme **Regularize** est facultative. En effet, cet algorithme a pour but de réduire la complexité du calcul en réduisant la taille des listes. Par exemple, en fusionnant les encadrements de solutions lorsqu'ils appartiennent au même mode $q \neq 0$, ce qui est fréquent après chaque franchissement d'une condition de garde.

IV.6 Cas des EDO coopératives

Dans le cas des EDO coopératives ou les EDO qui deviennent coopératives après un changement de variables, les EDO englobantes locales sont toujours découplées et leur détermination se fait séparément en utilisant la Règle 1. Par conséquent, l'automate hybride $\mathcal{H}$ peut être vu comme étant deux automates inférieur et supérieur qui évoluent indépendamment. L'exécution de ces deux automates génère respectivement une approximation inférieure de la solution minimale de (III.16) et une approximation supérieure de sa solution maximale. Alors la collection des champs de vecteurs $\mathbb{F}$ définie par la relation

(IV.2) devient dans ce contexte $\mathbb{F} = \{\underline{\mathbb{F}} \cup \overline{\mathbb{F}}, \mathbf{f}\}$ où

$$\forall q \in \mathbb{Q}, \quad \begin{cases} \underline{\mathbf{f}}_q & : \quad \mathbb{X} \times \mathbb{P}^2 \longrightarrow \mathbb{R}^n \in \underline{\mathbb{F}} \\ \overline{\mathbf{f}}_q & : \quad \mathbb{X} \times \mathbb{P}^2 \longrightarrow \mathbb{R}^n \in \overline{\mathbb{F}}. \end{cases} \tag{IV.17}$$

De plus, dans ce contexte, les tailles des intervalles $[I_s]$ sur lesquels le mode $q = 0$ est utilisé sont égales au plus au pas d'intégration. Donc, l'effet d'enveloppement est assez bien contrôlé. Par conséquent, nous n'avons pas besoin de recourir à la bissection pour lever l'ambiguïté sur les signes des dérivées partielles $g_{i,k}(.)$ et $h_{i,l}(.)$.

IV.6.1 Exemple illustratif

Nous considérons ici une EDO mono-dimensionnel avec deux paramètres incertains p_1 et p_2 :

$$\begin{cases} \dot{x} & = & f(x, p_1, p_2, t) \\ x_0 & \in & [\underline{x}_0, \overline{x}_0] \\ \mathbf{p} & \in & [\underline{p}_1, \overline{p}_1] \times [\underline{p}_2, \overline{p}_2]. \end{cases} \tag{IV.18}$$

Remarque : Par définition toutes les EDO de dimension 1 sont coopératives.

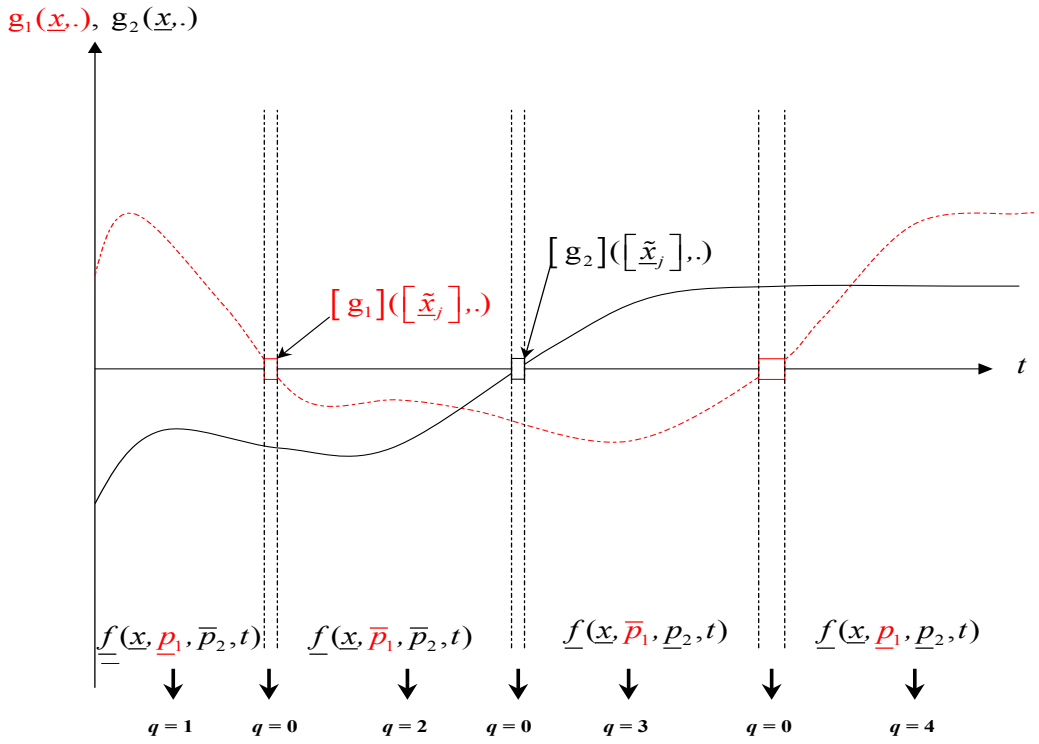


Fig. IV.7 *: Les différents modes d'encadrement inférieurs possibles de* (IV.18)

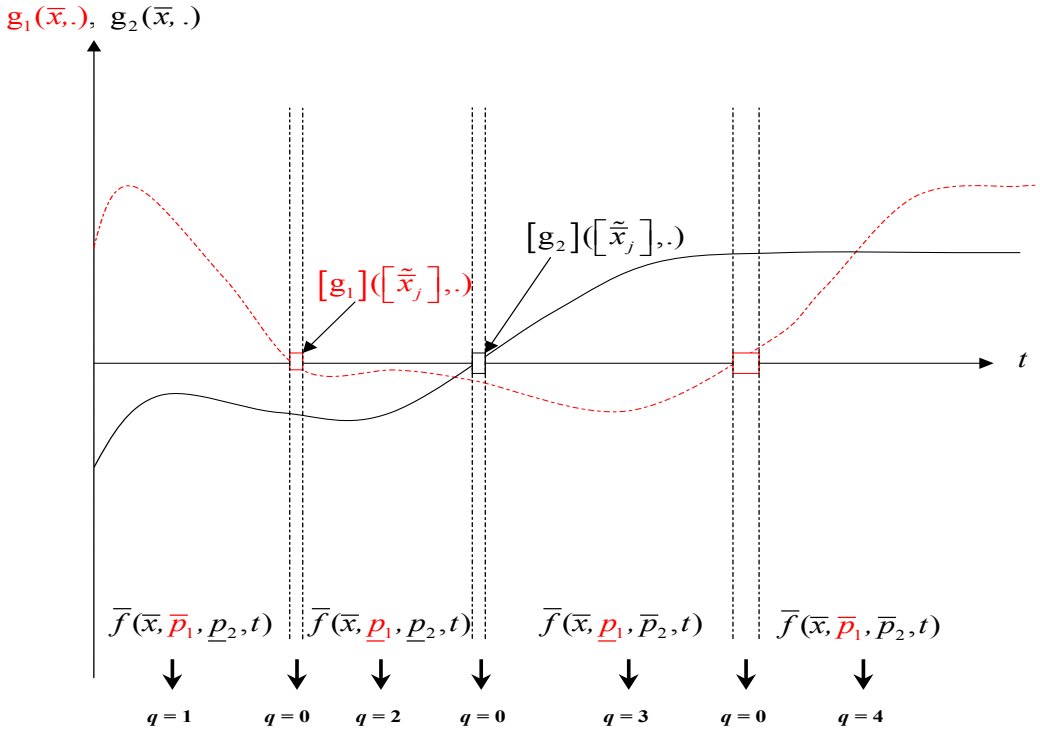


Fig. IV.8 : *Les différents modes d'encadrement supérieurs possibles de* (IV.18)

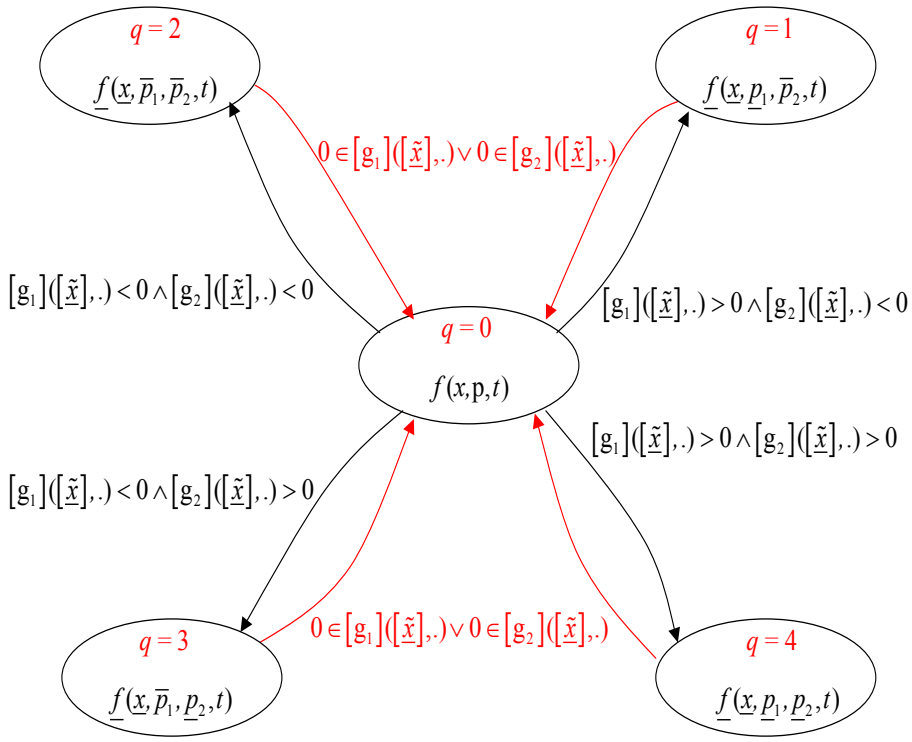


Fig. IV.9 : *Automate hybride inférieur*

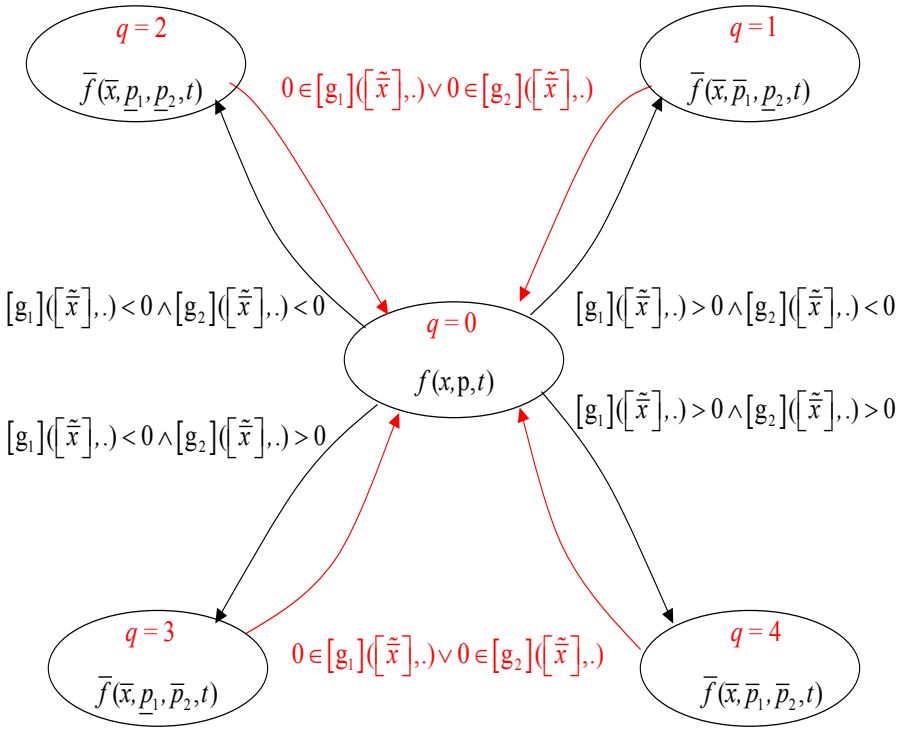


Fig. IV.10 *: Automate hybride supérieur*

Selon les signes des dérivées partielles $g_1(.)$ et $g_2(.)$ par rapport à $\underline{x}$ et par rapport à $\overline{x}$, présentées respectivement dans les figures IV.7 et IV.8, l'EDO (IV.18) admet quatre modes d'encadrement inférieur (IV.19) et quatre modes d'encadrement supérieur (IV.20).

1. Modes d'encadrement inférieur

$$\left\{\begin{array}{lcl} q & = & 1 \\ \dot{\underline{x}} & = & \underline{f}(\underline{x},\underline{p}_1,\overline{p}_2,t) \end{array}\right. \quad \left\{\begin{array}{lcl} q & = & 2 \\ \dot{\underline{x}} & = & \underline{f}(\underline{x},\overline{p}_1,\overline{p}_2,t) \end{array}\right. \tag{IV.19}$$

$$\left\{\begin{array}{lcl} q & = & 3 \\ \dot{\underline{x}} & = & \underline{f}(\underline{x},\overline{p}_1,\underline{p}_2,t) \end{array}\right. \quad \left\{\begin{array}{lcl} q & = & 4 \\ \dot{\underline{x}} & = & \underline{f}(\underline{x},\underline{p}_1,\underline{p}_2,t). \end{array}\right.$$

2. Modes d'encadrement supérieur

$$\left\{\begin{array}{lcl} q & = & 1 \\ \dot{\overline{x}} & = & \overline{f}(\overline{x},\overline{p}_1,\underline{p}_2,t) \end{array}\right. \quad \left\{\begin{array}{lcl} q & = & 2 \\ \dot{\overline{x}} & = & \overline{f}(\overline{x},\underline{p}_1,\underline{p}_2,t) \end{array}\right. \tag{IV.20}$$

$$\left\{\begin{array}{lcl} q & = & 3 \\ \dot{\overline{x}} & = & \overline{f}(\overline{x},\underline{p}_1,\overline{p}_2,t) \end{array}\right. \quad \left\{\begin{array}{lcl} q & = & 4 \\ \dot{\overline{x}} & = & \overline{f}(\overline{x},\overline{p}_1,\overline{p}_2,t). \end{array}\right.$$

L'EDO associée à chaque mode est obtenue par l'utilisation locale de la Règle 1. Le passage d'un mode à un autre se fait par l'intermédiaire du mode $q = 0$ dont la durée est inférieure ou égale à la largeur du pas d'intégration courant.

Evidemment, l'exécution séparée des deux automates hybrides inférieur et supérieur présentés respectivement dans les figures IV.9 et IV.10 permet de calculer un encadrement garanti de toutes les solutions possibles de (IV.18) sur toute la période de simulation.

IV.6.2 Algorithme d'encadrement dans le cas coopératif

Pour synthétiser cette méthode de calcul d'une approximation supérieure (resp. inférieure) de la solution maximale (resp. minimale) d'une EDO coopérative incertaine, nous proposons l'algorithme ci-dessous qui peut être utilisé indépendamment pour construire ces approximations.

Algorithm Hybrid_Bound_Solution

$(in : t_0, t_{n_T}, \mathbb{F}, \mathbf{Bnd}([\mathbf{x}_0]), [\mathbf{p}]; \quad out : \mathbf{Bnd}([\mathbf{x}_1]), \ldots, \mathbf{Bnd}([\mathbf{x}_{n_T}]))$

1. $j := 0$;
2. $q := \mathbf{Initialize}(\mathbf{f}, \mathbf{Bnd}([\mathbf{x}_0]), [\mathbf{p}])$;
3. while $(j < n_T)$ do
4. $\quad \{h_j, \mathbf{Bnd}([\mathbf{x}_{j+1}]), \mathbf{Bnd}([\tilde{\mathbf{x}}_j])\} := \mathbf{Integrate\text{-}one\text{-}step\text{-}ahead}(q, \mathbb{F}, t_j, \mathbf{Bnd}([\mathbf{x}_j]), [\mathbf{p}])$;
5. $\quad \{_jump_, q'\} := \mathbf{Check\text{-}Switching}(q, \{\mathbf{f}\}, \mathbf{Bnd}([\tilde{\mathbf{x}}_j]))$;
6. $\quad$ if $(_jump_)$ then
7. $\quad\quad$ if $(q = 0)$ then
8. $\quad\quad\quad q := q'$;
9. $\quad\quad\quad j := j + 1$;
10. $\quad\quad$ else
11. $\quad\quad\quad q := 0$;
12. $\quad\quad$ endif
13. $\quad$ else
14. $\quad\quad j := j + 1$;
15. $\quad$ endif

16. end.

Ainsi, les approximations de solutions minimale et maximale qui encadrent toutes les trajectoires solutions possibles de (III.16) sur la période $[t_0, t_{n_T}]$ sont calculées par ce même algorithme. En effet, il suffit de configurer la fonction **Bnd** pour obtenir soit l'approximation inférieure de la solution minimale ou l'approximation supérieure de la solution maximale. Par exemple, $\mathbf{Bnd}([\mathbf{x}])$ est égale à $\underline{\mathbf{x}}$ ou à $\overline{\mathbf{x}}$.

L'algorithme **Hybrid_Bound_Solution** cherche le mode discret initial $q \in \mathbb{Q}$ dans la ligne 2 en exécutant la fonction **Initialize**, cette fonction calcule la valeur de q selon les signes de toutes les dérivées partielles de $\mathbf{f}$ par rapport aux paramètres incertains. Il s'agit donc de mettre en œuvre la Règle 1. Par la suite, la boucle **while** réalise l'intégration numérique garantie du mode courant défini par $\mathbf{Bnd}(\mathbf{f})_q$ et ce jusqu'à ce qu'une transition $e \in \mathbb{E}$ soit activée. Ceci est détecté par la fonction **Check-Switching**, de la ligne 5, qui retourne en même temps le nouveau mode discret q'. Par ailleurs, pour franchir les conditions de garde détectées sur les intervalles $[I_s] = [t_j, t_j + h_{j+1}]$ sans perdre la garantie de l'encadrement, cet algorithme refait l'intégration numérique sur ces intervalles en utilisant le mode $q = 0$, c'est-à-dire l'EDO incertaine initiale.

Exemple : Soit l'EDO non linéaire ci-dessous de dimension 13 avec 3 paramètres incertains

$$\left\{\begin{array}{lcll} \dot{x}_1 & = & 5.44(x_2 - 2x_1 + u_1(t)) & \\ \dot{x}_2 & = & 10.88(x_1 - 1.18x_2 + 0.18x_3) & \\ \dot{x}_3 & = & 2(p_0 + p_1x_3)(x_4 - x_3 + 2.3p_2(x_2 - x_3)) & \\ \dot{x}_i & = & (p_0 + p_1x_i)(x_{i+1} - 2x_i + x_{i-1}) & i = 4, \ldots, 9 \\ \dot{x}_{10} & = & 2(p_0 + p_1x_{10})(x_9 - x_{10} + 2.3p_2(x_{11} - x_{10})) & \\ \dot{x}_{11} & = & 11.4(x_{12} - 1.09x_{11} + 0.09x_{10}) & \\ \dot{x}_{12} & = & 5.7(x_{13} - 2x_{12} + x_{11}) & \\ \dot{x}_{13} & = & 11.4(x_{12} - 1.0003x_{13} + 0.0003u_2(t)). & \end{array}\right. \tag{IV.21}$$

Avec l'état initial est $[x_i](t_0) = [90, 110], \forall i \in \{1, ..., 13\}$ et le vecteur de paramètres est $\mathbf{p}^T = (p_0, p_1, p_2,) \in [0.01, 0.015] \times [0.7, 1.2] \times [0.2, 0.6]$. Le vecteur d'excitations extérieures $\mathbf{u}(t)$ est donné par

$$\begin{array}{rcl} u_1(t) & = & \left\{\begin{array}{lll} 150 & \text{si} & 0s \leq t \leq 50s \\ 100 & \text{si} & 50s < t \leq 100s \end{array}\right. \\ u_2(t) & = & 100 \end{array}$$

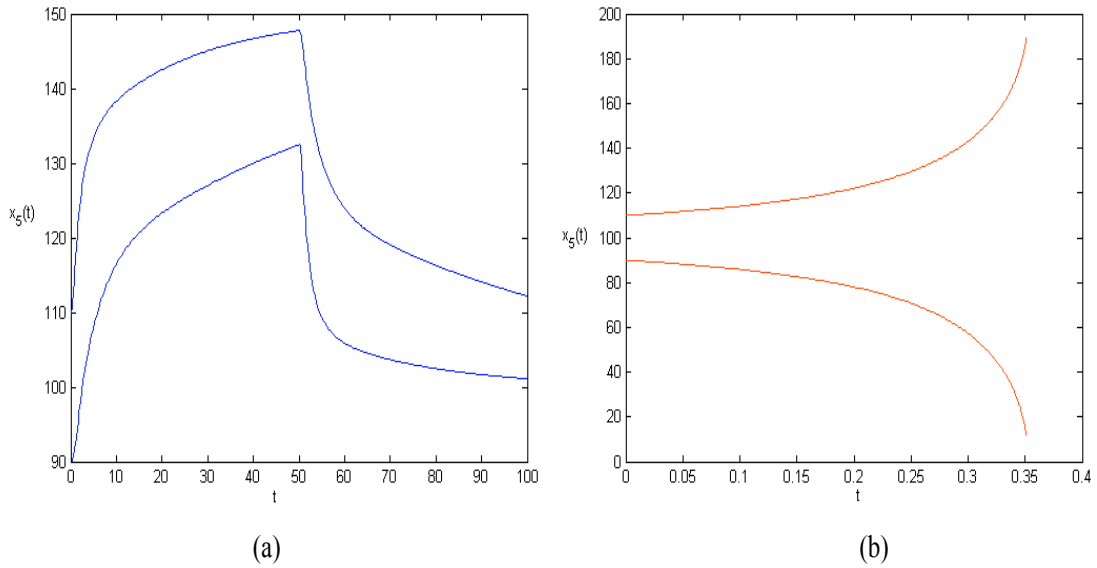


Fig. IV.11 *: Encadrement de toutes les solutions possibles de $x_5(t)$: (a) Méthode hybride d'encadrement; (b) Méthode de Hermite-Obreschkoff intervalle. Arrêt de l'algorithme de calcul après 0.35 seconde d'intégration.*

Tout d'abord, nous notons que, pour cet exemple, toutes les tentatives pour calculer un encadrement garanti de toutes les solutions possibles de (IV.21) par les méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalles ont échoué. En effet, la propagation du pessimisme dans ces méthodes conduit à une divergence rapide de la taille des encadrements de solutions. Ceci est illustré par les courbes rouges sur les figures IV.11 (b), IV.12 (b) et IV.13 (b) qui montrent bien cette divergence au bout de quelques pas d'intégration. En revanche, la méthode hybride permet de résoudre ce problème avec succès en utilisant l'algorithme **Hybrid_Bound_Solution** comme le montrent les figures IV.11 (a), IV.12 (a) et IV.13 (a), qui représentent respectivement les encadrements de toutes les solutions possibles de x_5, de x_7 et de x_{11}. Par ailleurs, les figures IV.14 (a) et IV.14 (b) montrent les différents modes d'encadrement utilisés le long de la durée d'intégration garantie de (IV.21). En fait, sur l'axe des ordonnées nous avons présenté des indices relatifs à ces modes d'encadrement. En réalité, pour cette EDO, comme nous avons 18 conditions de garde, alors elle admet 2^{18} modes d'encadrement possibles. En pratique, pour les conditions initiales de (IV.21) données ci-dessus l'automate hybride d'encadrement n'utilise que 8+1 modes.

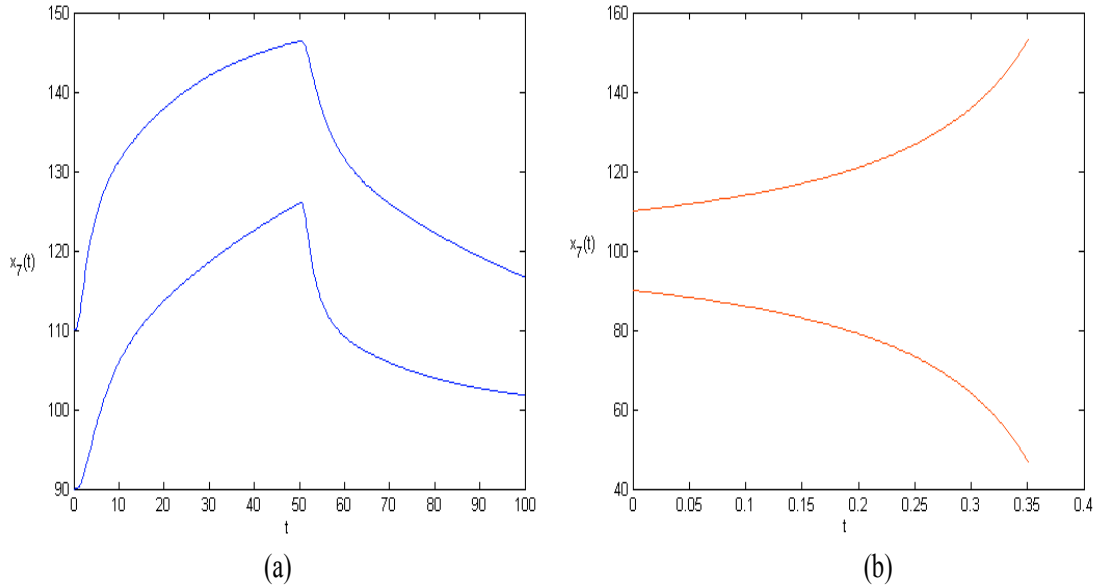


Fig. IV.12 *: Encadrement de toutes les solutions possibles de $x_7(t)$: (a) Méthode hybride d'encadrement ; (b) Méthode de Hermite-Obreschkoff intervalle. Arrêt de l'algorithme de calcul après 0.35 seconde d'intégration.*

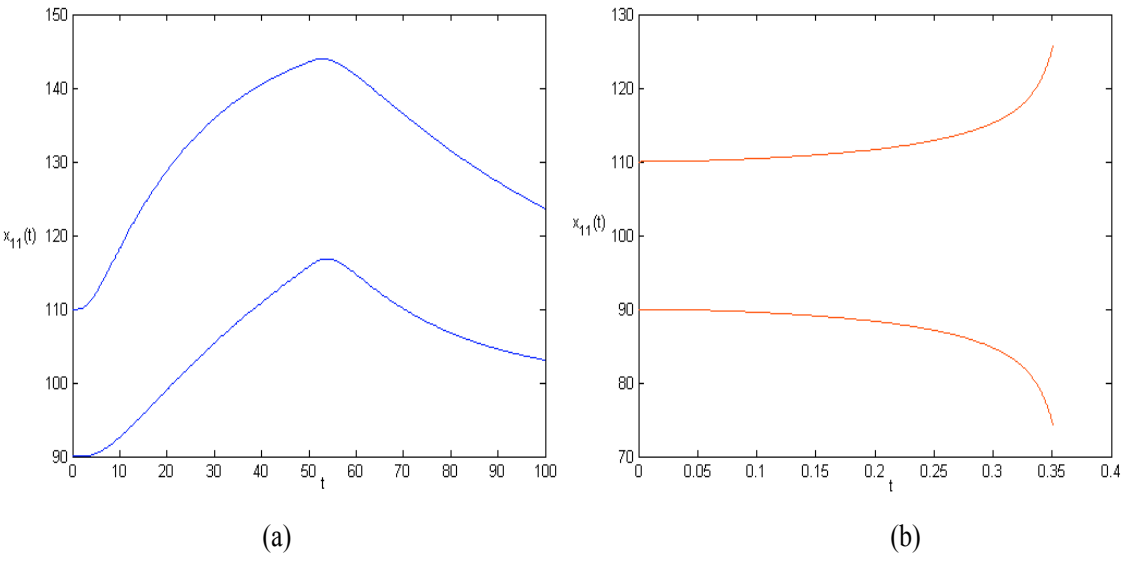


Fig. IV.13 *: Encadrement de toutes les solutions possibles de $x_{11}(t)$: (a) Méthode hybride d'encadrement ; (b) Méthode de Hermite-Obreschkoff intervalle. Arrêt de l'algorithme de calcul après 0.35 seconde d'intégration.*

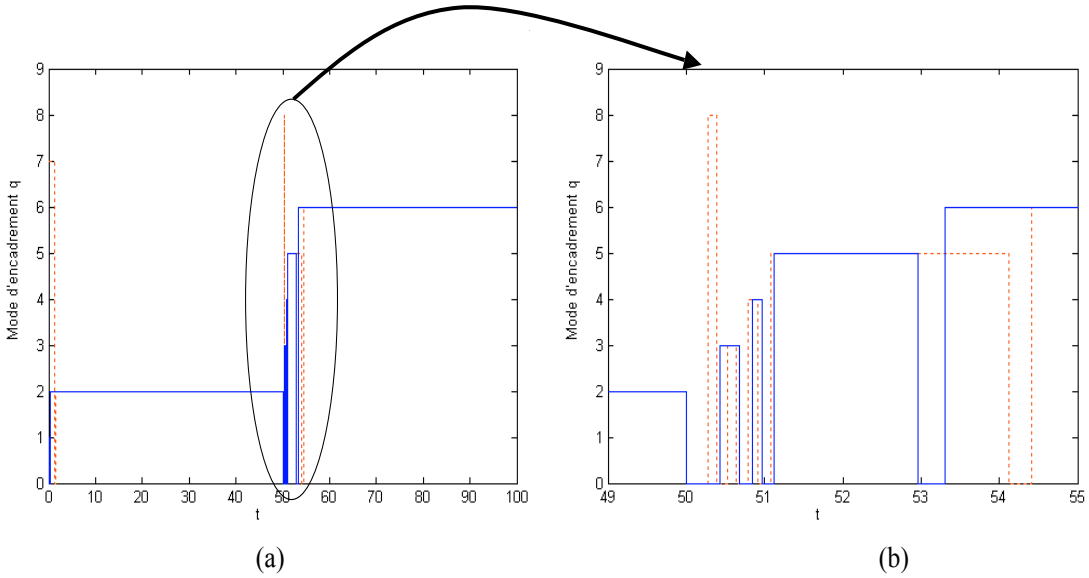


Fig. IV.14 *: Les différents modes d'encadrement utilisés sur la période d'intégration garantie de* (IV.21)*. La courbe rouge pointillée représente l'évolution de l'état discret du système hybride d'encadrement inférieur. La courbe bleue continue représente l'évolution de l'état discret du système hybride d'encadrement supérieur.*

IV.7 Conclusion

L'idée développée dans ce quatrième chapitre est de généraliser l'application des règles **Règle 1** et **Règle 2** présentées dans le chapitre III pour le cas des EDO qui ne vérifient pas les conditions des théorèmes **Théorème 1** et **Théorème 2**. Pour cela, nous avons proposé une approche hybride d'encadrement de solutions d'équations différentielles incertaines qui met en jeu un automate hybride. De plus, nous avons donné les algorithmes qui permettent son implémentation informatique.

Dans les chapitres suivants nous allons montrer comment on peut résoudre des problèmes liés au domaine de l'automatique continue, tel que l'identification, l'observation et l'étude d'atteignabilité des systèmes dynamiques continus incertains, par le biais de l'intégration numérique garantie des EDO et l'analyse par intervalles.

Chapitre V

Identification ensembliste des systèmes dynamiques à temps continu

V.1 Introduction

Le cadre statistique classiquement utilisé pour l'identification paramétrique consiste à modéliser l'écart entre les sorties du système physique et ceux de son modèle mathématique comme les réalisations d'un vecteur aléatoire de densité de probabilité connue ou paramétrée. Cet écart est essentiellement dû au choix de la structure du modèle et aux incertitudes sur les données disponibles (valeurs des mesures et valeurs des paramètres). La qualité des estimées ainsi obtenues est alors étroitement liée à cette loi de probabilité. Les différentes approches développées dans ce contexte classique ont obtenu de bons résultats pour un grand nombre de problèmes réels. Cependant, elles posent des difficultés fondamentales pour les problèmes d'identification non linéaires dans un contexte non asymptotique. Par exemple, ces approches sont mal adaptées à deux types d'erreurs [97] : (i) erreur structurelle déterministe ; (ii) erreur reliée au manque d'informations, par exemple le cas où la seule information disponible est sous forme de bornes.

Par ailleurs, dans ce contexte, l'estimée des paramètres est en général définie comme étant l'argument qui minimise un certain critère. Cependant, rien ne garantit que l'estimée calculée est unique. Pour cette raison, il est intéressant de pouvoir calculer l'ensemble des arguments du minimum de ce critère. En pratique, on ne dispose pas d'une expression analytique pour la valeur de cette estimée, il est d'usage d'utiliser des techniques itératives

locales. Cette approche pose des problème bien connus [96], comme par exemple : comment choisir le point de départ de ces techniques itératives locales, quand arrêter les itérations, comment ne pas se faire piéger par des minimiseurs locaux . . .

Récemment des approches ensemblistes pour l'identification ont été développées, notamment celles basées sur l'arithmétique par intervalles [94] [95]. Elles offrent une alternative séduisante pour caractériser de manière globale et garantie l'ensemble des valeurs de paramètres qui sont compatibles avec la structure du modèle choisi linéaire ou non-linéaire, avec les données expérimentales recueillies sur le système et avec l'hypothèse que les perturbations et bruits appartiennent à des domaines bornés. Dans ce contexte, il existe deux types de méthodes : le premier type s'appuie sur l'algorithme de l'optimisation globale de Hansen [38] basé sur l'analyse par intervalle et le second type repose sur l'étude de consistance entre les données expérimentales et les données simulées [79].

Dans ce chapitre, nous nous intéresserons au deuxième type de méthodes ensemblistes pour l'identification de modèles à temps continu dans un contexte à erreur inconnue mais bornée et de bornes connues *a priori*. Nous montrerons comment l'approche hybride d'encadrement développée dans le chapitre IV permet de résoudre avec succès un problème d'estimation que ne pourrait résoudre les méthodes d'intégration basées sur les modèles de Taylor intervalle.

V.2 Présentation de l'approche ensembliste pour l'identification

L'identification ensembliste dans le contexte à erreurs bornées est du type prédiction-correction et peut être présentée comme un problème d'inversion ensembliste. Dans le contexte de l'analyse par intervalles, des algorithmes de type *Branch-and-Bound*, par exemple l'algorithme SIVIA [51] présenté dans le chapitre I, permettent de caractériser l'ensemble des solutions du problème de façon complète et garantie. Contrairement aux méthodes classiques d'identification, cette approche ensembliste n'exige ni critère qui évalue l'écart modèle-système, ni beaucoup d'informations sur le système et son environnement pour modéliser explicitement les grandeurs incertaines, et ni méthode d'optimisation du critère. Le principe fondamental de cette approche est représenté sur la figure

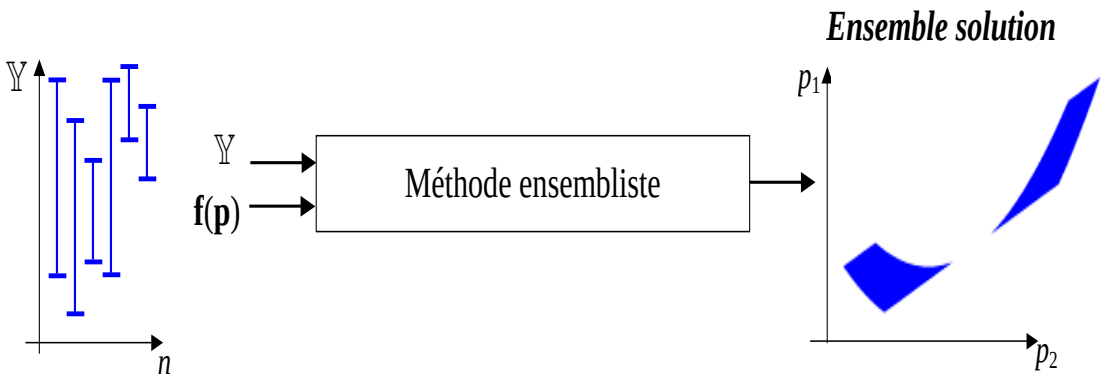


Fig. V.1 *: Principe d'identification à erreurs bornées*

V.1. En effet, cette figure montre les trois phases qui caractérisent l'approche ensembliste d'identification dans le contexte à erreurs bornées par le biais de l'analyse par intervalles :

1. Supposer que les données recueillies sur les systèmes physiques sont disponibles à des instants discrets sous forme de pavés bornés et de bornes connues *a priori*.

2. Réaliser les étapes *Branch-and-Bound* qui consistent à faire des comparaisons à chaque instant de mesure entre les données qui proviennent de la simulation du modèle et ceux de la mesure physique afin d'éliminer les données inconsistantes.

3. Obtenir une approximation globale et garantie du continuum de solutions, non nécessairement connexe. La projection de cet ensemble solution, ainsi calculé, sur les axes des paramètres donne des intervalles garantis d'incertitudes pour chaque paramètre identifié.

En pratique, l'évolution de très nombreux dispositifs physiques (mécaniques, thermiques, biologiques . . .) peut être décrite par des représentations d'état à temps continu ou à temps discret. Dans le cas des représentations d'état à temps discret, l'approche d'identification à erreurs bornées a fait l'objet de plusieurs travaux [64, 50]. Néanmoins, le cas des représentations d'état à temps continu est beaucoup moins développé [94, 75, 82, 54]. Dans la suite de ce chapitre, nous abordons le problème de l'identification du vecteur de paramètres $\mathbf{p}$ des systèmes non linéaires à temps continu modélisés par un système

d'équations différentielles et un système d'équations de sorties :

$$\left\{\begin{array}{c} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{p}, t) \\ \mathbf{y}(t) = \mathbf{h}(\mathbf{x}, \mathbf{p}, t) \\ \mathbf{x}(t_0) \in \mathbb{X} \\ \mathbf{p} \in \mathbb{P} \end{array}\right\} \tag{V.1}$$

où $t \in [t_0, T]$, $\mathbf{f} \in C^{k-1}(\mathbb{D} \times \mathbb{P})$, $\mathbb{D} \times \mathbb{P} \subseteq \mathbb{R}^{n+n_p}$ est un ensemble ouvert ; n, m et n_p représentent respectivement les dimensions du vecteur d'état $\mathbf{x}(t)$, du vecteur de sorties $\mathbf{y}(t)$ et du vecteur de paramètres $\mathbf{p}$. Les fonctions $\mathbf{f} : \mathbb{D} \times \mathbb{P} \to \mathbb{R}^n$ et $\mathbf{h} : \mathbb{D} \times \mathbb{P} \to \mathbb{R}^m$ peuvent être non linéaires. Ainsi, l'estimation ensembliste de $\mathbf{p}$ requiert le calcul des solutions de

$$\left\{\begin{array}{lcl} \dot{\mathbf{x}}(t) & = & \mathbf{f}(\mathbf{x}, \mathbf{p}, t) \\ \mathbf{x}(t_0) & \in & \mathbb{X} \\ \mathbf{p} & \in & \mathbb{P} \end{array}\right. \tag{V.2}$$

pour différentes valeurs de $\mathbf{x}(t_0)$ et de $\mathbf{p}$. Pour ce faire, nous utilisons les méthodes d'intégration numérique garantie des EDO incertaines présentées dans les chapitres II, III, et IV.

V.3 Identification ensembliste des systèmes dynamiques à temps continu

Dans ce contexte, l'état initial $\mathbf{x}(t_0)$ du système (V.2) est considéré incertain, donné par un pavé $[\mathbf{x}_0]$ qui contient toutes les valeurs possibles $\mathbf{x}(t_0) \in [\mathbf{x}_0]$. Les mesures $\mathbf{y}_j$ sont prélevées à des instants discrets $t_i \in \{t_1, t_2, \ldots, t_n\} \subset [t_0, T]$, où l'intervalle de mesure n'est pas nécessairement constant. Les bruits de mesures et les erreurs systématiques de modélisation sont considérées bornées et de bornes connues *a priori* et additives sur la sortie. Notons par $\mathbb{E}_j$ le domaine admissible des erreurs à l'instant t_j, par conséquent le domaine admissible de la sortie à ce même instant est donné par :

$$\mathbb{Y}_j = \mathbf{y}_j + \mathbb{E}_j. \tag{V.3}$$

Donc, l'identification ensembliste du vecteur de paramètres $\mathbf{p}$ consiste à caractériser l'ensemble solution $\mathbb{S}$ de tous les vecteurs de paramètres acceptables, c'est-à-dire

$$\mathbb{S} = \{\mathbf{p} \in \mathbb{P} \,|\, (\forall t \in [t_0, T],\, \dot{\mathbf{x}}(\mathbf{p}, t) = \mathbf{f}(\mathbf{x}, \mathbf{p}, t)) \wedge (\forall t_j \in \{t_1, t_2, \ldots, t_N\},\, \mathbf{h}(\mathbf{x}, \mathbf{p}, t_j) \in \mathbb{Y}_j)\}. \tag{V.4}$$

où le pavé $\mathbb{P}$ est l'espace initial de recherche des paramètres. D'un point de vue mathématique, caractériser l'ensemble $\mathbb{S}$ revient à résoudre un problème d'inversion ensembliste, c'est-à-dire, partant de l'ensemble d'arrivée $\mathbb{Y}$ nous souhaitons reconstruire l'ensemble de départ $\mathbb{S}$.

En pratique, l'application de l'algorithme d'inversion ensembliste **SIVIA** ne se limite pas qu'aux 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 $[\mathbf{x}_j]$ aux instants de mesures t_j [76, 82]. En effet, ces pavés peuvent être calculés par une des méthodes d'intégration numérique garantie présentées dans le deuxième, le troisième et le quatrième chapitres. Ainsi, l'utilisation conjointe de **SIVIA** avec les méthodes d'intégration numérique garantie permet de partitionner l'espace initial borné de recherche des paramètres $\mathbb{P} = [\mathbf{p}]$, selon un test d'inclusion que nous définissons ci-après, en trois listes de pavés. La première liste $\mathcal{L}_a$ contient les sous pavés $[\mathbf{p}_i] \subset [\mathbf{p}]$ dont le test d'inclusion est vrai, c'est-à-dire :

$$\forall t_j \in \{t_1, t_2, ..., t_{nT}\}, \quad [\mathbf{g}]([\mathbf{x}_j], [\mathbf{p}_i], t_j) \subseteq [\mathbf{y}_j] \tag{V.5}$$

L'union des pavés de cette première liste, $\mathcal{L}_a$, donne l'approximation intérieure, $\underline{\mathbb{S}}$, de l'ensemble $\mathbb{S}$. Elle vérifie $\underline{\mathbb{S}} \subseteq \mathbb{S}$.

La seconde liste, $\mathcal{L}_r$, est composée de tous les sous-pavés dont le test d'inclusion est faux, c'est-à-dire les sous-pavés pour lesquels la structure du modèle, les données expérimentales et les bornes d'erreurs sont incohérentes, c'est-à-dire :

$$\exists t_j \in \{t_1, t_2, ..., t_{nT}\}, \quad [\mathbf{g}]([\mathbf{x}_j], [\mathbf{p}_i], t_j) \cap [\mathbf{y}_j] = \emptyset. \tag{V.6}$$

Dans les situations où on ne peut pas conclure quant au résultat du test d'inclusion, le sous-pavé est indéterminé. Dans ce cas ni la condition (V.5) ni la condition (V.6) ne sont satisfaites. Pour lever cette ambiguïté, l'algorithme **SIVIA** bissecte ces sous pavés puis

teste de nouveau les sous pavés résultants. Le choix de la direction de bissection est basé par exemple sur la taille du pavé à partitionner [68]. Nous réitérons ce processus jusqu'à un seuil ϵ fixé *a priori*. Ainsi, la troisième liste, notée $\mathcal{L}_i$, contient l'ensemble des sous-pavés indéterminés et de taille inférieure à ϵ.

Pour résumer le principe de fonctionnement de cette approche d'identification ensembliste des systèmes dynamiques continus, nous proposons ci-dessous un algorithme qui génère les trois listes $\mathcal{L}_a$, $\mathcal{L}_r$ et $\mathcal{L}_i$:

Algorithme : Identification_Ensembliste

(Entrées : $[\mathbf{x}_0], [\mathbf{p}], \mathbf{f}, \mathbf{g}, \mathcal{L}_m, \epsilon$; Sorties : $\mathcal{L}_a, \mathcal{L}_r, \mathcal{L}_i$)

1. $\mathcal{L}_c \leftarrow [\mathbf{p}]$;
2. n_m :=**Size**$(\mathcal{L}_m)$;
3. while $(\mathcal{L}_c \neq \emptyset)$ do
4. $\quad [\mathbf{p}] := pop(\mathcal{L}_c)$;
5. $\quad inclusion := 0$;
6. $\quad$ while $(\mathcal{L}_m \neq \emptyset)$ do
7. $\quad\quad \{t_{mj}, [\mathbf{y}_{mj}]\} := pop(\mathcal{L}_m)$;
8. $\quad\quad [\mathbf{x}_j]$:=**Validated_Integration**$(t_j, t_{j-1}, [\mathbf{x}_{j-1}], [\mathbf{p}])$;
9. $\quad\quad [\mathbf{y}_j] := [\mathbf{g}]([\mathbf{p}], [\mathbf{x}_j])$;
10. $\quad\quad$ test :=**Inclusion_Test**$([\mathbf{y}_j], [\mathbf{y}_{mj}])$;
11. $\quad\quad$ If (test = false) Then
12. $\quad\quad\quad \mathcal{L}_r \leftarrow [\mathbf{p}]$;
13. $\quad\quad\quad$ go to line (3) ;
14. $\quad\quad$ End If
15. $\quad\quad$ Else If (test = true) Then
16. $\quad\quad\quad inclusion$:=$inclusion$+ 1 ;
17. $\quad\quad$ End Else If
18. $\quad$ End while
19. If $(inclusion = n_m)$ Then

20. $\mathcal{L}_a \leftarrow [\mathbf{p}]$;
21. $\underline{\mathbb{S}} := \underline{\mathbb{S}} \cup [\mathbf{p}]$;
22. End If
23. Else If $(w([\mathbf{p}]) \geq \epsilon)$ Then
24. $\{[\mathbf{p}]_l, [\mathbf{p}]_r\} :=$ **Partition**$([\mathbf{p}])$;
25. $\mathcal{L}_c \leftarrow [\mathbf{p}]_l$;
26. $\mathcal{L}_c \leftarrow [\mathbf{p}]_r$;
27. End Else If
28. Else
29. $\mathcal{L}_i \leftarrow [\mathbf{p}]$;
30. $\overline{\mathbb{S}} := \underline{\mathbb{S}} \cup [\mathbf{p}]$;
31. End If
32. End while.

Avant l'exécution de l'algorithme **Identification_Ensembliste**, les domaines admissibles des erreurs de sorties à chaque instant de mesure doivent être enregistrés dans une liste $\mathcal{L}_m$. La liste $\mathcal{L}_c$ contient tous les sous-pavées de $[\mathbf{p}]$ qui ne sont pas encore traités et elle est initialisée dans la ligne 1 par le domaine initial de recherche de paramètres. Tant que cette liste n'est pas vide, le dernier sous-pavés entrant sera extrait dans la ligne 4, ensuite les lignes de code 6 à 10 consistent à réaliser à chaque instant de mesure t_{mj} les trois étapes suivantes : (1) Reconstruire un encadrement garanti du vecteur d'état $[\mathbf{x}_j]$ du système défini par $\mathbf{f}$ avec le sous-pavé $[\mathbf{p}]$ comme paramètres (ligne 8). En effet, la fonction **Validated_Integration** implémente une des méthodes d'intégration numérique garantie ; (2) Calculer un encadrement garanti de toutes les solutions possibles $[\mathbf{y}_j]$ (ligne 9) en utilisant une fonction d'inclusion de $\mathbf{g}$; (3) Vérifier le test d'inclusion (V.5) (ligne 10). Si pour un instant de mesure la fonction **Inclusion_Test** retourne *false*, alors le sous-pavé actuel $[\mathbf{p}]$ sera mis dans la liste $\mathcal{L}_r$ (ligne 12), suivi par un saut vers la ligne 3. Par ailleurs, si l'indicateur *inclusion* est égal à n_m, la taille de la liste $\mathcal{L}_m$, alors le sous-pavé courant est empilé dans la liste $\mathcal{L}_a$ (ligne 20). Si la taille de $[\mathbf{p}]$ est supérieure ou égale à ϵ, alors il sera bissecté dans la ligne 24 et les sous-pavés résultants seront mis dans la liste $\mathcal{L}_c$ (lignes

25 et 26). En revanche, si $[\mathbf{p}]$ est de taille inférieure à ϵ, il sera enregistré dans la liste $\mathcal{L}_i$. Enfin, le calcul de $\underline{\mathbb{S}}$ et $\overline{\mathbb{S}}$ se fait progressivement dans les lignes 21 et 30 respectivement.

V.4 Application

Dans cette application, le problème à résoudre est l'identification des propriétés thermiques (diffusivité et conductivité) d'un matériau dans un contexte ensembliste. Le principe du banc d'essai développé au CERTES (voir figure V.2, [12]), consiste à placer un échantillon de conductivité et de diffusivité inconnues entre deux plaques métalliques. La plaque de la face avant est soumise à un signal d'excitation thermique, celle de la face arrière est en contact avec l'air à température ambiante. Le contact thermique entre les différents éléments est assuré par une couche de graisse à haute conductivité thermique. L'ensemble du dispositif est placé dans une enceinte sous vide de forme cylindrique. La température est prélevée sur la plaque arrière à l'aide d'un thermocouple. Une représen-

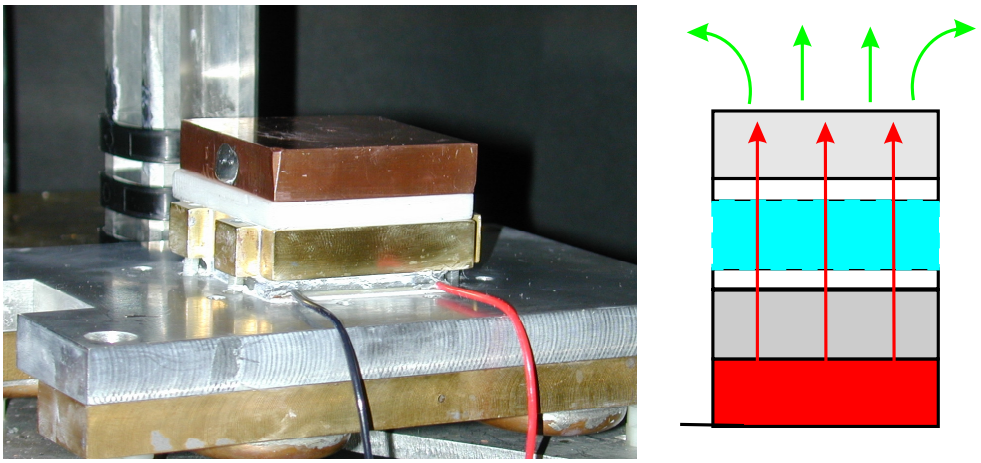


Fig. V.2 *: Banc d'essai thermique*

tation d'état de ce dispositif est obtenue par une discrétisation spatiale de l'équation de la chaleur. Les techniques de discrétisation spaciale sont suffisamment maîtrisées pour que l'on puisse considérer que l'erreur introduite par les schémas numériques est bien plus faible que toutes les autres. La non-influence de la discrétisation sur les résultats peut être vérifiée *a posteriori* en faisant varier le nombre de nœuds de discrétisation.

Ce problème d'identification a déjà été abordé sans succès par Raïssi dans le cadre de sa thèse [82]. D'après Raïssi, l'effet d'enveloppement et le phénomène de dépendance

liés au calcul par intervalles conduisent à une divergence de la taille des pavés, contenant les trajectoires du vecteur d'état aux instants de mesure, obtenus par les méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalles. Dans la suite, avant de procéder à l'identification, nous allons faire une étude comparative entre les performances de l'intégration numérique garantie du modèle de ce dispositif d'identification thermique obtenues par trois méthodes :

1. HOI : Méthode de Hermite-Obreschkoff intervalle avec factorisation **QR** telle que implémentée dans le logiciel VNODE.
2. HOIS : Méthode de Hermite-Obreschkoff intervalle sans pré-conditionnement matriciel. Pour la mettre en œuvre, nous avons modifié le code source du logiciel VNODE.
3. TCID : Méthodes basées sur les théorèmes de comparaison des inégalités différentielles.

Ainsi, nous montrons comment éviter les problèmes rencontrés par Raïssi [82] et résoudre avec succès ce problème d'identification dans le contexte à erreurs bornées.

V.4.1 Modèle

Le modèle d'état du dispositif d'identification présenté sur la figure V.2 est donné par le système d'équations différentielles ci-dessous. Pour réduire le pessimisme qui pourrait être engendré par l'effet de dépendance, nous avons pris soin de minimiser le nombre d'occurrences de chaque paramètre intervalle dans chaque composante du champ de vecteurs lors de l'écriture de ce modèle

$$\begin{cases} \dot{x}_1 &= \alpha_1(x_2 - 2x_1 + u_1(t)) \\ \dot{x}_2 &= 2\alpha_1(x_1 - (1 + \frac{\rho_1}{\rho_2})x_2 + \frac{\rho_1}{\rho_2}x_3) \\ \dot{x}_3 &= 2p_1(x_4 - x_3 + p_2\frac{\delta_2}{\rho_2}(x_2 - x_3)) \\ \dot{x}_i &= p_1(x_{i+1} - 2x_i + x_{i-1}) \quad i = 4, \ldots, 9 \\ \dot{x}_{10} &= 2p_1(x_9 - x_{10} + p_2\frac{\delta_2}{\rho_2}(x_{11} - x_{10})) \\ \dot{x}_{11} &= 2\alpha_2(x_{12} - (1 + \frac{\rho_3}{\rho_2})x_{11} + \frac{\rho_3}{\rho_2}x_{10}) \\ \dot{x}_{12} &= \alpha_2(x_{13} - 2x_{12} + x_{11}) \\ \dot{x}_{13} &= 2\alpha_2(x_{12} - (1 + \frac{\rho_3}{\rho_4})x_{13} + \frac{\rho_3}{\rho_4}u_2(t)) \end{cases} \tag{V.7}$$

où le vecteur d'état $\mathbf{x}(t)$ est de dimension 13 qui représente les températures aux différents nœuds. Les paramètres connus de (V.7) sont $\alpha_1 = 5.44s^{-1}$, $\frac{\rho_1}{\rho_2} = 0.18$, $\frac{\delta_2}{\rho_2} = 2.3Wm^{-1}K^{-1}$,

$\alpha_2 = 5.7s^{-1}$, $\frac{\rho_3}{\rho_2} = 0.09$, $\frac{\rho_3}{\rho_4} = 10^{-4}$. Le vecteur des paramètres à identifier est $\mathbf{p} = (p_1\ p_2)^T$ où $p_1 = \frac{a}{\Delta l^2}$ et $p_2 = \frac{1}{\lambda}$ avec a, λ et Δl qui représentent respectivement la diffusivité thermique, la conductivité thermique et le pas de la discrétisation spatiale. Le vecteur d'entrées est $\mathbf{u}(t) = (u_1(t)\ u_2(t))^T$ et le modèle de la sortie est :

$$y(t) = x_{12}(t). \tag{V.8}$$

Les mesures sont prélevées à des intervalles de temps réguliers tel que $\triangle t = 5s$. Ce choix de pas de mesure n'est pas obligatoire. En effet, nous pouvons résoudre ce problème avec des instants de mesure beaucoup plus éloignés. La durée totale de l'expérience est de $500s$ et le domaine initial de recherche de paramètres est choisi en tenant compte des valeurs physiquement admissibles pour les propriétés thermo-physiques $\mathbb{P} = [0,\ 3]s^{-1} \times [0,\ 3]mKW^{-1}$. Les bornes de l'erreur de sortie sont fixées à $\mathbb{E}_j = [-0.05,\ +0.05]°C$; ces bornes représentent un majorant des bruits de mesure et des erreurs de modélisation.

Dans la suite, nous montrons comment résoudre ce problème pour deux types de signaux d'excitation : un échelon et un signal harmonique.

V.4.2 Signal d'excitation du type échelon

Pour cette première expérience, nous considérons que les températures initiales aux différents nœuds sont $x_i = 10°C, i = 1, ..., 13$ et $u_2(t) = 10°C$. Le signal d'excitation $u_1(t)$ est un échelon d'amplitude $15°C$.

$$u_1(t) = \begin{cases} 10°C \text{ si } t = 0 \\ 15°C \text{ si } t > 0 \end{cases} \tag{V.9}$$

V.4.2.1 Image directe

La taille des pavés $[y](t_j)$ qui contiennent toutes les valeurs possibles de la sortie (V.8) aux instants de mesures t_j, joue un rôle déterminant pour le succès de l'algorithme **Identification_Ensembliste** appliqué au système (V.7). Pour cela, nous testons les performances des trois méthodes d'intégration numérique garantie HOI, HOIS et TCID pour choisir la meilleure, en terme de précisons des pavés solutions obtenus et de temps de calcul nécessaire, à implémenter dans la fonction **Validated_Integration** de l'algorithme **Identification_Ensembliste**. Pour réaliser cette comparaison, nous utilisons le logiciel VNODE [71] avec une stratégie de contrôle du pas d'intégration (pas variable).

V.4.2.1.1 Simulation par la méthode HOI

Dans cette section, nous simulons le système (V.7) par la méthode HOI pour différentes tailles du pavé **p** des paramètres incertains. Les résultats de ces simulations sont présentés sur les figures V.3, V.4 et V.5. En effet, ces figures montrent les encadrements

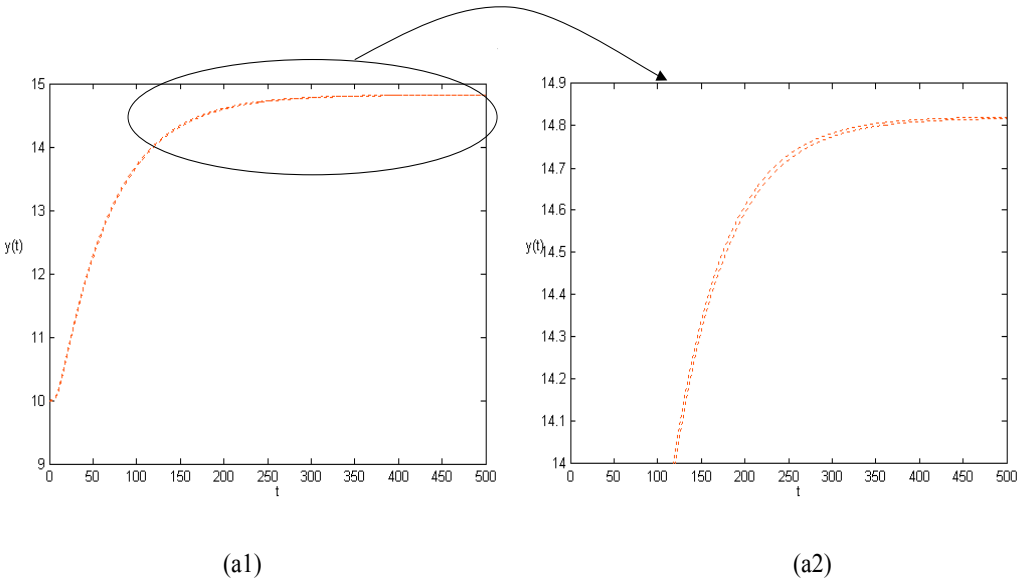


Fig. V.3 *: L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes à un pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.981] \times [0.5, 0.501]$. *Temps CPU=131s*

de toutes les sorties possibles de (V.7) correspondants aux différents pavés de paramètres $[\mathbf{p}]$. Comme prévu, la taille de ces encadrements de la sortie (V.8) divergent lorsque la taille du pavé de paramètres est supérieure à 0.003 (voir figures V.5 (d) et V.5 (e)). Cela nous oblige à réaliser les tests d'inclusion (V.5) et (V.6) en intégrant l'EDO (V.7) avec des pavés de paramètres de taille inférieure à 0.003 voire égale à 0.002 (voir figure V.4 (b)). Par ailleurs, étant donné que le temps de calcul pour réaliser cette intégration sur un PIII 1.2MHz est d'environ 150 secondes, alors le temps nécessaire à l'algorithme **Identification_Ensembliste** pour qu'il puisse traiter l'ensemble des sous-pavés du domaine initial des paramètres $\mathbb{P} = [0,\ 3]s^{-1} \times [0,\ 3]mKW^{-1}$ est très élevé (de l'ordre de plusieurs mois de calcul).

Etant donné que le système (V.7) est coopératif, c'est-à-dire que tous les éléments non diagonaux de son Jacobien sont non-négatifs, alors le pré-conditionnement matriciel basé sur la factorisation **QR** n'est pas adapté au contrôle de la propagation du pessimisme

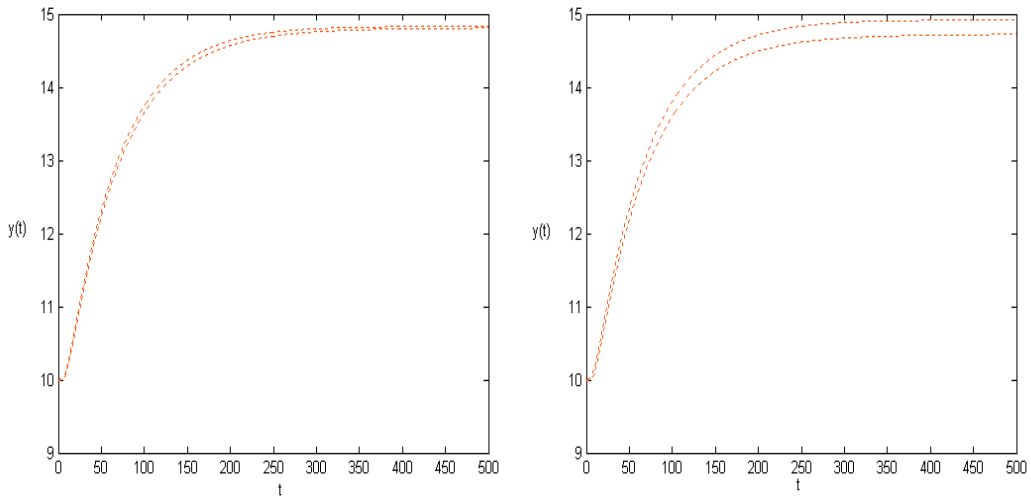


Fig. V.4 *: L'encadrement de toutes les sorties possibles $[y](t)$: (b) le pavé de paramètres est* $[\mathbf{p}] = [0.98, 0.982] \times [0.5, 0.502]$*, temps CPU=145s ; (c) le pavé de paramètres est* $[\mathbf{p}] = [0.98, 0.983] \times [0.5, 0.503]$*, temps CPU=157s.*

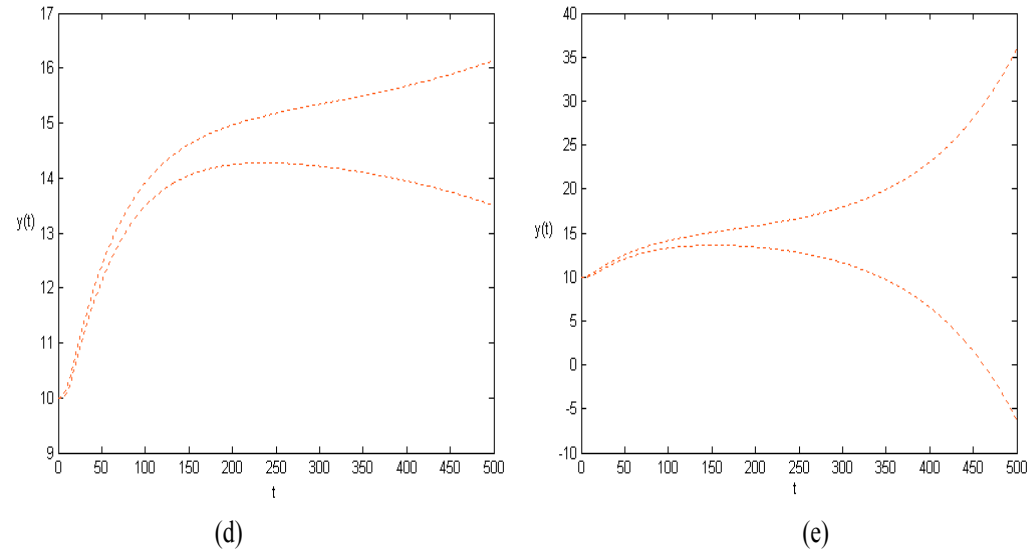


Fig. V.5 *: L'encadrement de toutes les sorties possibles $[y](t)$: (d) le pavé de paramètres est* $[\mathbf{p}] = [0.98, 0.984] \times [0.5, 0.504]$*, temps CPU=170s ; (e) le pavé de paramètres est* $[\mathbf{p}] = [0.98, 0.985] \times [0.5, 0.505]$*, temps CPU=190s.*

(voir section II.3.2.6). Pour cela, nous testons dans la section suivante les performances de la méthode HOIS et nous les comparons à celles obtenues par la méthode HOI.

V.4.2.1.2 Simulation par la méthode HOIS

Tout d'abord, nous avons simulé le système (V.7) par la méthode HOIS avec les mêmes pavés de paramètres que ceux utilisés pour tester les performances de la méthode HOI. Nous présentons sur les figures ci-dessous les encadrements de toutes les solutions possibles (V.8) relatives à chaque pavé de paramètres.

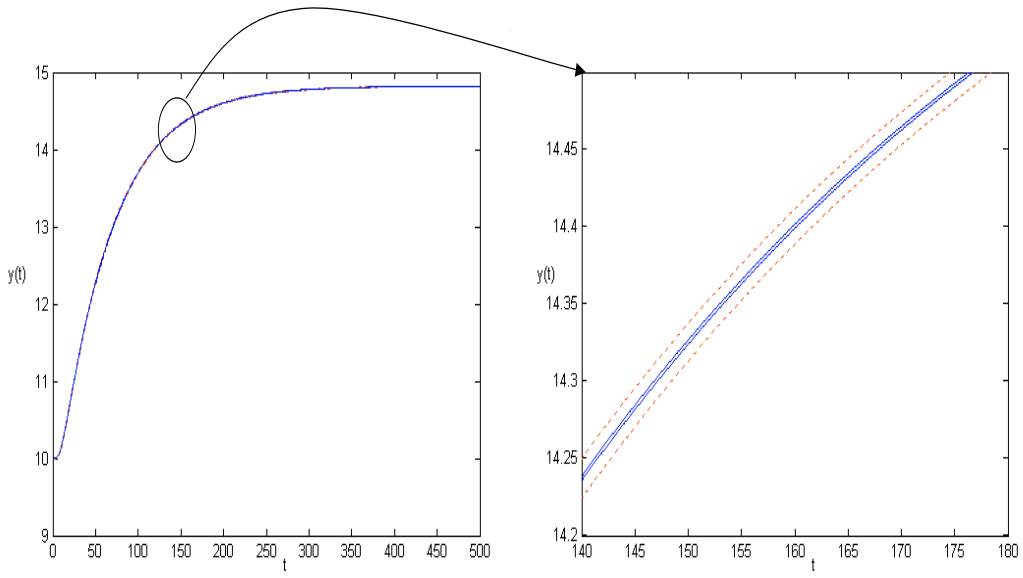


FIG. V.6 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.981] \times [0.5, 0.501]$. Courbes bleues continues méthode HOIS, temps CPU=42.7s ; Courbes rouges discontinues méthode HOI.*

Les courbes bleues en ligne continue sur les figures V.6 à V.10 montrent bien l'avantage de la méthode HOIS par rapport à la méthode HOI dans le cas des systèmes coopératifs. En effet, la méthode HOIS peut simuler avec succès l'EDO (V.7) avec des pavés de paramètres de taille égale à 0.013, c'est-à-dire 4 fois plus grande que celle permise par la méthode HOI. De plus, son temps de calcul est beaucoup plus petit que celui de la méthode HOI. Dans le tableau V.1, nous comparons les temps de calcul obtenus pour les deux méthodes pour les cas où la taille de l'encadrement de la sortie (V.8) ne diverge pas. Par comparaison des valeurs de ce tableau, il apparaît clairement que le temps de calcul de la méthode HOIS

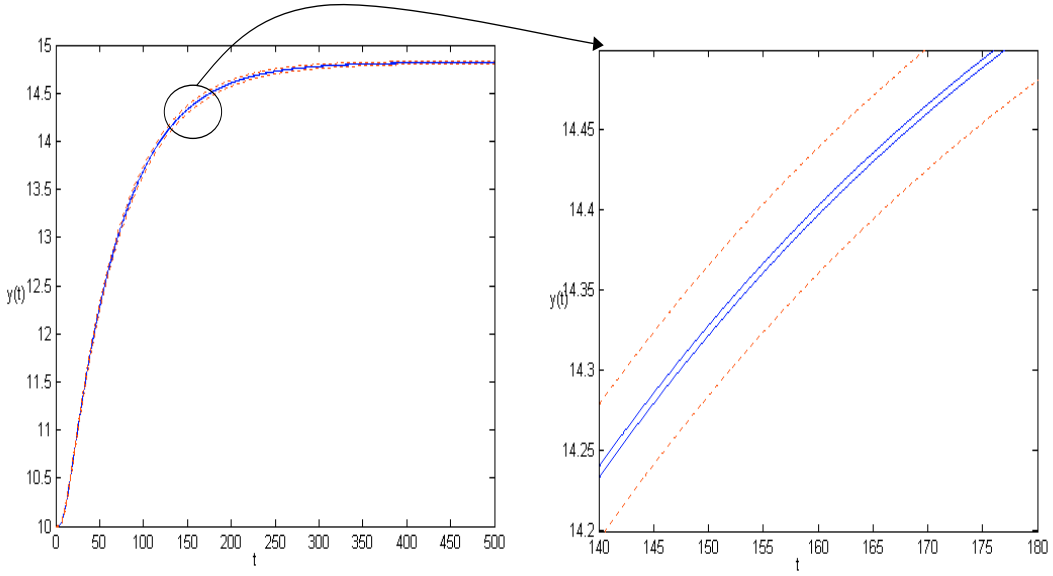


Fig. V.7 : *L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes à un pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.982] \times [0.5, 0.502]$. *Courbes bleues continues méthode HOIS, temps CPU=43.4s ; Courbes rouges discontinues méthode HOI.*

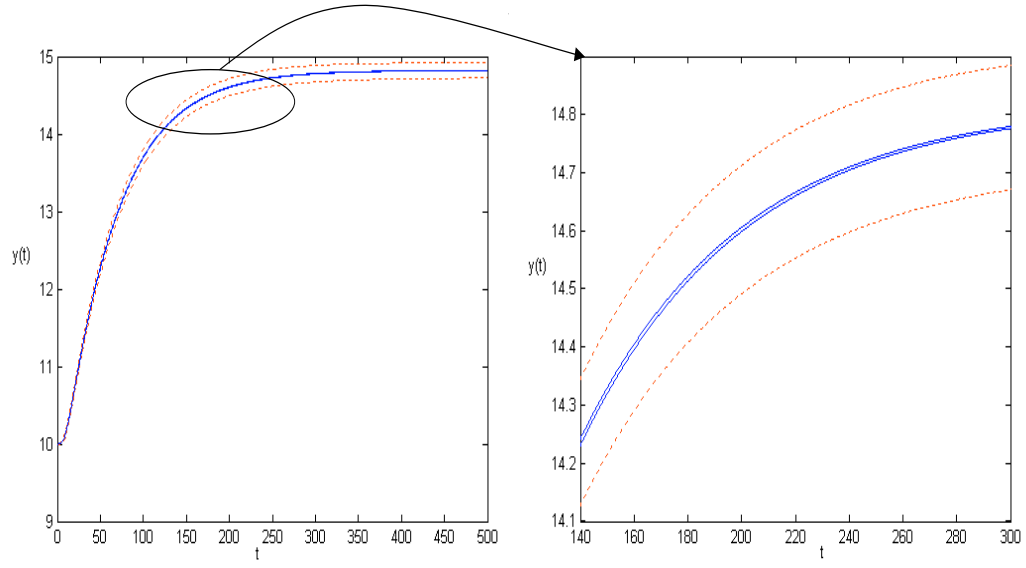


Fig. V.8 : *L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes à un pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.983] \times [0.5, 0.503]$. *Courbes bleues continues méthode HOIS, temps CPU=44.8s ; Courbes rouges discontinues méthode HOI.*

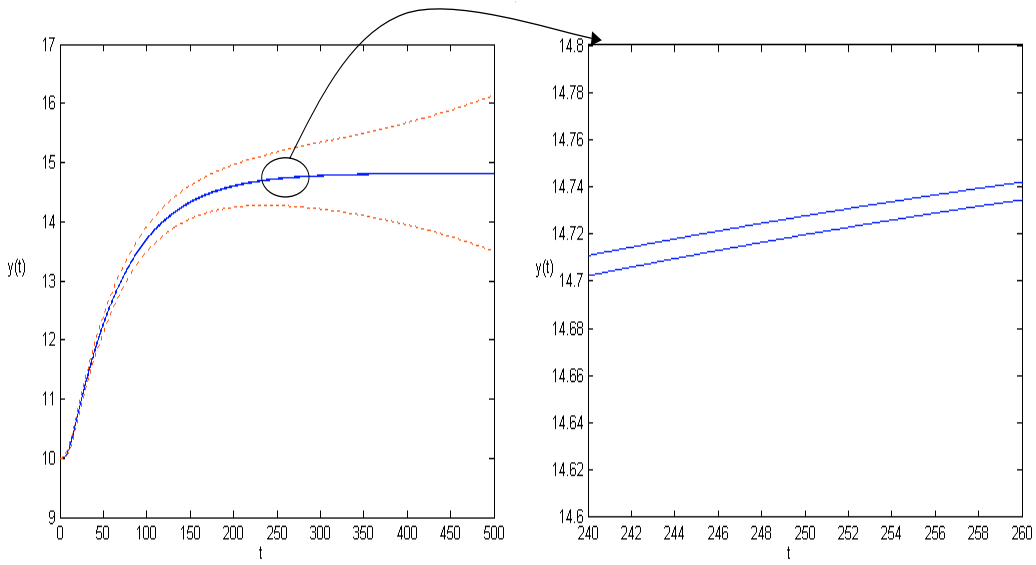


FIG. V.9 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.984] \times [0.5, 0.504]$. Courbes bleues continues méthode HOIS, temps CPU=45.6s ; Courbes rouges discontinues méthode HOI.*

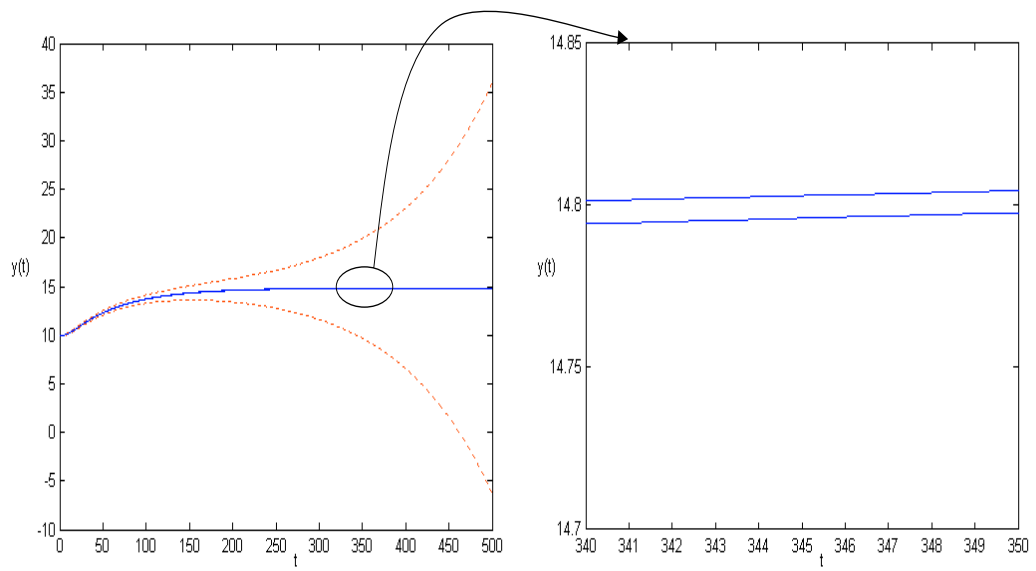


FIG. V.10 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.985] \times [0.5, 0.505]$. Courbes bleues continues méthode HOIS, temps CPU=48.1s ; Courbes rouges discontinues méthode HOI.*

$w([\mathbf{p}])$	0.001	0.002	0.003
CPU (HOI)	131s	145s	157s
CPU (HOIS)	42.7s	43.4s	44.8s

Tab. V.1 *: Temps CPU sur un PIII 1.2MHz*

est environ 4 fois inférieur à celui de la méthode HOI.

Néanmoins, malgré les avantages de la méthode HOIS par rapport à la méthode HOI, son utilisation dans l'algorithme **Identification_Ensembliste** nécessite un temps de calcul élevé pour identifier les paramètres inconnus du système (V.7). Cela est dû au fait que la taille des pavés de paramètres, pour laquelle l'encadrement de la sortie (V.8) donné par la méthode HOIS ne diverge pas, est faible. En effet, pour intégrer l'EDO (V.7) par la méthode HOIS avec succès, la taille du pavé de paramètres $[\mathbf{p}]$ ne doit pas excéder 0.012. Ceci nécessite un nombre très élevé de bissection du domaine initial de paramètres $\mathbb{P}$ dans l'algorithme **Identification_Ensembliste**, ce qui engendre un temps de calcul important.

Pour réduire ce nombre de bissection et améliorer le temps de calcul, nous nous intéressons dans la section suivante aux méthodes d'intégration numérique garantie basées sur la théorie de comparaison des inégalités différentielles, en particulier dans le cas des systèmes coopératifs.

V.4.2.1.3 Simulation par la méthode TCID

Comme le système (V.7) est coopératif, la connaissance des signes de ses dérivées partielles par rapport aux paramètres p_1 et p_2 est essentielle pour construire les systèmes englobants. Pour ce système, nous avons deux types de dérivées partielles :

1. dérivées partielles par rapport à p_1 :

$$g_{3,1}(.) = \frac{\partial f_3(.)}{\partial p_1} = 2(x_4 - x_3 + p_2\frac{\delta_2}{\rho_2}(x_2 - x_3)) \quad \text{(V.10)}$$

$$g_{i,1}(.) = \frac{\partial f_i(.)}{\partial p_1} = x_{i+1} - 2x_i + x_{i-1},\ i \in \{4, ..., 9\} \quad \text{(V.11)}$$

$$g_{10,1}(.) = \frac{\partial f_1 0(.)}{\partial p_1} = 2(x_9 - x_{10} + p_2\frac{\delta_2}{\rho_2}(x_{11} - x_{10})) \quad \text{(V.12)}$$

2. dérivées partielles par rapport à p_2 :

$$g_{3,2}(.) = \frac{\partial f_3}{\partial p_2} = 2p_1\frac{\delta_2}{\rho_2}(x_2 - x_3) \tag{V.13}$$

$$g_{10,2}(.) = \frac{\partial f_{10}}{\partial p_2} = 2p_1\frac{\delta_2}{\rho_2}(x_{11} - x_{10}). \tag{V.14}$$

Etant donné que le signal d'excitation (V.9) est positif et les dérivées par rapport au temps des variables d'état de (V.7) à l'instant initial $t_0 = 0$ sont $\dot{x}_1(0) > 0$ et $\dot{x}_i(0) = 0$ pour tout $i \in \{2, ...13\}$, alors compte tenu de la **Propriété 3** des systèmes coopératifs nous pouvons affirmer que :

$$\forall t \geq 0,\ \forall i \in \{1, ..., 13\},\ \dot{x}_i(t) \geq 0. \tag{V.15}$$

Donc, comme $p_1 \in \mathbb{P} \subset \mathbb{R}^+$, alors nous pouvons déduire que toutes les dérivées partielles de (V.7) par rapport à p_1 sont positives pour tout $t \in [0, +\infty[$. Par conséquent, l'application de la **Règle 1** permet de borner les composantes $f_i(.),\ i \in \{3, ..., 10\}$ de (V.7) par rapport à p_1 comme ci-après :

$$\begin{gathered} \forall p_1 \in [p_1] \\ \underline{f}_i(., ., \underline{p}_1, .) \leq f_i(., ., p_1, .) \leq \overline{f}_i(., ., \overline{p}_1, .). \end{gathered} \tag{V.16}$$

Maintenant, il ne reste qu'à borner les composantes d'indice 3 et 10 par rapport à p_2. Cela nécessite la connaissance du signe de $g_{3,2}(.)$ et $g_{10,2}(.)$. Par des considérations thermiques, il est facile de démontrer que le système (V.7) est stable, au sens de Lyapunov, pour tout $\mathbf{p} \in \mathbb{P}$ et qu'il atteint son régime permanent avec $x_i^\infty \geq x_{i+1}^\infty$ aux différents nœuds de la discrétisation. De plus, grâce aux propriétés des systèmes monotones et étant donné que les conditions initiales de (V.7) sont ordonnées $x_i(0) \geq x_{i+1}(0)$, nous pouvons alors en déduire que

$$\forall t \geq 0, \quad x_i(t) \geq x_{i+1}(t). \tag{V.17}$$

Ainsi, le signe de $g_{3,2}(.)$ est toujours positif ou nul. Par contre le signe de $g_{10,2}(.)$ est toujours négatif ou nul. Donc, la **Règle 1** permet de borner les composantes f_3 et f_{10} par rapport à p_2 comme ci-dessous :

$$\begin{gathered} \forall p_2 \in [p_2] \\ \underline{f}_3(., ., \overline{p}_2, .) \leq f_3(., ., p_2, .) \leq \overline{f}_3(., ., \underline{p}_2, .) \\ \underline{f}_{10}(., ., \underline{p}_2, .) \leq f_{10}(., ., p_2, .) \leq \overline{f}_{10}(., ., \overline{p}_2, .). \end{gathered} \tag{V.18}$$

Finalement, l'intégration numérique garantie des systèmes englobants donnés par les doubles inégalités (V.16) et (V.18) fournit un encadrement garanti de taille très raisonnable de toutes les solutions possibles de (V.7) pour tout pavé $[\mathbf{p}]$ de paramètres de taille quelconque. Pour montrer l'intérêt de la méthode TCID d'intégration, nous comparons ses performances par rapport à celles de la méthode HOIS. Pour ce faire, nous simulons le système (V.7) par les deux méthodes pour différents pavés de paramètres.

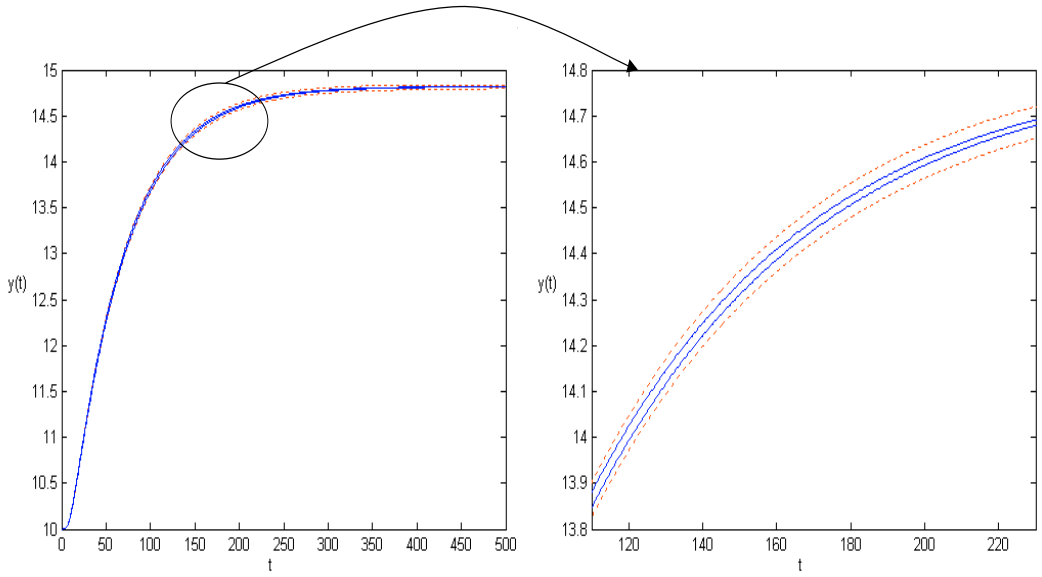


Fig. V.11 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.99] \times [0.5, 0.51]$. Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS, temps CPU=164s .*

Les courbes sur les figures V.11 à V.14 présentent les résultats de ces simulations. Ces courbes montrent clairement que la méthode TCID donne de meilleurs résultats, en terme de précision de l'encadrement de la sortie et de temps de calcul, par rapport à la méthode HOIS. De plus, cette dernière méthode diverge si la taille du pavé de paramètres est supérieure à 0.013. Le tableau V.2 montre les temps de calcul des deux méthodes dans le cas où la taille de l'encadrement de la sortie ne diverge pas. Sur ce tableau, on voit bien que le temps de calcul de la méthode TCID est beaucoup moins important que celui de la méthode HOIS. De plus, il est constant et ne dépend pas de la taille du pavés de paramètres. Ceci est dû au fait que les systèmes englobants de la méthode TCID sont

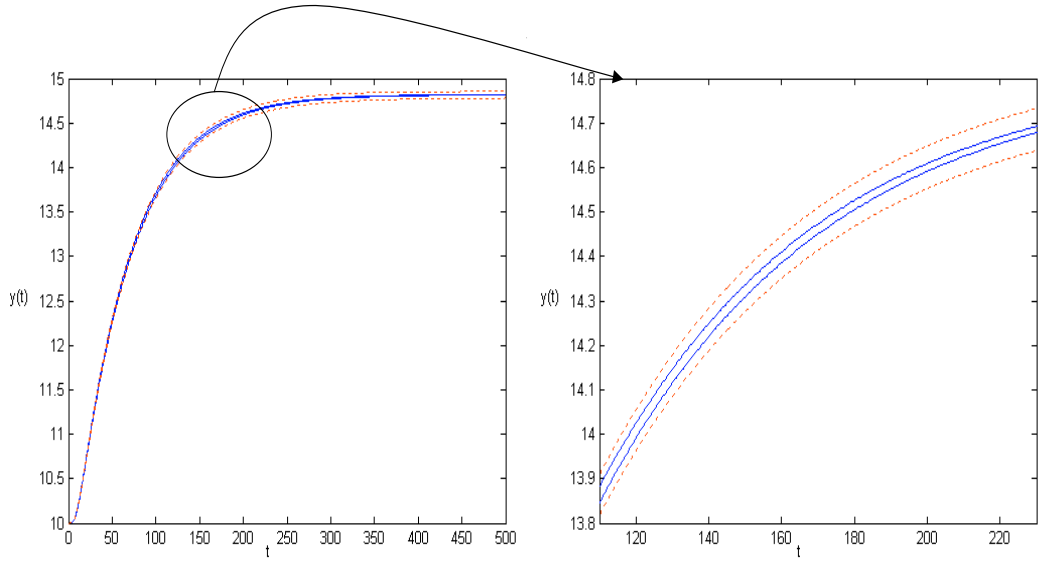


FIG. V.12 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.991] \times [0.5, 0.511]$. Courbes bleues continues méthode HOIS, temps CPU=47s ; Courbes rouges discontinues méthode HOI, temps CPU=173s.*

$w([\mathbf{p}])$	0.01	0.011	0.012	0.013
CPU (HOIS)	164s	173s	219s	304s
CPU (TCID)	47s	47s	47s	47s

TAB. V.2 *: Temps CPU sur un PIII 1.2MHz*

déterministes (ponctuels).

De plus, la méthode TCID résout aussi avec succès ce problème avec des pavés de paramètres de tailles importantes. Par exemple, les figures V.15 (a) et (b) montrent respectivement les encadrement de la sortie (V.8) de (V.7) pour deux pavés de paramètres de taille 0.5 et 1.

Après cette étude comparative entre les trois méthodes d'intégration garantie HOI, HOIS et TCID, nous présentons dans la section suivante les résultats de l'identification ensembliste de (V.7).

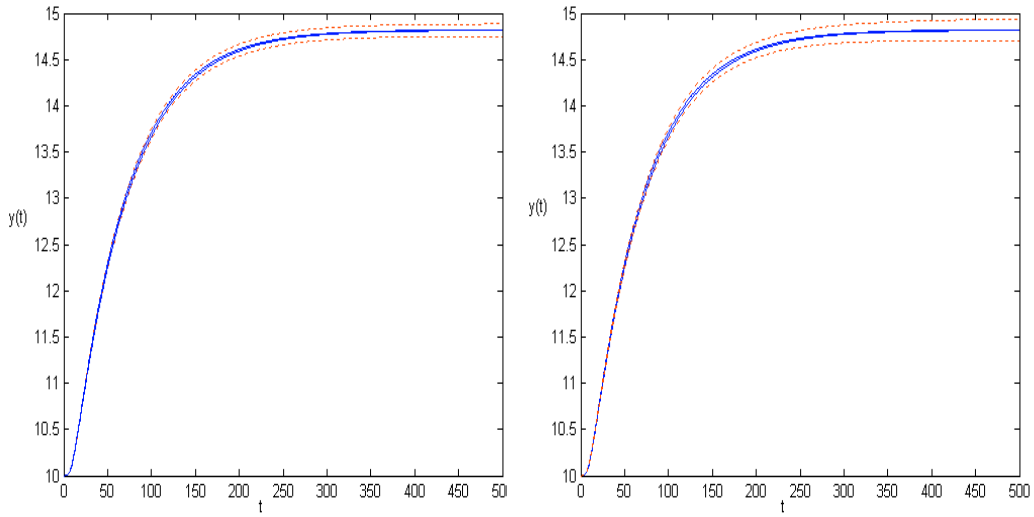


Fig. V.13 *: L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes : (a) pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.992] \times [0.5, 0.512]$. *Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS. temps CPU=219s ; (b) pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.993] \times [0.5, 0.513]$. *Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS. temps CPU= 304s.*

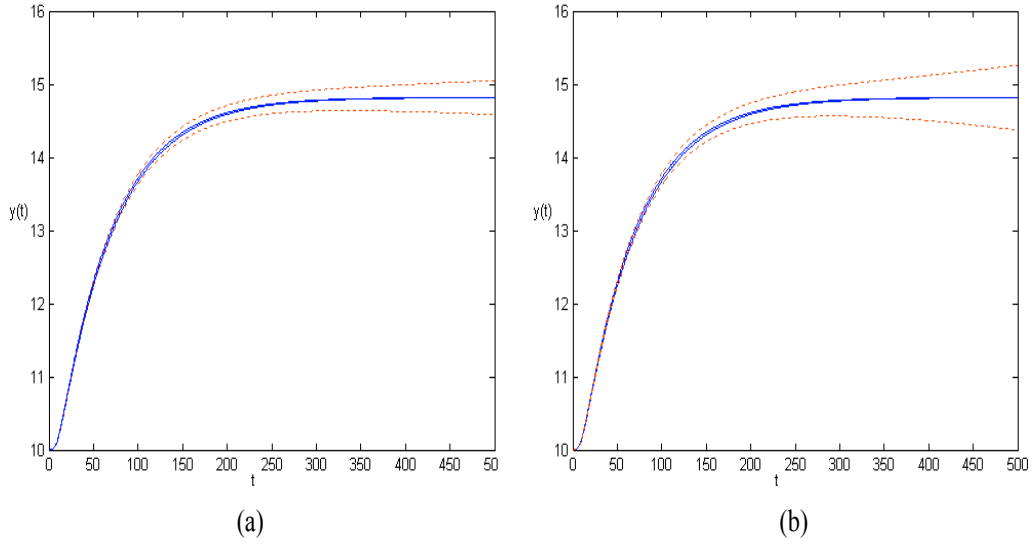


Fig. V.14 *: L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes : (a) pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.994] \times [0.5, 0.514]$. *Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS. temps CPU=368s ; (b) pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.995] \times [0.5, 0.515]$. *Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS. temps CPU= 482s.*

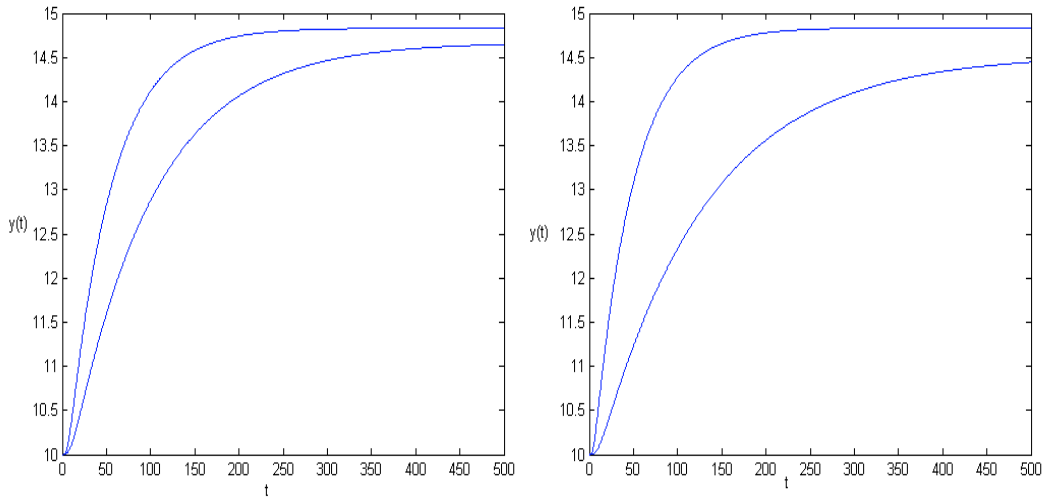


FIG. V.15 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 1.48]\times[0.5, 1]$; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 1.98]\times[0.5, 1.5]$. Temps CPU=47s.*

V.4.2.2 Inversion ensembliste

L'exécution de l'algorithme **Identification_ensembliste**, dont la fonction **Validated_Integration** implémente la méthode TCID et avec un seuil de bissection $\epsilon_1 = 0.01$, donne une caractérisation globale et garantie du continuum solution $\mathbb{S}$ représenté sur la figure V.16. En effet, l'approximation intérieure $\underline{\mathbb{S}}$ est donnée par l'union des sous-pavés rouges et la couche d'incertitude $\triangle\mathbb{S}$ est l'union des sous-pavés noirs. Le temps de calcul relatif à ce seuil de bissection ϵ_1 est d'environ 6 heures sur un PIV 2.4 GHZ.

La figure V.17 présente les mêmes approximations avec une précision beaucoup plus fine $\epsilon_2 = 0.002$ et un temps de calcul plus élevé (environ 15.5 heures). La projection de l'approximation extérieure $\overline{\mathbb{S}} = \underline{\mathbb{S}} \cup \triangle\mathbb{S}$ sur les axes des paramètres donne un intervalle garanti de l'incertitude associée à chaque paramètre identifié :

$$\begin{cases} p_1 \in [0.35,\ 0.62]s^{-1} \\ p_2 \in [0.88,\ 1.13]mKW^{-1}. \end{cases}$$

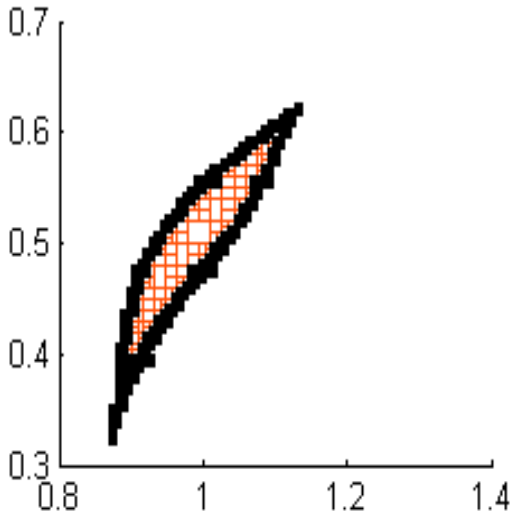


Fig. V.16 *: Approximation intérieure (pavés en rouge) de l'ensemble solution. Couche d'incertitude (pavés en noir), TC=6h.*

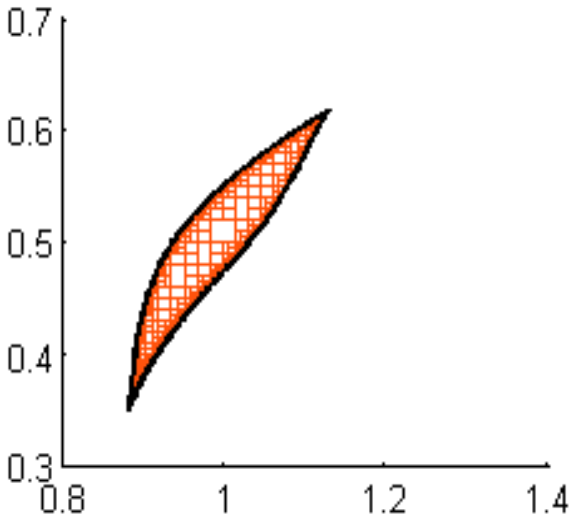


Fig. V.17 *: Approximation intérieure (pavés en rouge) de l'ensemble solution. Couche d'incertitude (pavé en noir), TC=15h.*

V.4.3 Signal d'excitation du type harmonique

Dans cette deuxième expérience, nous utilisons un signal d'excitation harmonique et nous estimons les mêmes paramètres sur le même domaine de recherche que dans la première expérience. De plus, nous considérons les mêmes domaines admissibles pour les erreurs de sorties et les mêmes conditions initiales. Dans ce cas, le signal d'excitation est composé de cinq sinusoïdes de fréquences et d'amplitudes différentes

$$u_1(t) = a_m + \sum_{n=1}^{n=5} a_n \sin(2\pi 2^{n-1} f_0 t) \tag{V.19}$$

où $a_m = u_2 = 10°$, $a_1 = 1.014$, $a_2 = 1.384$, $a_3 = 1.74$, $a_4 = 2.387$, $a_5 = 3.081$ et $f_0 = 5.10^{-4}$Hz.

V.4.3.1 Image directe

V.4.3.1.1 Simulation par les méthodes HOI et HOIS

Pour cette deuxième expérience, nous constatons que les performances de la méthode d'intégration numérique garantie HOIS sont meilleures que celles de la méthode HOI. Effectivement, les figures V.18 à V.21 montrent bien la précision des encadrements de toutes les sorties possibles (V.8) obtenus par la méthode HOIS pour différentes tailles du pavé de paramètres [**p**].

En effet, les courbes en ligne bleue continue (sur ces figures) représentent les encadrements obtenus par la méthode HOIS et les courbes rouges pointillées montrent les encadrement donnés par la méthode HOI.
De plus, le temps de calcul de la méthode HOIS est nettement inférieur à celui de la méthode HOI comme le montre les valeurs du tableau V.3.

$w([\mathbf{p}])$	0.001	0.002	0.003
CPU (HOI)	245s	265s	282s
CPU (HOIS)	98s	120s	178s

Tab. V.3 *: Temps CPU sur un PIII 1.2MHz*

Cependant, comme le temps de calcul pour réaliser ces simulations est très important (voir tableau V.3) et la taille des pavés de paramètres, qui ne conduit pas à la divergence de

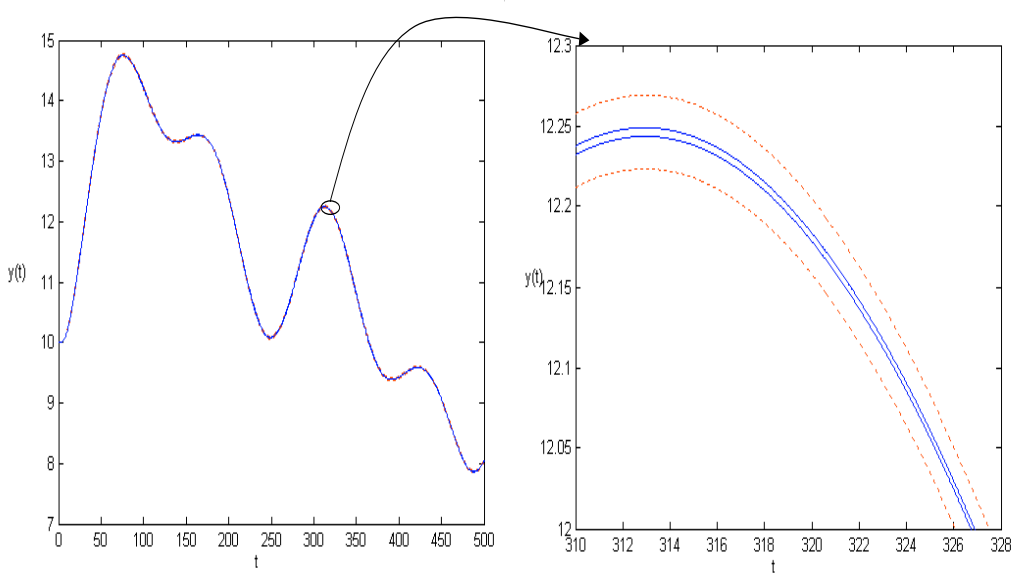


Fig. V.18 : *L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes à un pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.981] \times [0.5, 0.501]$. *Courbes bleues continues méthode HOIS, temps CPU=98s ; Courbes rouges discontinues méthode HOI, temps CPU=245s.*

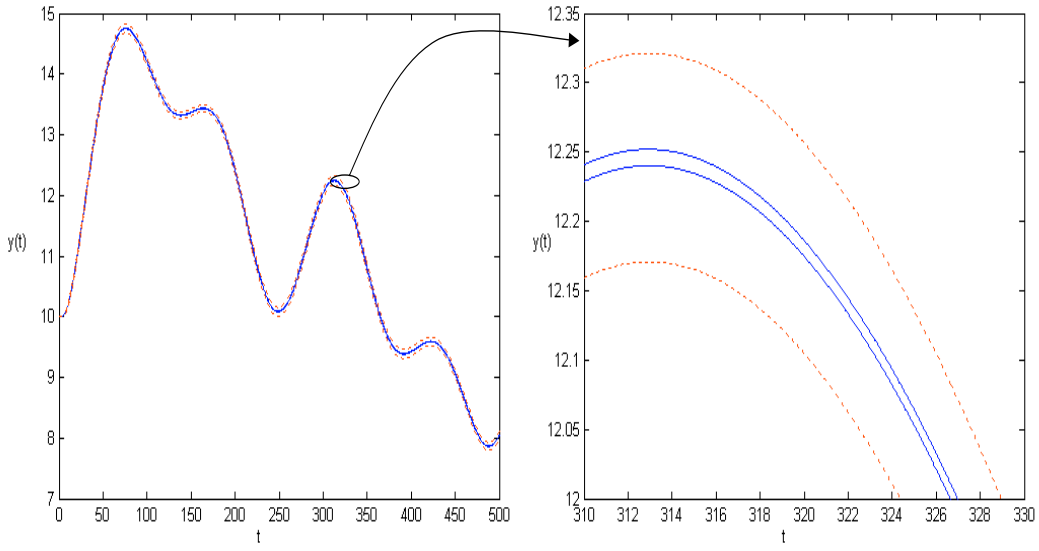


Fig. V.19 : *L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes à un pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.982] \times [0.5, 0.502]$. *Courbes bleues continues méthode HOIS, temps CPU=120s ; Courbes rouges discontinues méthode HOI, temps CPU=265s.*

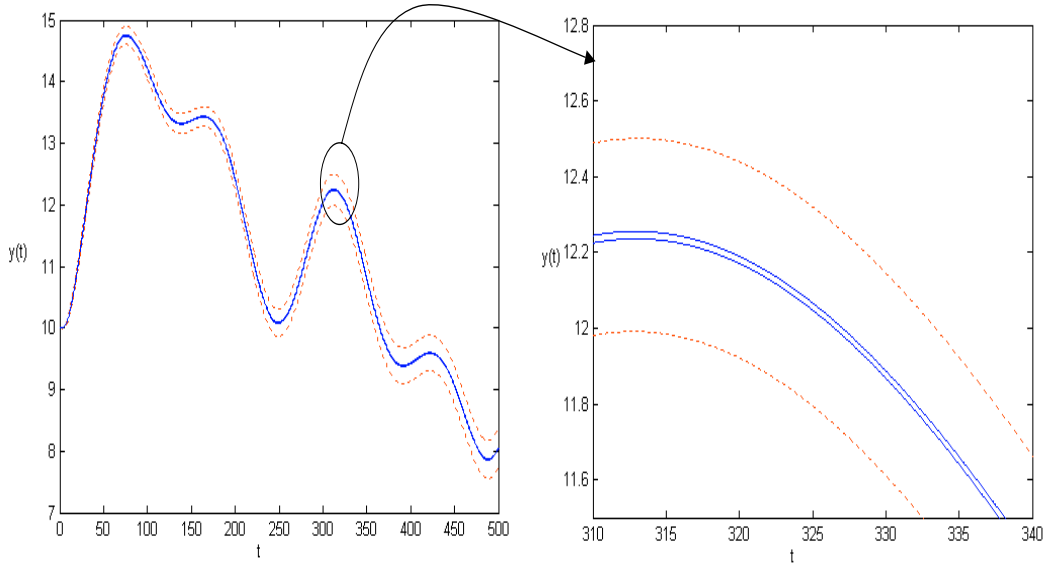
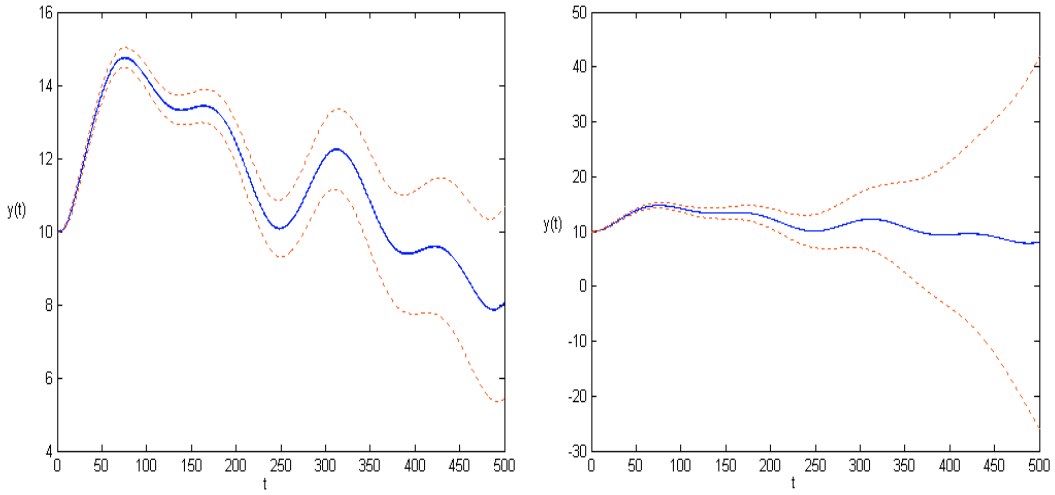


Fig. V.20 *: L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes à un pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.983] \times [0.5, 0.503]$. *Courbes bleues continues méthode HOIS, temps CPU=178s ; Courbes rouges discontinues méthode HOI, temps CPU=282s.*



(a) (b)

Fig. V.21 *: L'encadrement de toutes les sorties possibles* $[y](t)$ *correspondantes : (a) pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.984] \times [0.5, 0.504]$. *Courbes bleues continues méthode HOIS, temps CPU=213s ; Courbes rouges discontinues méthode HOI. temps CPU=300s ; (b) pavé de paramètres* $[\mathbf{p}]^T = [0.98, 0.985] \times [0.5, 0.505]$. *Courbes bleues continues méthode HOIS, temps CPU=240s ; Courbes rouges discontinues méthode HOI. temps CPU= 329s.*

ces deux méthodes, est très faible ; l'algorithme **Identification_Ensembliste** nécessite un temps de calcul très élevé, qui peut être de l'ordre de plusieurs mois de calcul, pour identifier les paramètres du système (V.7). Pour cette raison, nous montrons dans la suite comment, comme pour le cas avec échelon, l'intégration numérique garantie basée sur les théorèmes de comparaison des inégalités différentielles (TCID) permet de réduire considérablement ce temps de calcul.

V.4.3.1.2 Simulation par la méthode TCID

Bien que le système (V.7) soit coopératif, la **Règle 1** ne peut pas être utilisée pour construire les systèmes englobants comme nous l'avons fait pour la première expérience. Cela est dû aux changements de signe des dérivées partielles de (VI.29) par rapport aux paramètres sur la période de l'expérience. Par exemple, la figure V.22 montre l'évolution temporelle de la dérivée partielle $g_{5,1}(.) = \frac{\partial f_5(.)}{\partial p_1}$. Sur cette figure, il est clair que cette dérivée partielle change de signe plusieurs fois sur la période de simulation.

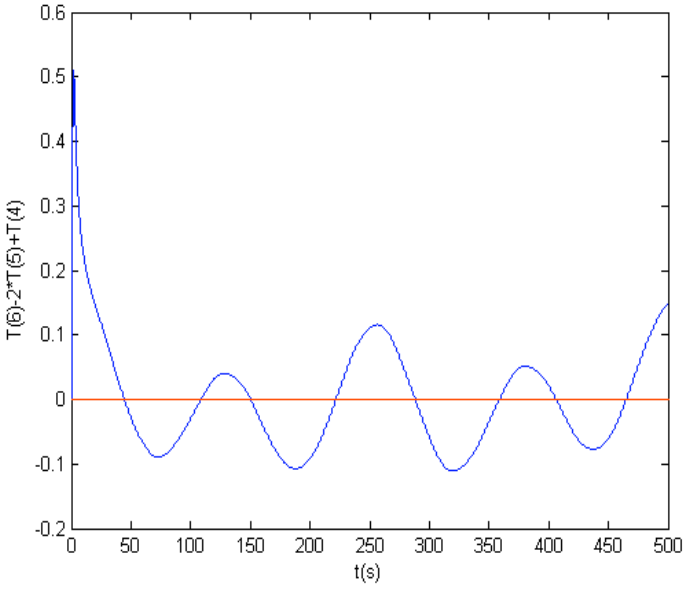


Fig. V.22 *: L'évolution temporelle de la dérivée partielle* $g_{5,1}(.)$ *du système* (V.7)

Cependant, notre méthode hybride d'intégration numérique garantie résumée dans l'algorithme **Hybrid_bound_Solution** présenté dans la section IV.6.2 peut résoudre ce problème. Les figures V.23, V.24 et V.25 illustrent l'avantage de la méthode TCID, qui implémente l'algorithme **Hybrid_bound_Solution**, par rapport à la méthode HOIS.

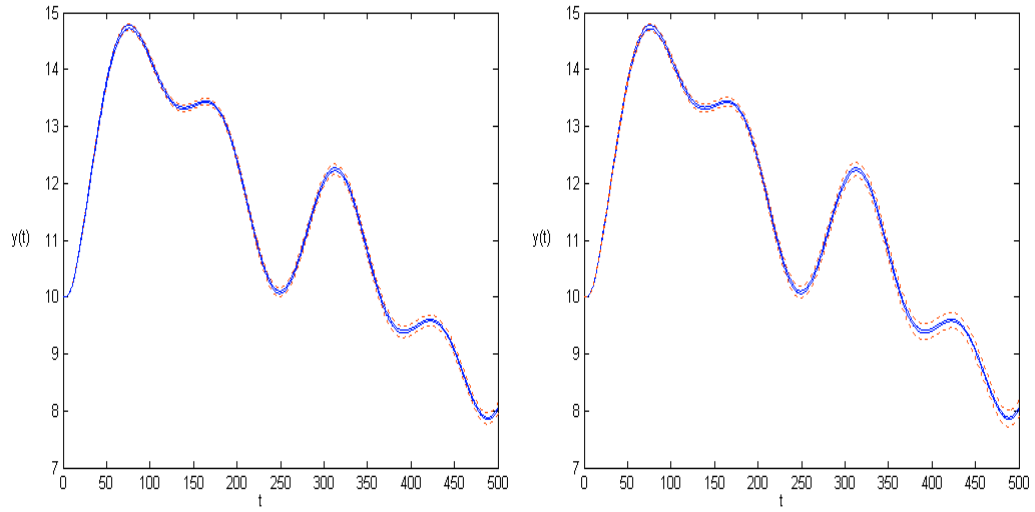


FIG. V.23 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.99] \times [0.5, 0.51]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU=311s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.991] \times [0.5, 0.511]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU= 348s.*

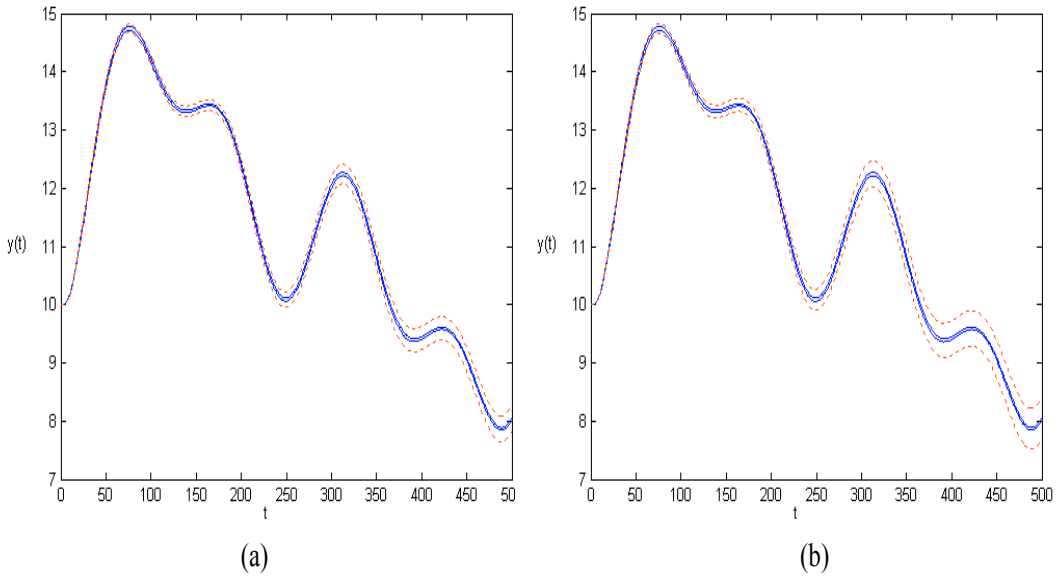


FIG. V.24 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.992] \times [0.5, 0.512]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU=383s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.993] \times [0.5, 0.513]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU=433s.*

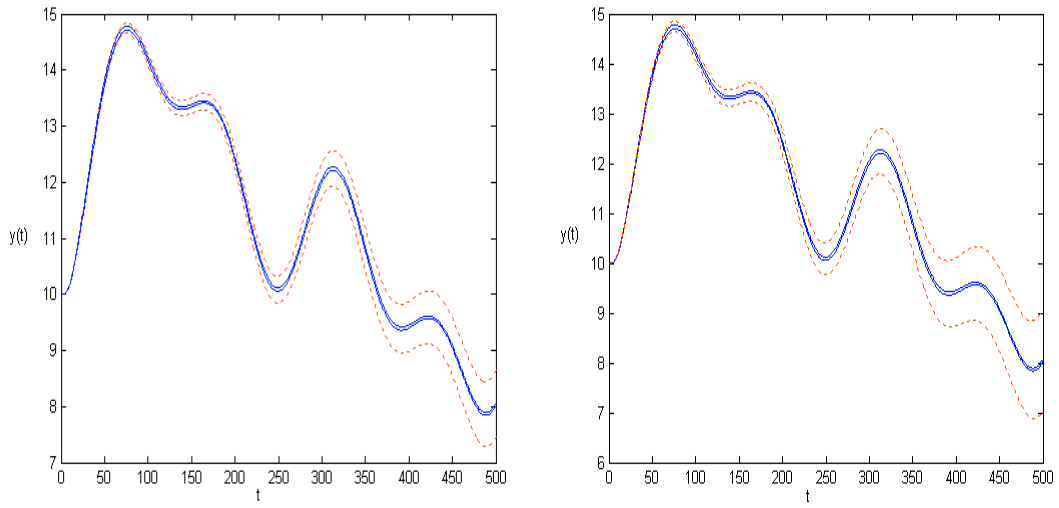


Fig. V.25 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.994] \times [0.5, 0.514]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU=500s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.995] \times [0.5, 0.515]$. Courbes bleues continues méthode TCID, temps CPU=240s ; Courbes rouges discontinues méthode HOIS. temps CPU=578s.*

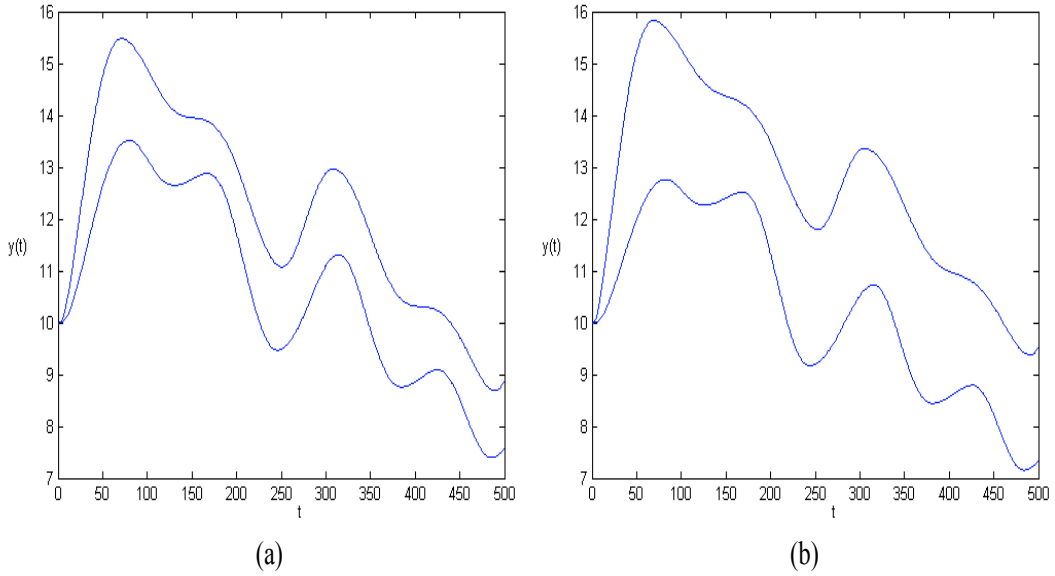


Fig. V.26 *: L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 1.48] \times [0.5, 1]$, temps CPU=200s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 1.98] \times [0.5, 1.5]$, temps CPU=200s ; Courbes rouges discontinues méthode HOI. temps CPU= 329s.*

En effet, on voit bien sur ces figures que la précision des encadrements de sorties obtenus par la méthode TCID est bien meilleure que celle des encadrements de sorties calculés par la méthode HOIS. Par ailleurs, le tableau V.4 montre que les temps de calcul de la méthode TCID sont largement inférieurs à ceux de la méthode HOIS.

$w([\mathbf{p}])$	0.01	0.011	0.012	0.013	0.014
CPU (HOIS)	311s	348s	383s	433s	500s
CPU (TCID)	176s	176s	176s	176s	176s

Tab. V.4 *: Temps CPU sur un PIII 1.2MHz.*

De plus, même pour des pavés de paramètres de taille très large la méthode TCID, basée sur l'algorithme **Hybrid_bound_Solution**, fournit des encadrements de sortie de taille raisonnable. Les figures V.26 (a) et (b) représentent respectivement l'évolution des températures de sortie $[y](t)$ générées par des pavés de paramètres de taille 0.5 et 1, soit des incertitudes de 100% et de 300% sur les vraies valeurs de paramètres.

L'évolution des modes d'encadrement dans l'algorithme **Hybrid_bound_Solution** est représentée sur la figure V.27. Les trajectoires bleues continues et rouges pointillées montrent respectivement les différents modes d'encadrement utilisés pour générer les trajectoires d'encadrement inférieure et supérieure sur la période de simulation. Ces modes d'encadrement correspondent aux différentes combinaisons possibles entre les signes des dérivées partielles $g_{i,1}(.) = \frac{\partial f_i(.)}{\partial p_1},\ i = 3, \ldots, 10$ et $g_{i,2}(.) = \frac{\partial f_i(.)}{\partial p_2},\ i \in \{3, 10\}$. Donc, au total nous avons $2^{10} = 1024 + 1$ modes possibles mais qui ne seront pas tous nécessairement atteignables. Par exemple, pour le cas représenté par la figure V.26 (b) le nombre de modes d'encadrement utilisé est 20+1.

V.4.3.2 Inversion ensembliste

Ainsi, l'utilisation de l'algorithme d'intégration numérique garantie des EDO incertaines **Hybrid_Bound_Solution** dans la fonction **Validated_Integration** de l'algorithme **Identification_Ensembliste**, permet de caractériser de manière garantie l'ensemble des paramètres p_1 et p_2 compatibles avec le modèle (V.7), les données simulées et les bornes d'erreurs. La figure V.28 montre une approximation extérieure $\overline{\mathbb{S}}$ de cet ensemble solution obtenue pour un seuil $\epsilon = 0.002$ en un temps de calcul d'environ 24 heures

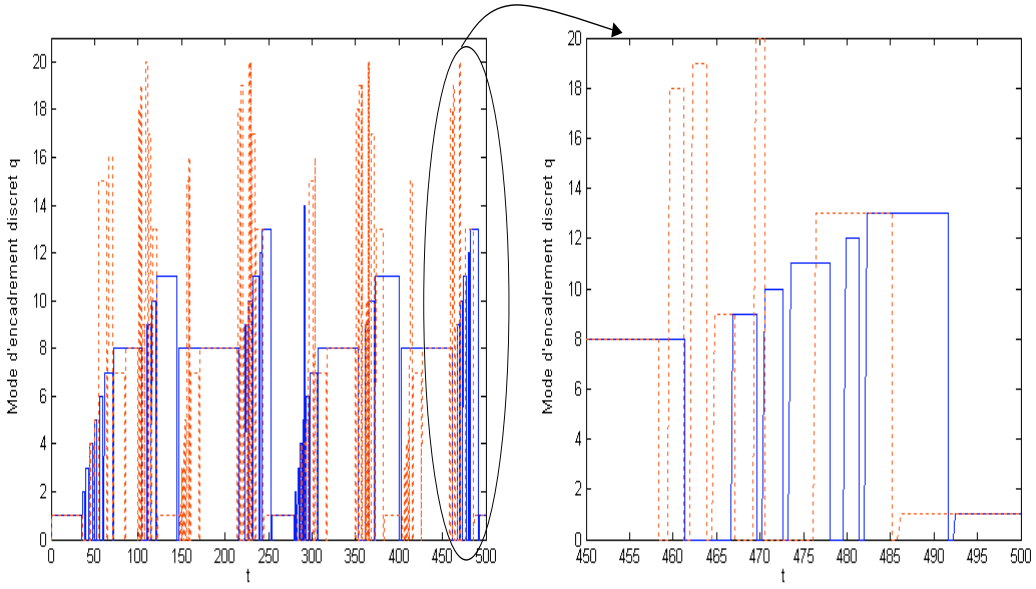


Fig. V.27 *: Evolution du mode d'encadrement q utilisé par la méthode TCID. Le pavé de paramètres est* $[\mathbf{p}]^T = [0.98, 1.98] \times [0.5, 1.5]$.

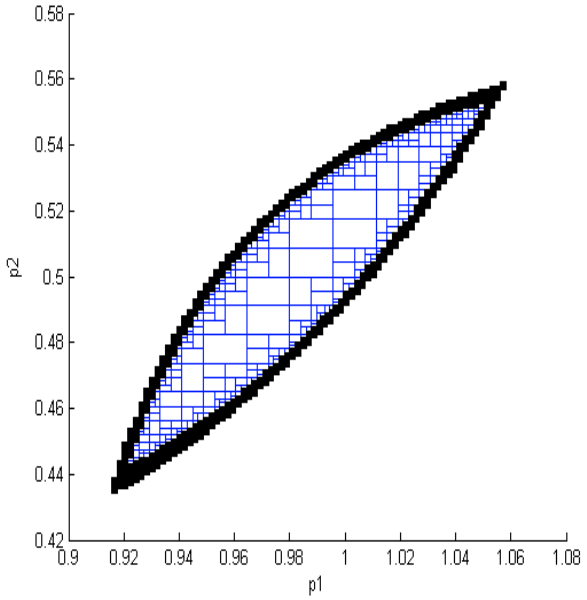


Fig. V.28 *: Approximation intérieure (pavés en bleu) de l'ensemble solution. Couche d'incertitude (pavé en noir), TC=24h.*

sur un PIV 2.4 GHZ. Comme dans la première expérience, la projection de $\overline{\mathbb{S}}$ sur les axes des paramètres donne un majorant garanti de l'incertitude associée à chaque paramètre identifié

$$\begin{cases} p_1 \in [0.918,\ 1.06] mKW^{-1} \\ p_2 \in [0.437,\ 0.559] s^{-1}. \end{cases}$$

V.5 Conclusion

Dans ce chapitre, nous avons abordé le problème de l'identification ensembliste des systèmes dynamiques modélisés par des représentations d'état continues dans un contexte à erreurs bornées et sous l'hypothèse que les mesures sont disponibles à des instants discrets. Notre contribution principale dans ce domaine consiste à introduire la méthode hybride d'intégration numérique garantie présentée dans le chapitre IV. L'intérêt de cette méthode est de pouvoir calculer à chaque instant de mesures un encadrement garanti et précis de toutes les sorties possibles du systèmes à identifier et ce en un temps de calcul raisonnable. Par ailleurs, nous avons appliqué cette approche pour résoudre le problème d'identification simultanée des propriétés thermo-physiques des matériaux. Pour cela, avant de lancer l'algorithme **Identification_Ensembliste** pour caractériser le domaine admissible de ces propriétés thermo-physiques, nous avons comparé les performances (précision des encadrements et temps de calcul) de trois méthodes d'intégration numérique garantie et nous avons montré l'avantage de notre méthode hybride d'encadrement.

Chapitre VI

Observateurs intervalles pour les systèmes dynamiques continus incertains

VI.1 Introduction

L'estimation du vecteur d'état des systèmes dynamiques continus est reconnu comme étant un problème de grande importance en pratique. En effet, pour appliquer les méthodes avancées de commande et de diagnostic de systèmes dynamiques, on a souvent besoin de connaître en ligne leur état interne. Généralement, la mesure directe de cet état par le biais des capteurs industriels n'est pas possible pour différentes raisons physiques, pratiques, économiques ... Cependant, il est possible de réaliser cette tâche par des capteurs logiciels appelés observateurs qui fournissent en ligne, sous certaines conditions d'observabilité [43, 44], une estimation de l'état réel du système considéré en s'appuyant sur son modèle et sur la mesure de son vecteur de sorties.

Dans le cadre des systèmes dynamiques *linéaires*, il existe deux types standards d'observateurs qui sont couramment utilisés et qui fournissent de bon résultats : L'observateur de Luenberger [61] pour le contexte déterministe et le filtre de Kalman [52] pour le contexte aléatoire gaussien. En revanche, dans le cas des systèmes continus *non linéaires* plusieurs approches de construction d'observateurs ont été développées. Cependant, chaque approche s'applique à une classe particulière de systèmes non linéaires comme par exemple :

l'observateur de Luenberger étendu [91], le filtre de Kalman étendu [65], les observateurs à grand gain [26], les observateurs à modes glissants [23, 87], les observateurs par linéarisation par injection de sorties et transformation de coordonnées d'état [29] ... En pratique, toutes ces approches non linéaires sont plus au moins robustes en présence de perturbations sur le vecteur d'état et de bruit de mesures. Par contre, elles sont insatisfaisantes et elles fournissent des estimés d'état imprécis et non robustes en présence d'incertitudes de type paramétrique, c'est-à-dire lorsque les modèles des systèmes dynamiques possèdent des paramètres de valeurs inconnues ou de variations inconnues en fonction du temps.

En pratique, la présence d'incertitudes dans les modèles est très fréquente. Par conséquent, pour remédier aux insuffisances des approches classiques de construction d'observateurs, deux nouvelles approches ont été développées récemment dans le contexte déterministe ensembliste. Il s'agit de pouvoir caractériser l'ensemble de toutes les trajectoires d'état compatibles à la fois avec le modèle, les bornes d'incertitudes sur l'état initial et sur les paramètres et les erreurs additives sur les sorties.

La première approche est adaptée au cas des systèmes continus pour lesquels les mesures sont disponibles à des instants discrets. Cette approche est du type prédiction-correction : (i) la phase de prédiction consiste à calculer une approximation extérieure garantie de l'espace d'état atteignable généré par le système incertain ; (ii) la phase de correction consiste à réduire cette approximation extérieure aux instants de mesure par le biais de l'étude de la consistance entre le domaine admissible de mesures expérimentales et celui obtenu par simulation. Plusieurs formes géométriques ont été utilisées pour mettre en œuvre cette approche comme par exemple : des parallèlotopes [15], des ellipsoïdes [14, 13] et des zonotopes [17]. Cependant, l'application de ces formes géométriques n'est adaptée qu'au cas des systèmes dynamiques linéaires. En revanche, la forme vecteurs intervalles s'applique aussi pour le cas des systèmes non linéaires [49], [76], [77], [53]. Dans la suite de ce chapitre, nous montrons comment l'utilisation des méthodes d'intégration numérique garantie, en particulier celles présentées dans les chapitres III et IV, contribuent à la résolution de ce problème de construction d'observateurs intervalles pour une large classe de systèmes dynamiques non linéaires.

La seconde approche est développée pour le cas des systèmes dynamiques dont les mesures sont des signaux continus. Elle a été proposée pour la première fois dans [30,

37, 31]. Son principe consiste en l'écriture d'un observateur minimal et d'un observateur maximal qui fournissent de manière garantie un encadrement de toutes les trajectoires d'état cohérentes avec les domaines d'incertitudes et les domaines admissibles pour les mesures. Nous allons montrer dans la suite que, sous certaines hypothèses, il existe une matrice gain d'observation qui assure la convergence de la taille de l'encadrement garanti des trajectoires d'état.

VI.2 Première approche : Prédiction-correction

VI.2.1 Contexte

Considérons le système dynamique continu incertain (VI.1) dont les incertitudes sont naturellement représentées par des intervalles bornés avec des bornes connues *a priori*,

$$\left\{\begin{array}{c} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{p}, \mathbf{u}(t)) \\ \mathbf{y}(t) = \mathbf{g}(\mathbf{x}, \mathbf{p}, \mathbf{u}(t)) \\ \mathbf{x}(t_0) \in [\mathbf{x}_0] \subset \mathbb{D} \\ \mathbf{p} \in [\mathbf{p}] \end{array}\right\} \tag{VI.1}$$

où $\mathbf{f} \in C^{k-1}(\mathbb{D} \times \mathbb{U})$, $\mathbb{D} \times \mathbb{U} \subseteq \mathbb{R}^{n+n_u}$ est un domaine ouvert ; n, n_u et m représentent respectivement les dimensions du vecteur d'état $\mathbf{x}$, du vecteur d'entrée $\mathbf{u}$ et du vecteur de sortie $\mathbf{y}$. Les fonctions $\mathbf{f} : \mathbb{D} \times \mathbb{U} \rightarrow \mathbb{R}^n$ et $\mathbf{g} : \mathbb{D} \times \mathbb{U} \rightarrow \mathbb{R}^m$ peuvent être non linéaires. L'état initial $\mathbf{x}_0$ est supposé appartenir à un pavé $[\mathbf{x}_0]$ de bornes connues *a priori*.

Par ailleurs, nous supposons que les signaux de mesures, $\mathbf{y}_m(t_j)$, sont récoltés à des instants de mesure $t_j \in \{t_1, t_2, \ldots, t_{nT}\}$ non nécessairement équidistants. Les erreurs de modélisation et celles dues aux bruits et perturbations sont considérées bornées de bornes connues et additives au vecteur de mesures. Ainsi le domaine admissible des sorties du système (VI.1) à un instant donné t_j est défini par :

$$\mathbb{Y}(t_j) = \mathbf{y}(t_j) + \mathbb{E}(t_j) \tag{VI.2}$$

où $\mathbb{E}(t_j)$ représente le domaine admissible borné de toutes les erreurs à l'instant t_j.

L'objectif de cette approche d'observation, en présence de mesures discrètes, consiste à calculer une approximation extérieure garantie de l'ensemble des trajectoires d'état

définies comme suit :

$$\forall t \in [t_0, t_{nT}],\ \mathbb{X}(t) = \left\{ \begin{array}{c} \mathbf{x}(t) \mid (\forall \mathbf{x}_0 \in [\mathbf{x}_0], \forall \mathbf{p} \in [\mathbf{p}], \forall \tau \in [t_0,\ t_{nT}]\ \dot{\mathbf{x}}(\tau) \in [\mathbf{f}](\mathbf{x}, [\mathbf{p}], \mathbf{u}(\tau))) \\ \wedge \\ (\forall t_j \in \{t_1, t_2, \ldots, t_{nT}\},\ \mathbf{x}(t_j) \in [\mathbf{g}]^{-1}(\mathbb{Y}(t_j), [\mathbf{p}], \mathbf{u}(t_j))) \end{array} \right\}. \tag{VI.3}$$

Pour ce faire, nous allons utiliser les méthodes d'intégration numérique garantie des EDO pour la phase de prédiction et les techniques d'inversion ensembliste basées sur l'arithmétique d'intervalles pour la phase de correction [49], [76], [77], [53].

VI.2.2 Principe

La figure VI.1 montre le principe fondamental des deux phases prédiction et correction de cette approche ensembliste d'estimation dynamique du vecteur d'état entre deux instants successifs de mesures t_j et t_{j+1}. En effet, en utilisant les méthodes d'intégration numérique garantie, la phase de *prédiction* calcule un encadrement garanti $[\mathbf{x}_{j+1}]^p$ de toutes les solutions de (VI.1) à l'instant de mesure t_{j+1} ayant comme conditions initiales $(t_j,\ [\mathbf{x}_j])$

$$\mathbf{x}(t_{j+1}; t_j, [\mathbf{x}_j]) \subseteq [\mathbf{x}_{j+1}]^p. \tag{VI.4}$$

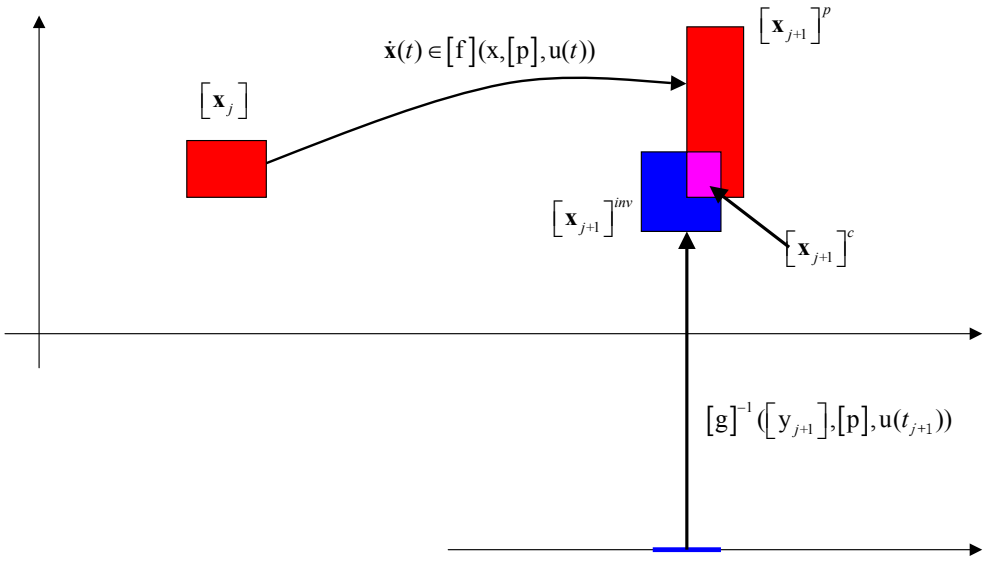


Fig. VI.1 *: Phases prédiction et correction de la première approche d'estimation ensembliste*

La phase de *correction* utilise les techniques d'inversion ensembliste basées sur l'analyse par intervalles [51, 50] pour caractériser à chaque instant t_{j+1} l'ensemble $[\mathbf{x}_{j+1}]^{inv}$ dont l'image par le modèle de sorties $\mathbf{g}$ est incluse dans le domaine admissible des mesures $\mathbb{Y}_{j+1}$, c'est-à-dire

$$[\mathbf{x}_{j+1}]^{inv} = [\mathbf{g}]^{-1}([\mathbf{y}_{j+1}], [\mathbf{p}]) \tag{VI.5}$$

où $[\mathbf{y}_{j+1}] = \mathbb{Y}_{j+1}$. Ensuite, elle corrige le pavé d'état prédit en le comparant avec le pavé $[\mathbf{x}_{j+1}]^{inv}$ puis en éliminant les vecteurs d'état inconsistants

$$[\mathbf{x}_{j+1}]^{c} = [\mathbf{x}_{j+1}]^{inv} \cap [\mathbf{x}_{j+1}]^{p}. \tag{VI.6}$$

Ainsi, pour la prochaine mesure la phase de prédiction sera réinitialisée par $[\mathbf{x}_{j+1}] = [\mathbf{x}_{j+1}]^{c}$.

Enfin, l'algorithme ci-dessous montre le déroulement de cette approche d'estimation ensembliste sur un horizon de temps fini.

Algorithme : Estimation_Prédiction_Correction

1. **Entrées** : $([\mathbf{x}_0], [\mathbf{p}], \mathbf{f}, \mathbf{g}, [\mathbf{y}_1], \ldots, [\mathbf{y}_{nT}])$
2. $t_j = t_0$; $[\mathbf{x}_j] = [\mathbf{x}_0]$;
3. **while** $(t_j < t_{nT})$
4. $\quad \{t_{j+1}, [\mathbf{x}_{j+1}]^p\}$=**Validated_Integration**$([\mathbf{x}_j], [\mathbf{p}], t_j)$;
5. $\quad [\mathbf{x}_{j+1}]^{inv} = [\mathbf{g}]^{-1}([\mathbf{y}_{j+1}], [\mathbf{p}])$;
6. $\quad [\mathbf{x}_{j+1}]^{c} = [\mathbf{x}_{j+1}]^{inv} \cap [\mathbf{x}_{j+1}]^{p}$;
7. $\quad [\mathbf{x}_{j+1}] = [\mathbf{x}_{j+1}]^{c}$
8. $\quad j = j + 1$;
9. **end**
10. **Sorties** : $\mathbb{X}(t)$.

Généralement, la capacité de cette approche à caractériser avec précision l'ensemble (VI.3) dépend essentiellement de la qualité des encadrements de vecteur d'état obtenus par les méthodes d'intégration numérique garantie. Dans la suite, nous montrons comment notre méthode hybride d'intégration contribue à la résolution des problèmes d'estimation d'état pour lesquels les méthodes basées sur les modèles de Taylor intervalles échouent [49], [76], [77].

VI.2.3 Application

VI.2.3.1 Modèle

Nous considérons comme application le domaine de la biotechnologie. Par définition, un bioréacteur est un procédé de transformation de matière dont l'élément actif est vivant, constitué par des bactéries ou des champignons [31, 80].

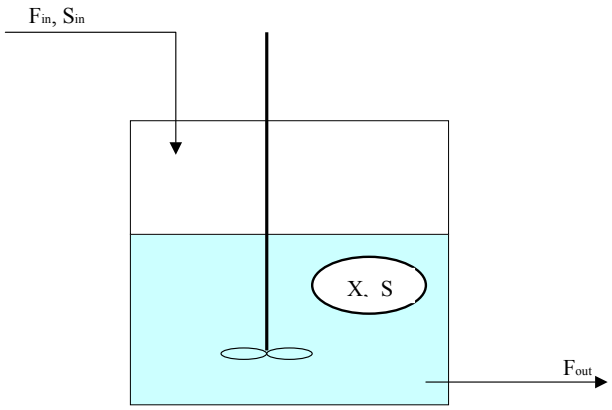


Fig. VI.2 *: Schéma de principe d'un bioréacteur. F_{in} et F_{out} représentent respectivement le débit de la solution d'entrée et le débit de la solution de sortie.*

En pratique, la modélisation de ce genre de procédés est difficile. Ceci est dû au caractère vivant des bactéries représenté par des fonctions complexes mettant en jeu des paramètres mal connus. Dès lors, l'approche intervalle est intéressante pour représenter ces incertitudes.

Pour cette application, nous considérons un modèle simple de bioréacteur décrivant une seule population de bactérie. Ainsi, pour représenter l'état de ce bioréacteur, deux variables sont nécessaires : la concentration en bactéries, appelée biomasse et notée X, et la concentration en substrat, notée S. Le modèle ci-dessous représente l'évolution dynamique de ces deux concentrations

$$\begin{cases} \dot{X}(t) = \mu(S)X - \alpha DX \\ \dot{S}(t) = -k_1\mu(S)X + D(S_{in} - S) \end{cases} \tag{VI.7}$$

où l'état initial est inconnu $(X_0, S_0) \in ([X_0], [S_0])$. La fonction $\mu(S)$ représente le taux de croissance de la biomasse, modélisé par la loi de Haldane :

$$\mu(S) = \mu_0 \frac{S}{S + k_s + S^2/k_i} \tag{VI.8}$$

avec le paramètre μ_0 incertain mais borné :

$$\underline{\mu}_0 \leq \mu_0 \leq \overline{\mu}_0. \tag{VI.9}$$

Ce bioréacteur est alimenté par une solution contenant du substrat en concentration S_{in} qui n'est pas mesurée de manière parfaite

$$\underline{S}_{in}(t) \leq S_{in}(t) \leq \overline{S}_{in}(t). \tag{VI.10}$$

Nous supposons que la biomasse est accessible à la mesure à des instants discrets t_j. Ces mesures sont affectées par des erreurs additives inconnues mais bornées de bornes connues *a priori*

$$\mathbb{Y}(t_j) = X_m(t_j) + \mathbb{E}(t_j). \tag{VI.11}$$

Dans ce contexte, l'objectif de l'estimation d'état consiste à reconstruire à chaque instant tous les vecteurs d'état compatibles avec le modèle (VI.7) et ses incertitudes (VI.9), (VI.10) et les domaines de mesures donnés par (VI.11).

VI.2.3.2 Simulation (tests et résultats)

Pour réaliser la phase de prédiction de cette approche ensembliste d'estimation, c'est-à-dire calculer toutes les trajectoires d'état possibles de (VI.7) générées par son domaine d'état initial et celui de ces paramètres incertains, nous allons utiliser les deux types de méthodes d'intégration numérique garantie des EDO incertaines : (i) les méthodes basées sur le développement de Taylor intervalle ou le polynôme de Hermite-Obreschkoff intervalle, notées dans la suite par MI (méthodes intervalles) ; et (ii) les méthodes fondées sur la théorie de comparaison des inégalités différentielles TCID.

Les données considérées pour ce problème d'estimation d'état sont : $\alpha = 0.5$, $k = 42.14$, $k_s = 9.28$ mmol/l, $k_i = 256$ mmol/l, $\mu_0 \in [0.64, 0.84]$, $X_0 \in [0, 10]$, $S_0 \in [0, 100]$, $S_{in}(t) \in [62 + 15\cos(1/5t), 68 + 15\cos(1/5t)]$ et

$$D(t) = \begin{cases} 2 \text{ si } 0 \leq t \leq 5 \\ 0.5 \text{ si } 5 \leq t \leq 10 \\ 1.14 \text{ si } 10 \leq t \leq 20. \end{cases}$$

Le signal discret de mesures $X_m(t_j)$ est construit par un échantillonnage du signal continu $X(t)$ obtenus par la simulation du système (VI.7) avec les valeurs suivantes des paramètres

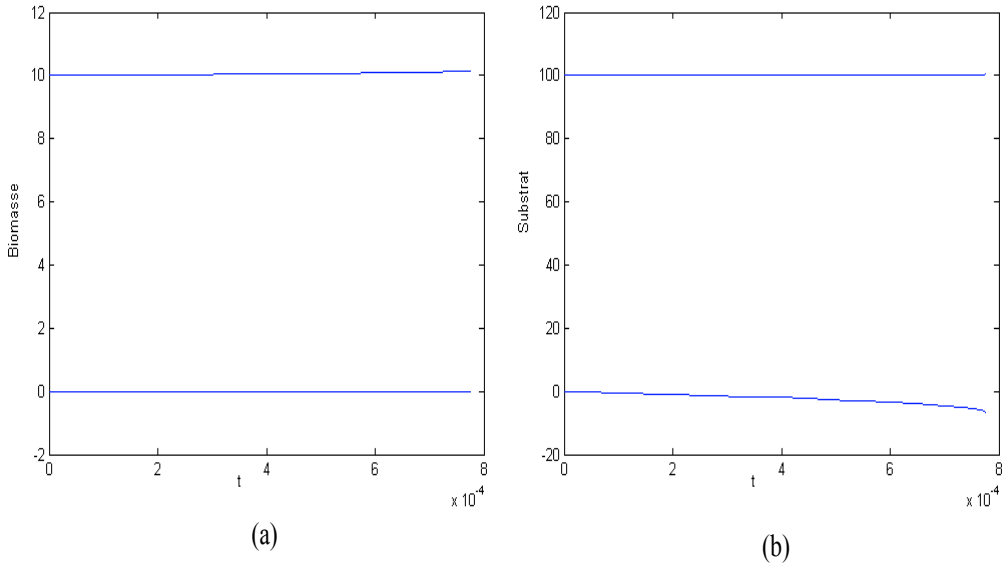


Fig. VI.3 *: Arrêt de l'algorithme d'estimation d'état après quelques pas d'intégration (essai 1).*

incertains et d'état initial : $\mu_0 = 0.74$, $X_0 = 5$, $S_0 = 40$ et $S_{in}(t) = 65 + 15\cos(1/5t)$. Le pas de cette discrétisation est noté h_m, et la période de simulation est $[T] = [0, 20]\,h$.

VI.2.3.2.1 Estimation d'état par le biais des méthodes MI

Essai 1 : Pour ce premier cas d'étude, nous considérons que les mesures sont disponibles à des intervalles de temps h_m réguliers, $h_m = t_{j+1} - t_j = 1\,h$. Leurs domaines admissibles sont donnés par

$$\mathbb{Y}(t_j) = [0.98X_m(t_j), 1.02X_m(t_j)].$$

Comme prévu, la propagation du pessimisme, dû à l'effet d'enveloppement dans les méthodes d'intégration numérique garantie MI, provoque l'arrêt de l'algorithme **Estimation_Prédiction_Correction** au bout de quelques pas d'intégration. En effet, les courbes sur figures VI.3 (a) et (b) montrent bien que les tailles des encadrements des deux variables d'état divergent avant l'arrivée de la première mesure.

Essai 2 : Dans ce deuxième essai, pour réduire le pessimisme nous supposons que l'état initial du bioréacteur (VI.7) est parfaitement connu $(X_0, S_0)^T = (5, 40)$ et que les mesures sont disponibles à des instants très proches $h_m = t_{j+1} - t_j = 0.1\,h$.

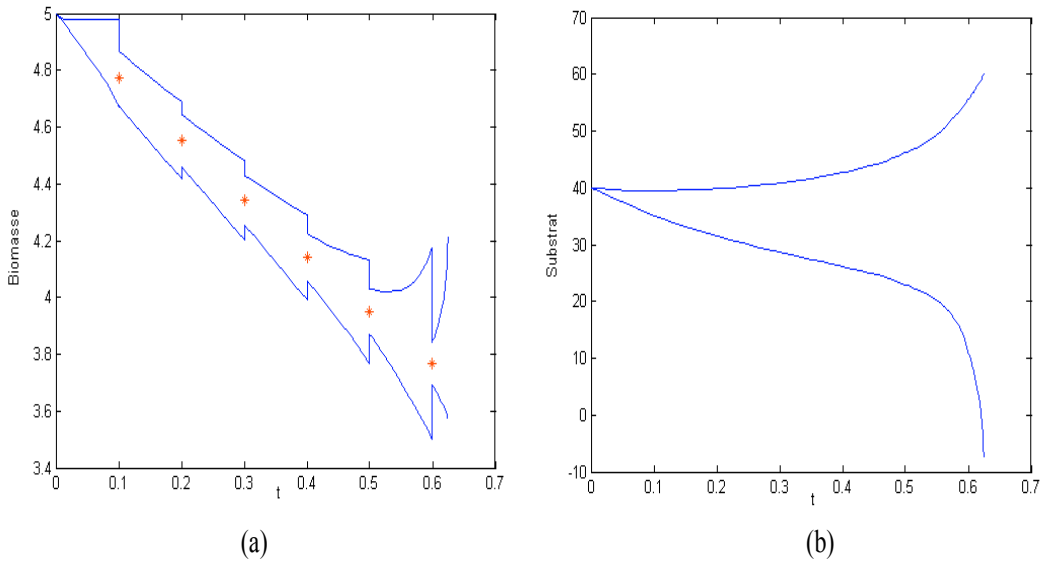


Fig. VI.4 *: Divergence de la taille des encadrements des variables d'état représentées par les courbes bleues continues (essai 2). La courbe rouge pointillée représente le signal discret de mesure.*

Malgré ces considérations, les courbes sur les figures VI.4 (a) et (b) montrent que la propagation du pessimisme persiste et provoque aussi l'arrêt de l'algorithme **Estimation_Prédiction_Correction** après une courte durée de simulation.

Essai 3 : Pour ce troisième essai, nous considérons que les domaines de mesures sont plus précis que ceux utilisés dans les deux premiers essais.

$$\mathbb{Y}(t_j) = [0.999X_m(t_j), 1.001X_m(t_j)].$$

De plus, nous gardons les mêmes suppositions faites dans le deuxième essai.

Même pour ce cas d'étude, l'effet d'enveloppement fait que l'algorithme **Estimation_Prédiction_Correction** ne peut pas estimer l'état du bioréacteur (VI.7), comme l'illustrent les figures VI.5 (a) et (b).

Essai 4 : Considérons un cas où l'incertitude sur les paramètres est petite $[\mu_0] = [0.72, 0.76]$ et $[S_{in}(t)] = [64.5 + 15\cos(1/5t), 65.5 + 15\cos(1/5t)]$, l'état initial est parfaitement connu $(X_0, S_0)^T = (5, 40)$, les mesures sont très proches $h_m = t_{j+1} - t_j = 0.1\,h$ et

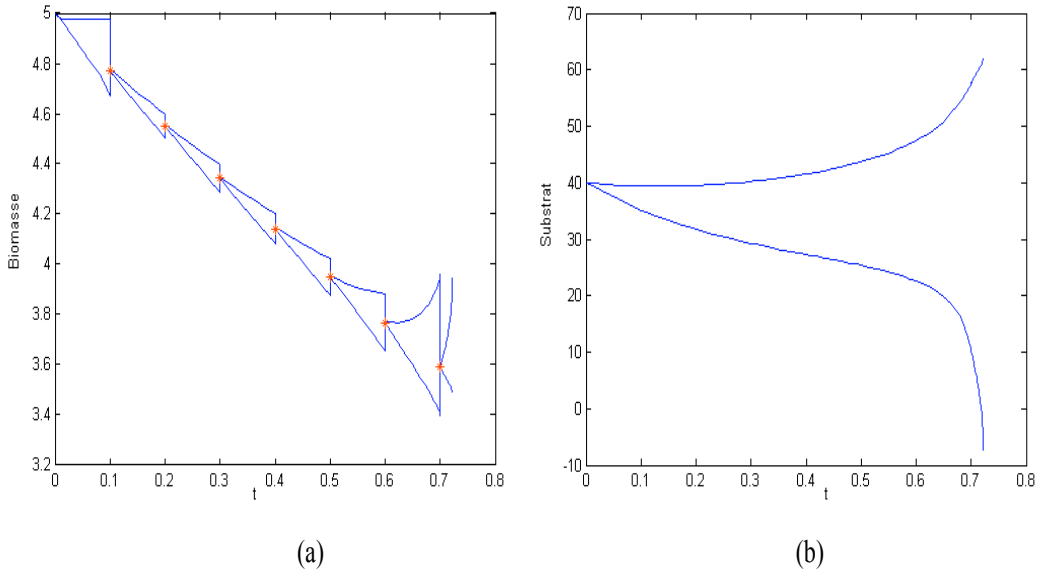


Fig. VI.5 *: Divergence de la taille des encadrements des variables d'état, même dans le cas où les mesures sont précises (essai 3).*

très précises,

$$\mathbb{Y}(t_j) = [0.999X_m(t_j), 1.001X_m(t_j)].$$

Les résultats de cette simulation sont présentés sur la figure VI.6.

Sur cette figure, on voit bien que l'accumulation du pessimisme sur une grande période de simulation provoque l'arrêt de l'algorithme d'estimation d'état avant qu'il atteint l'instant final.

Le tableau VI.1 récapitule les caractéristiques de simulation de chaque essai d'observation.

	$[X_0] \times [S_0]$	$[\mu_0]$	$[S_{in}](t)$	$\mathbb{Y}(t_j)$	h_m
Essai 1	$[0, 10] \times [0, 100]$	$[0.64, 0.84]$	$[62, 68] + 15\cos(1/5t)$	$[0.98, 1.02]X_m(t_j)$	$1\,h$
Essai 2	$[5] \times [40]$	$[0.64, 0.84]$	$[62, 68] + 15\cos(1/5t)$	$[0.98, 1.02]X_m(t_j)$	$0.1\,h$
Essai 3	$[5] \times [40]$	$[0.64, 0.84]$	$[62, 68] + 15\cos(1/5t)$	$[0.999, 1.001]X_m(t_j)$	$0.1\,h$
Essai 4	$[5] \times [40]$	$[0.72, 0.76]$	$[64.5, 65.5] + 15\cos(1/5t)$	$[0.999, 1.001]X_m(t_j)$	$0.1\,h$

Tab. VI.1 *: Caractéristiques de simulation de chaque essai*

Ainsi, nous constatons que les méthodes d'intégration numérique garantie MI ne sont

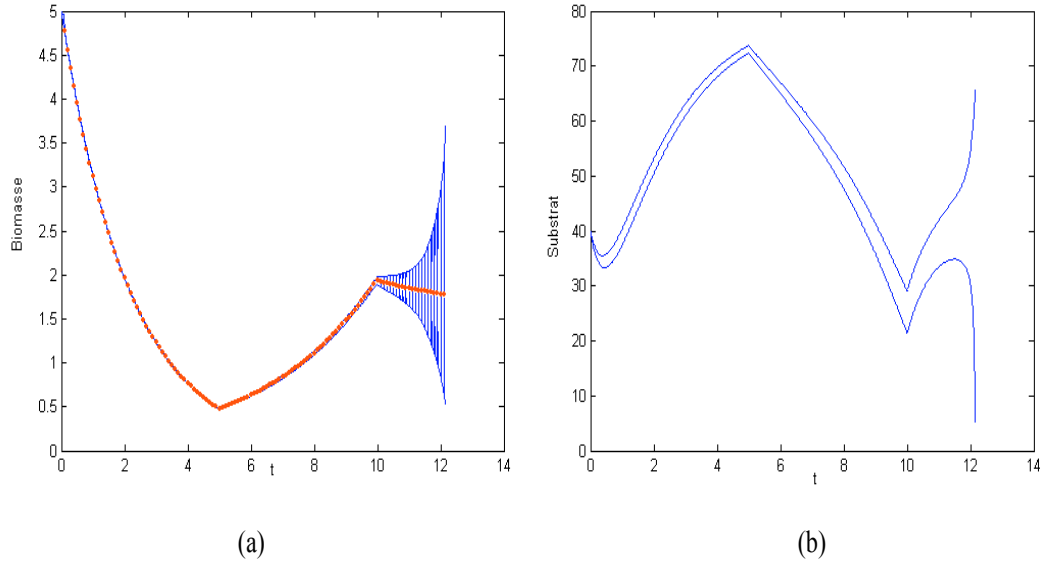


FIG. VI.6 *: Divergence de la taille des encadrements des variables d'état sous l'effet de l'accumulation du pessimisme (essai 4).*

pas capables de résoudre ce problème d'estimation d'état. Par ailleurs, nous allons montrer dans la section suivante comment les méthodes TCID, et en particulier notre méthode hybride, peuvent le résoudre avec succès.

VI.2.3.2.2 Estimation d'état par le biais des méthodes TCID

Etant donné que le système (VI.7) n'est pas coopératif, le théorème de Müller (**Théorème 2**) peut être utilisé pour obtenir ses systèmes englobants. Cependant, la **Règle 2** de construction de ces systèmes n'est pas applicable à cause du changement de signe de la dérivée partielle $\frac{\partial \dot{X}}{\partial S}$ sur l'intervalle de simulation $[T]$. En effet, le système (VI.7) admet deux modes d'encadrement possibles selon le signe de la dérivée de μ par rapport à S,

$$sign(\frac{d\mu(S)}{dS}) = \begin{cases} > 0 \text{ si } \forall S \in [S],\ S < \sqrt{k_s k_i} \\ \leq 0 \text{ si } \forall S \in [S],\ S \geq \sqrt{k_s k_i} \\ \text{ambigu si } \sqrt{k_s k_i} \in [S] \end{cases} \quad \text{(VI.12)}$$

Le premier mode, $q = 1$, correspond à la période de temps où cette dérivée est négative

$$(q=1)\begin{cases}\dot{\overline{X}}(t) = \overline{\mu}(\overline{S})\overline{X} - \alpha D\overline{X}\\ \dot{\overline{S}}(t) = -k_1\underline{\mu}(\overline{S})\underline{X} + D(\overline{S}_{in} - \overline{S})\\ \dot{\underline{X}}(t) = \underline{\mu}(\underline{S})\underline{X} - \alpha D\underline{X}\\ \dot{\underline{S}}(t) = -k_1\overline{\mu}(\underline{S})\overline{X} + D(\underline{S}_{in} - \underline{S})\end{cases} \tag{VI.13}$$

et le second mode, $q = 2$ correspond au cas où cette dérivée est positive

$$(q=2)\begin{cases}\dot{\overline{X}}(t) = \overline{\mu}(\underline{S})\overline{X} - \alpha D\overline{X}\\ \dot{\overline{S}}(t) = -k_1\underline{\mu}(\overline{S})\underline{X} + D(\overline{S}_{in} - \overline{S})\\ \dot{\underline{X}}(t) = \underline{\mu}(\overline{S})\underline{X} - \alpha D\underline{X}\\ \dot{\underline{S}}(t) = -k_1\overline{\mu}(\underline{S})\overline{X} + D(\underline{S}_{in} - \underline{S}).\end{cases} \tag{VI.14}$$

Ces deux modes d'encadrement sont construits par l'utilisation de la **Règle 2**. Pour les intervalles de temps où le signe de cette dérivée est ambigu, nous pouvons utiliser deux approches : la première utilise la modélisation intervalle des grandeurs incertaines (paramètres, état initial) dans (VI.7) ; la seconde utilise un raisonnement trivial pour construire les systèmes englobants. Par exemple, nous pouvons utiliser la double inégalité ci-dessous pour encadrer $\mu(S)$ quel que soit son signe,

$$\underline{\mu}_0\frac{\underline{S}}{\underline{S} + k_s + \underline{S}\overline{S}/k_i} \leq \mu(S) \leq \overline{\mu}_0\frac{\overline{S}}{\overline{S} + k_s + \underline{S}\overline{S}/k_i}. \tag{VI.15}$$

Cette double inégalité est tirée de [67].

Preuve :

Sachant que $\forall t > 0\, S(t) \geq 0$, il est facile de démontrer que :

$$\forall S \in [\underline{S}, \overline{S}]\begin{cases}\frac{S}{S+k_s+S^2/k_i} - \frac{\underline{S}}{\underline{S}+k_s+\underline{S}\overline{S}/k_i} \geq 0\\ \frac{S}{S+k_s+S^2/k_i} - \frac{\overline{S}}{\overline{S}+k_s+\underline{S}\overline{S}/k_i} \leq 0.\end{cases}$$

De plus, dans ce contexte l'encadrement donné par la double inégalités (VI.15) est moins pessimiste que celui obtenu par la méthode intervalle comme le montre la relation ci-dessous :

$$\forall S \in [\underline{S}, \overline{S}]\begin{cases}\frac{\underline{S}}{\underline{S}+k_s+\underline{S}\overline{S}/k_i} \geq \frac{\underline{S}}{\overline{S}+k_s+\overline{S}^2/k_i} \in \frac{[S]}{[S]+k_s+[S]^2/k_i}\\ \frac{\overline{S}}{\overline{S}+k_s+\underline{S}\overline{S}/k_i} \leq \frac{\overline{S}}{\underline{S}+k_s+\underline{S}^2/k_i} \in \frac{[S]}{[S]+k_s+[S]^2/k_i}.\end{cases}$$

Ainsi, nous proposons le système d'équations différentielles (VI.16) pour simuler le

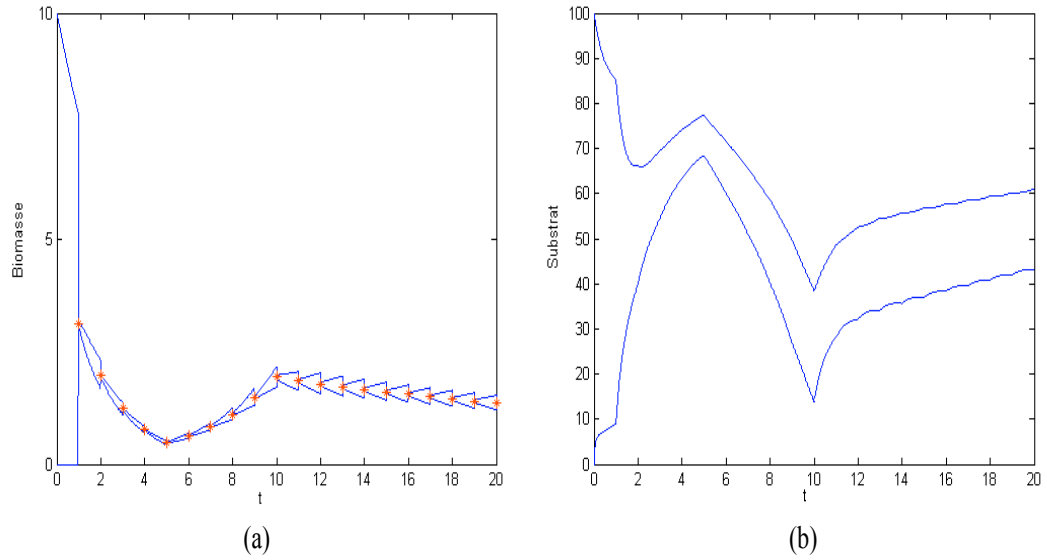


FIG. VI.7 *: Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 1 : (a) Biomasse ; (b) Substrat.*

système (VI.7) sur les périodes où le signe de $\frac{\partial \dot{X}}{\partial S}$ est ambigu.

$$(q=3)\begin{cases} \dot{\overline{X}}(t) = \overline{\mu}_0 \frac{\overline{S}}{\overline{S}+k_s+\underline{S}\overline{S}/k_i}\overline{X} - \alpha D\overline{X} \\ \dot{\overline{S}}(t) = -k_1\underline{\mu}(\overline{S})\underline{X} + D(\overline{S}_{in}-\overline{S}) \\ \dot{\underline{X}}(t) = \underline{\mu}_0 \frac{\underline{S}}{\underline{S}+k_s+\underline{S}\overline{S}/k_i}\underline{X} - \alpha D\underline{X} \\ \dot{\underline{S}}(t) = -k_1\overline{\mu}(\underline{S})\overline{X} + D(\underline{S}_{in}-\underline{S}). \end{cases} \tag{VI.16}$$

Finalement, l'utilisation de l'algorithme **Estimation_Prédiction_Correction**, dont la fonction **Validated_Integration** implémente l'algorithme **Hybrid_Bounding** avec ces trois modes d'encadrement (VI.13),(VI.14),(VI.16) (ici le mode $q = 3$ remplace le mode $q = 0$ de l'algorithme **Hybrid_Bounding**), permet de reconstruire avec garantie tous les vecteurs d'état de (VI.7) consistants avec les données disponibles et leurs bornes d'incertitudes. Pour montrer les résultats d'estimation fournis par cette approche, nous réalisons plusieurs essais que nous détaillons ci-après.

Essai 1 : Nous considérons les mêmes données que celles présentées dans le premier essai de la section VI.2.3.2.1. Les résultats de la simulation sont tracés sur la figure VI.7 où la courbe rouge pointillée représente le signal discret de mesures. Les courbes en lignes bleues continues montrent un encadrement garanti de toutes les trajectoires d'état

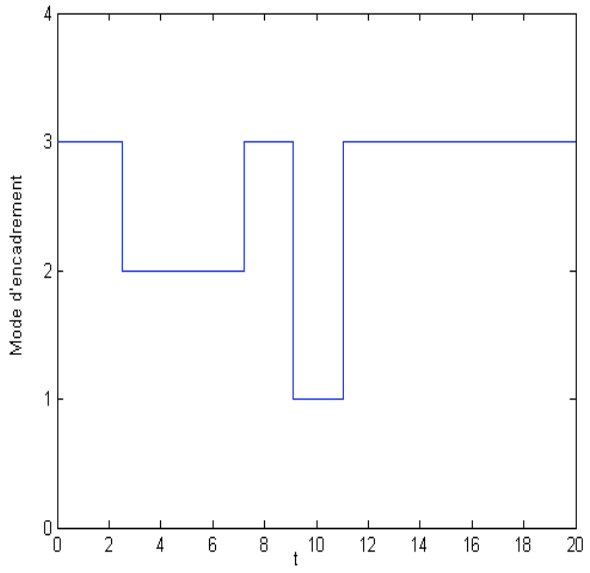


Fig. VI.8 *: Evolution du mode d'encadrement dans le contexte de l'essai 1.*

admissibles. Il est clair que la taille de cet encadrement ne diverge pas, contrairement au premier essai de la section VI.2.3.2.1 où nous avons utilisé les méthodes d'intégration garantie MI pour réaliser la phase de prédiction.
La figure VI.8 montre les transitions entre les différents modes d'encadrement sur la période de la simulation.

Essai 2 : Pour ce deuxième cas d'étude, nous réduisons le nombre de mesures et nous observons l'effet sur la taille de l'encadrement dynamique des vecteurs d'état admissibles. Les courbes bleues continues représentées sur la figure VI.9 sont obtenues pour un pas de mesure $h_m = 2\,h$, c'est-à-dire que nous avons divisé le nombre de mesures par deux par rapport au premier essai.

Comme le montre la figure VI.9, nous constatons que la taille de l'encadrement obtenu dans ce deuxième essai est un peu plus grande que celle du premier essai. En effet, les courbes noires discontinues sur cette figure représentent les résultats obtenus dans l' essai 1. Néanmoins, nous pouvons tout de même dire que cet encadrement des vecteurs d'état admissibles est raisonnable vu la taille du pavés d'état initial et celle du pavé de paramètres incertains.

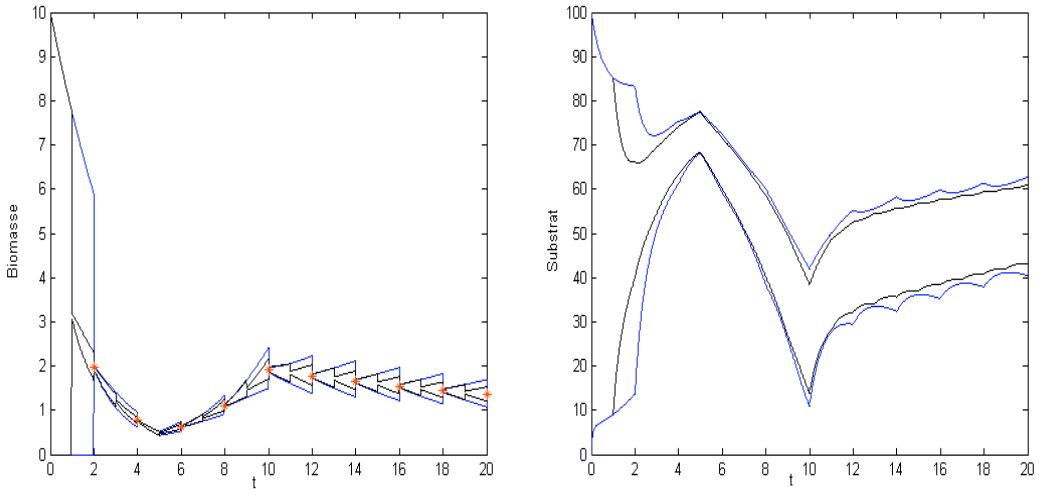


Fig. VI.9 *: Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 2 : (a) Biomasse ; (b) Substrat.*

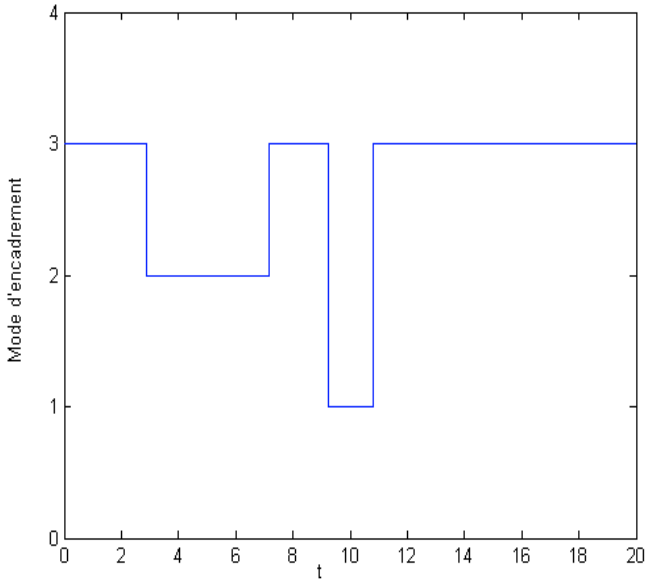


Fig. VI.10 *: Evolution du mode d'encadrement dans le contexte de l'essai 2.*

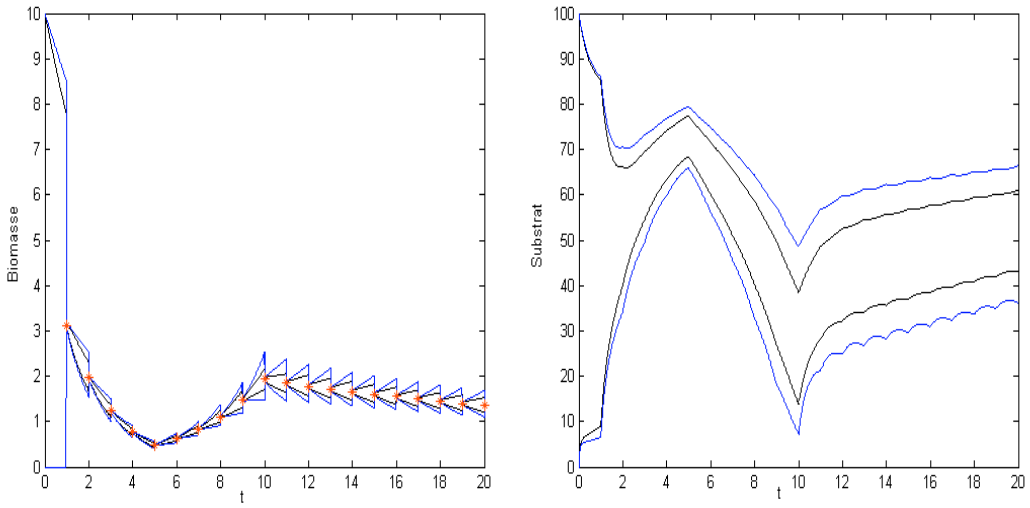


Fig. VI.11 *: Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 3 : (a) Biomasse ; (b) Substrat.*

Essai 3 : Ce troisième essai consiste à augmenter l'incertitude sur les paramètres par rapport au premier essai, c'est-à-dire de prendre un pavé de paramètres plus grand, et de constater son effet sur les résultats de l'estimation d'état.

Les courbes en bleu continues sur la figure VI.11 représentent un encadrement garanti de toutes les trajectoires d'état obtenues pour $[\mu_0] = [0.54, 0.94]$ et $[S_{in}(t)] = [61 + 15\cos(1/5t), 69 + 15\cos(1/5t)]$. Evidemment, la taille de cet encadrement est nettement plus grande que celle du premier essai. Cela est justifiable vu la taille du pavé de paramètres considérée dans ce troisième essai.

Essai 4 : Maintenant, nous augmentons l'incertitude sur les mesures, c'est-à-dire les domaines admissibles des mesures deviennent :

$$\mathbb{Y}(t_j) = [0.9X_m(t_j), 1.1X_m(t_j)].$$

Comme prévu, étant donné que la deuxième phase de cette méthode d'estimation d'état corrige moins l'encadrement de la variable X aux instants de mesures, la précision de l'encadrement de toutes les trajectoires d'état obtenus dans ce contexte est inférieure à celle de l'essai 1. Cela est bien illustré sur la figure VI.13 où les courbes noires conti-

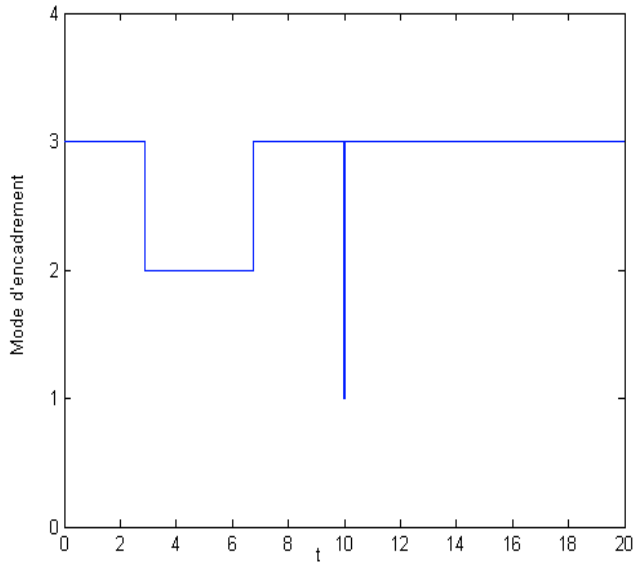


Fig. VI.12 *: Evolution du mode d'encadrement dans le contexte de l'essai 3.*

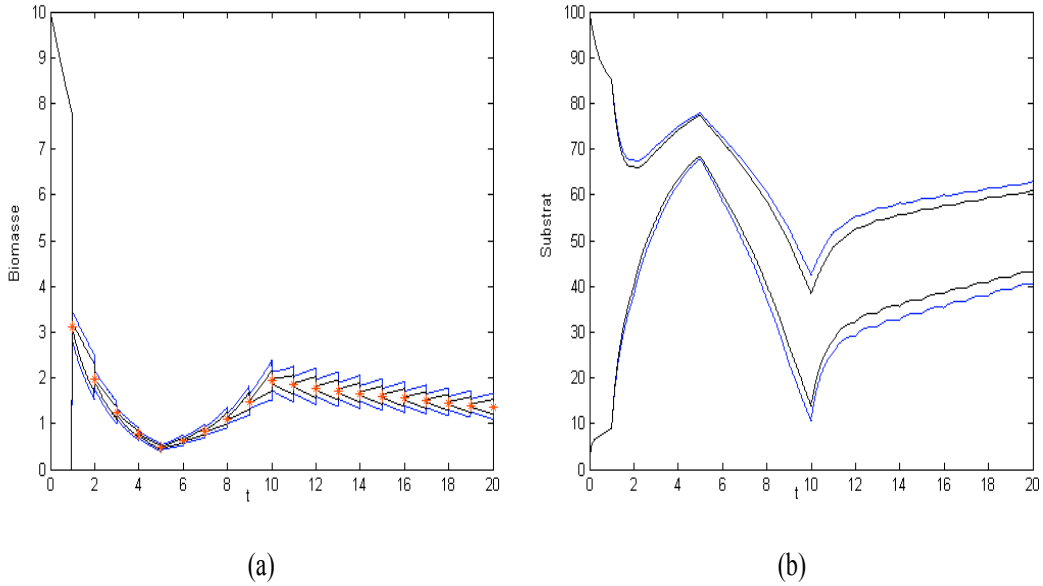


Fig. VI.13 *: Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 4 : (a) Biomasse ; (b) Substrat.*

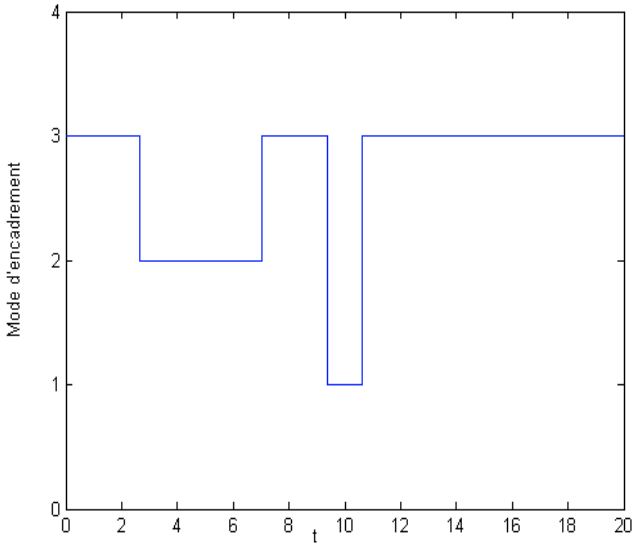


Fig. VI.14 *: Evolution du mode d'encadrement dans le contexte de l'essai 4.*

nues représentent les résultats de l'estimation du premier essai et celles en bleu continu représentent les résultats obtenus dans ce quatrième essai.

Essai 5 : Pour terminer cette série d'essais, nous considérons un cas où l'incertitude sur les paramètres est faible. Par exemple, $[\mu_0] = [0.72, 0.76]$ et $[S_{in}(t)] = [64.5 + 15\cos(1/5t), 65.5 + 15\cos(1/5t)]$, et les autres données étant identiques à celles du premier essai.

Dans ce contexte, la figure VI.15 montre que l'algorithme de l'estimation ensembliste fournit des résultats très satisfaisants.

Le tableau VI.2 résume les caractéristiques de simulation de chaque test d'observation.

Pour cette première approche de construction d'observateurs intervalles, nous avons montré les limites des méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalles pour résoudre le problème d'observation pour des systèmes réels (bioréacteurs). En revanche, nous avons montré l'intérêt et les avantages de construire ce type d'observateur en s'appuyant sur la théorie de comparaison des inégalités différentielles, en particulier notre méthode hybride pour l'encadrement des solutions des EDO.

Nous abordons dans la section suivante, ce problème d'estimation ensembliste d'état

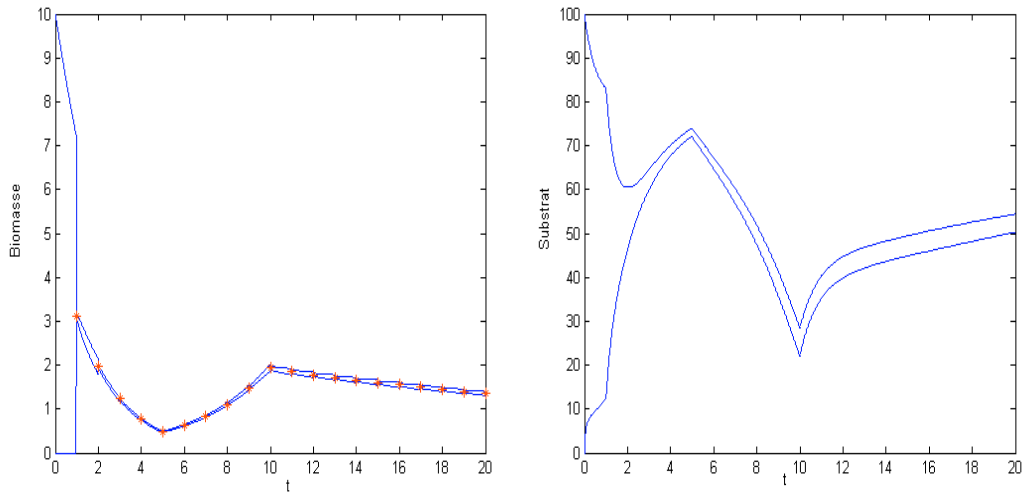


Fig. VI.15 *: Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 5 : (a) Biomasse ; (b) Substrat.*

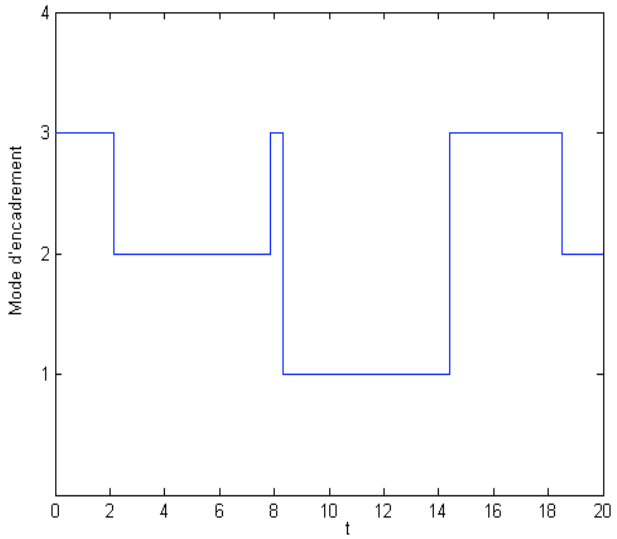


Fig. VI.16 *: Evolution du mode d'encadrement dans le contexte de l'essai 5.*

	$[X_0] \times [S_0]$	$[\mu_0]$	$[S_{in}](t)$	$\mathbb{Y}(t_j)$	h_m
Essai 1	$[0, 10] \times [0, 100]$	$[0.64, 0.84]$	$[62, 68] + 15\cos(1/5t)$	$[0.98, 1.02]X_m(t_j)$	$1\,h$
Essai 2	$[0, 10] \times [0, 100]$	$[0.64, 0.84]$	$[62, 68] + 15\cos(1/5t)$	$[0.98, 1.02]X_m(t_j)$	$2\,h$
Essai 3	$[0, 10] \times [0, 100]$	$[0.54, 0.94]$	$[61, 69] + 15\cos(1/5t)$	$[0.999, 1.001]X_m(t_j)$	$1\,h$
Essai 4	$[0, 10] \times [0, 100]$	$[0.64, 0.84]$	$[62, 68] + 15\cos(1/5t)$	$[0.9, 1.1]X_m(t_j)$	$1\,h$
Essai 5	$[0, 10] \times [0, 100]$	$[0.72, 0.76]$	$[64.5, 65.5] + 15\cos(1/5t)$	$[0.98, 1.02]X_m(t_j)$	$1\,h$

Tab. VI.2 *: Caractéristiques de simulation de chaque test d'observation*

lorsque les mesures sont des signaux continus et nous montrons comment construire des observateurs intervalles en boucle fermée.

VI.3 Deuxième approche : Observateurs intervalles

VI.3.1 Contexte

Dans cette section, nous considérons une classe particulière des systèmes dynamiques continus non linéaires constitués par une partie linéaire observable et une partie non linéaire incertaine bornée par rapport aux paramètres,

$$\left\{\begin{array}{lcl} \dot{\mathbf{x}}(t) & = & \mathbf{A}\mathbf{x}(t) + \Psi(\mathbf{x}, \mathbf{p}, \mathbf{u}(t)) \\ \mathbf{y}(t) & = & \mathbf{C}\mathbf{x}(t) \\ \mathbf{x}(t_0) & \in & [\mathbf{x}_0] \\ \mathbf{p} & \in & [\mathbf{p}] \end{array}\right\} \tag{VI.17}$$

où $\mathbf{A}$ est une matrice de dimension $n \times n$, $\mathbf{C}$ une matrice de dimension $n \times m$, $\Psi \in C^{k-1}(\mathbb{D} \times \mathbb{U} \times [\mathbf{p}])$, $\mathbb{D} \times \mathbb{U} \times [\mathbf{p}] \subseteq \mathbb{R}^{n+n_u+n_p}$ est un ensemble fermé ; n, n_u, m et n_p sont respectivement les dimensions du vecteur d'état $\mathbf{x}$, du vecteur d'entrée $\mathbf{u}$, du vecteur de sortie $\mathbf{y}$ et du vecteur de paramètres incertains $\mathbf{p}$.

Le choix du type de systèmes (VI.17) n'est pas réellement restrictif. En effet, il est possible de transformer un système non linéaire donné sous la forme (VI.17) en utilisant un changement de coordonnées d'état réversible et les techniques de linéarisation par injection de sorties [48].

Les mesures sont considérées comme étant des signaux continus dans le temps incertains mais bornées. Leur domaine admissible est donné par

$$\mathbb{Y}(t) = \mathbf{y}_m(t) + \mathbb{E}(t) \tag{VI.18}$$

où $\mathbb{E}(t)$ représente le domaine admissible des erreurs de sorties qui inclut les erreurs de la modélisation, les erreurs dues aux bruits de mesures et les perturbations agissants sur le système.

Nous supposons aussi que le système (VI.17) satisfait les quatre hypothèses ci-dessous :

Hypothèse 1 : La paire de matrices $(\mathbf{A}, \mathbf{C})$ est observable.

Hypothèse 2 : Il existe une matrice positive $\mathbf{K}$ tel que les éléments non diagonaux de la matrice $\mathbf{A} - \mathbf{KC}$ soient non négatifs.

Hypothèse 3 : Toutes les composantes $\psi_i(.)$ de la fonction vectorielle $\Psi(.)$ sont Lipschitziennes par rapport au vecteur d'état pour tout $\mathbf{p} \in [\mathbf{p}]$.

Hypothèse 4 : Il existe deux fonctions déterministes $\underline{\Psi}(.)$ et $\overline{\Psi}(.)$ établies à partir des bornes de $[\mathbf{p}]$ et un réel positif $M < +\infty$ tel que :

$$\begin{cases} \forall \mathbf{p} \in [\mathbf{p}],\ \forall (\mathbf{u}(t), \mathbf{x}) \in \mathbb{U} \times \mathbb{D}, \\ \underline{\Psi}(\mathbf{x}, \mathbf{p}, \mathbf{u}(t)) \leq \Psi(\mathbf{x}, \mathbf{p}, \mathbf{u}(t)) \leq \overline{\Psi}(\mathbf{x}, \mathbf{p}, \mathbf{u}(t)), \\ \|\overline{\Psi}(\mathbf{x}, \mathbf{p}, \mathbf{u}(t)) - \underline{\Psi}(\mathbf{x}, \mathbf{p}, \mathbf{u}(t))\| \leq M. \end{cases} \tag{VI.19}$$

Dans le contexte déterministe, c'est-à-dire lorsque le système (VI.17) est parfaitement connu et s'il satisfait les conditions d'observabilité des systèmes non linéaires [43, 44], alors le modèle général de son observateur d'état est défini par

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \Psi(\hat{\mathbf{x}}, \mathbf{p}, \mathbf{u}(t)) + \mathbf{K}(\hat{\mathbf{x}})(\mathbf{y}_m^*(t) - \mathbf{y}(t)) \tag{VI.20}$$

où $\mathbf{y}_m^*(t) \in \mathbb{Y}(t)$. Il s'agit donc d'une duplication du modèle de système plus un terme correctif qui consiste à forcer l'erreur d'observation à converger vers zéro pour un t suffisamment grand,

$$\lim_{t \to \infty} \| \hat{\mathbf{x}}(t) - \mathbf{x}(t) \| = 0 \quad \forall \mathbf{x}_0 \neq \hat{\mathbf{x}}_0(t). \tag{VI.21}$$

En revanche, dans le contexte ensembliste à erreurs bornées, l'objectif des observateurs intervalles consiste à : (i) Fournir une enveloppe variable dans le temps qui encadre toutes les trajectoires d'état compatibles avec le domaine admissible de paramètres et avec le domaine admissible des mesures ; (ii) Calculer le gain du terme correctif tel que la taille

de cette enveloppe converge vers une constante qui ne dépend que des tailles des domaines admissibles de mesures et de vecteur de paramètres.

$$\begin{cases} \forall \mathbf{x}(t_0) \in [\underline{\hat{\mathbf{x}}}(t_0), \overline{\hat{\mathbf{x}}}(t_0)], \quad \forall t > t_0 \\ (\underline{\hat{\mathbf{x}}}(t) \leq \mathbf{x}(t) \leq \overline{\hat{\mathbf{x}}}(t)) \ \wedge \ (\lim_{t\to\infty} \|\overline{\hat{\mathbf{x}}}(t) - \underline{\hat{\mathbf{x}}}(t)\| = \varepsilon(w([\mathbf{p}]), w(\mathbb{E}))) \end{cases} \tag{VI.22}$$

où $\varepsilon(.)$ est une fonction de $w([\mathbf{p}])$ et de $w(\mathbb{E})$.

Initialement, ce problème a été résolu dans [31] sous les hypothèses que la matrice $\mathbf{A}$ est coopérative et que la fonction $\Psi(.)$ est bornée et ne dépend pas du vecteur d'état $\mathbf{x}$. Pour satisfaire ces hypothèses, les auteurs de [80] ont utilisé une transformation de coordonnées d'état linéaire adaptée au cas des bioréacteurs [80]. L'idée mise en œuvre est de construire à partir du système incertain VI.17 deux observateurs certains qui estiment les trajectoires d'état minimale et maximale englobant de manière garantie tous les vecteurs d'état cohérents avec le modèle, les données expérimentales disponibles et avec la connaissances *a priori* sur le système. En d'autres termes, cette approche d'observation consiste à fournir un encadrement dynamique garanti du vecteur d'état quelle que soit l'incertitude sur les données disponibles. Pour augmenter la précision de cet encadrement, la notion du *faisceau* d'observateurs intervalles est utilisée dans [9]. Il s'agit de faire évoluer simultanément plusieurs observateurs intervalles de matrices gain d'observation différentes et de sélectionner sur chaque intervalle de temps le meilleur observateur, c'est-à-dire celui qui fournit l'encadrement le plus précis.

Le cas où la fonction $\Psi(.)$ dépend du vecteur d'état $\mathbf{x}$ est considéré dans [67]. Les auteurs ont proposé une méthode d'encadrement de $\Psi(.)$ qui nécessite un couplage entre les dynamiques des deux observateurs minimal et maximal.

Dans cette optique, notre contribution consiste à utiliser le théorème de Müller et la méthode hybride d'encadrement des systèmes dynamiques continus, pour construire ces deux observateurs de sorte que nous garantissons la positivité de l'erreur d'observation. De plus, dans ce contexte, nous nous appuyons sur la démonstration donnée dans [81] pour déterminer le domaine de la matrice gain d'observation qui assure la convergence de l'erreur d'observation vers une constante sans faire recours à la notion de *faisceau* d'observateurs intervalles.

VI.3.2 Observateur intervalle

Le modèle général de cet observateur intervalle est donné par le système d'équations différentielles couplées ci-dessous :

$$\begin{cases} \dot{\overline{\mathbf{x}}}(t) = (\mathbf{A}-\mathbf{KC})\overline{\mathbf{x}}(t) + \overline{\Psi}(\underline{\mathbf{x}},\overline{\mathbf{x}},\underline{\mathbf{p}},\overline{\mathbf{p}},\mathbf{u}(t)) + \mathbf{K}\overline{\mathbf{y}}_m(t) \\ \dot{\underline{\mathbf{x}}}(t) = (\mathbf{A}-\mathbf{KC})\underline{\mathbf{x}}(t) + \underline{\Psi}(\underline{\mathbf{x}},\overline{\mathbf{x}},\underline{\mathbf{p}},\overline{\mathbf{p}},\mathbf{u}(t)) + \mathbf{K}\underline{\mathbf{y}}_m(t) \end{cases} \tag{VI.23}$$

où la première équation correspond à la dynamique de l'observateur maximal et la seconde correspond à la dynamique de l'observateur minimal. Les fonctions $\overline{\Psi}(.)$ et $\underline{\Psi}(.)$ sont construites en analysant la monotonie de $\Psi(.)$ par rapport à ses paramètres incertains et ses variables d'état, par application de la **Règle 2** et de la *méthode hybride d'encadrement* dans le cas général. Donc, elles remplissent, à chaque instant, la double inégalité ci-après :

$$\begin{gathered} \forall t \geq t_0, \forall \mathbf{x}(t_0) \in [\mathbf{x}_0], \forall \mathbf{p} \in [\mathbf{p}] \\ \underline{\Psi}(\underline{\mathbf{x}},\overline{\mathbf{x}},\underline{\mathbf{p}},\overline{\mathbf{p}},\mathbf{u}(t)) \leq \Psi(\mathbf{x},\mathbf{p},\mathbf{u}(t)) \leq \overline{\Psi}(\underline{\mathbf{x}},\overline{\mathbf{x}},\underline{\mathbf{p}},\overline{\mathbf{p}},\mathbf{u}(t)). \end{gathered} \tag{VI.24}$$

Enfin la matrice $\mathbf{K}$ de (VI.23) représente le gain d'observation. En pratique, un choix adéquat de cette matrice assure la convergence de l'erreur d'observation vers une constante donnée.

VI.3.3 Positivité de l'erreur d'observation

Tout d'abord, nous soulignons que pour tout $\mathbf{K}$ les solutions de (VI.17) sont aussi solutions du système suivant :

$$\dot{\mathbf{x}}(t) = (\mathbf{A}-\mathbf{KC})\mathbf{x}(t) + \Psi(\mathbf{x},\mathbf{p},\mathbf{u}(t)) + \mathbf{K}\overline{\mathbf{y}}_m(t). \tag{VI.25}$$

Ainsi, pour déterminer le signe de l'erreur d'observation de l'observateur maximal $\overline{\mathbf{e}}(t) = \overline{\mathbf{x}}(t) - \mathbf{x}(t)$ nous devons analyser l'équation différentielle qui régit la dynamique de cette erreur, soit :

$$\dot{\overline{\mathbf{e}}}(t) = (\mathbf{A}-\mathbf{KC})\overline{\mathbf{e}}(t) + \overline{\Psi}(\underline{\mathbf{x}},\overline{\mathbf{x}},\underline{\mathbf{p}},\overline{\mathbf{p}},\mathbf{u}(t)) - \Psi(\mathbf{x},\mathbf{p},\mathbf{u}(t)) + \mathbf{K}\overline{\mathbf{y}}_m(t). \tag{VI.26}$$

Etant donné que la matrice $\mathbf{K}$ satisfait l'**Hypothèse 2** et les fonctions $\overline{\Psi}(.)$ et $\Psi(.)$ satisfont la double inégalité (VI.24) alors nous pouvons écrire :

$$\overline{\Psi}(\underline{\mathbf{x}},\overline{\mathbf{x}},\underline{\mathbf{p}},\overline{\mathbf{p}},\mathbf{u}(t)) - \Psi(\mathbf{x},\mathbf{p},\mathbf{u}(t)) + \mathbf{K}\overline{\mathbf{y}}_m(t) \geq 0 \quad \forall t \geq t_0.$$

Par conséquent, le système (VI.26) est coopératif avec une entrée positive. Dès lors, selon la **Propriété 2** des systèmes coopératifs, nous pouvons affirmer que, si $\overline{\mathbf{e}}(t_0) \geq 0$ alors :

$$\forall t \geq t_0 \;\; \overline{\mathbf{e}}(t) \geq 0. \tag{VI.27}$$

Ainsi, nous venons de démontrer la positivité de l'erreur d'observation $\overline{\mathbf{e}}(t)$ de l'observateur maximal. Avec un raisonnement similaire, et en définissant l'erreur d'observation de l'observateur minimal par $\underline{\mathbf{e}}(t) = \mathbf{x}(t) - \underline{\mathbf{x}}(t)$, nous démontrons également la positivité de cette erreur d'observation.
Enfin, il est clair que l'erreur d'observation globale $\mathbf{e}(t) = \overline{\mathbf{e}}(t) + \underline{\mathbf{e}}(t) = \overline{\mathbf{x}}(t) - \underline{\mathbf{x}}(t)$ est positive pour tout $t \geq 0$, c'est-à-dire que :

$$\begin{gathered} \forall \mathbf{x}(t_0) \in [\underline{\mathbf{x}}(t_0), \overline{\mathbf{x}}(t_0)], \quad \forall \mathbf{p} \in [\mathbf{p}] \quad \forall t > t_0 \\ \underline{\mathbf{x}}(t) \leq \mathbf{x}(t) \leq \overline{\mathbf{x}}(t) \end{gathered} \tag{VI.28}$$

où $\overline{\mathbf{x}}(t)$ et $\underline{\mathbf{x}}(t)$ sont les solutions de (VI.23) et $\mathbf{x}(t)$ représente toutes les solutions possibles de (VI.17).

VI.4 Applications

Pour illustrer cette approche de construction observateurs d'intervalles, nous considérons deux exemples tirés du domaine de la biotechnologie [18]. Dans le premier exemple, l'application de la **Règle 2** est suffisante pour construire les systèmes englobants. En revanche, pour le deuxième exemple la propriété de monotonie du système par rapport à ses variables incertaines n'est pas satisfaite sur toute la période de simulation, donc nous devrons utiliser la méthode hybride d'encadrement.

VI.4.1 Exemple 1

Dans ce premier exemple nous supposons que le taux de croissance de la biomasse du bioréacteur est donné par le modèle de Contois [18, 26]. Ainsi, le modèle du bioréacteur est

$$\left\{ \begin{array}{rcl} \dot{x}_1 & = & \frac{a_1 x_1 x_2}{a_2 x_1 + x_2} - u x_1 \\ \dot{x}_2 & = & -\frac{a_3 a_1 x_1 x_2}{a_2 x_1 + x_2} - u x_2 + u a_4 \end{array} \right. \tag{VI.29}$$

où le vecteur de paramètres mal connus est $\mathbf{p} = [a_1\ a_4]^T = [0.95, 1.05] \times [0.095, 0.105]$, les autres paramètres sont $a_2 = a_3 = 1$. L'état initial de système est incertain $x_1 = [0.001, 0.1]$ et $x_2 = [0.001, 0.1]$. L'entrée est considérée constante par morceau $u(t) = 0.08$ pour $t \leq 10h$, $u(t) = 0.02$ pour $10h \leq t \leq 20h$ et encore une fois $u(t) = 0.08$ pour $t \geq 20h$. La sortie de ce système est donnée par

$$y(t) = x_1(t). \tag{VI.30}$$

L'erreur maximale de sortie qu'on peut commettre sur la mesure est $E(t) = \pm 2\% y_m(t)$.

Le système (VI.29) n'est pas coopératif. Cependant il est facile de vérifier que toutes les composantes de son champ de vecteurs sont monotones par rapport à chaque variable d'état et par rapport à chaque paramètre incertain. En utilisant la **Règle 1**, nous obtenons l'observateur intervalle suivant

$$\begin{cases} \dot{\overline{x}}_1 &= -u\overline{x}_1 + \frac{\overline{a}_1\overline{x}_1\overline{x}_2}{a_2\overline{x}_1+\overline{x}_2} + k_1(\overline{y}_m - \overline{x}_1) \\ \dot{\overline{x}}_2 &= -u\overline{x}_2 - \frac{a_3\underline{a}_1\underline{x}_1\overline{x}_2}{a_2\underline{x}_1+\overline{x}_2} + u\overline{a}_4 + k_2(\overline{y}_m - \overline{x}_1) \\ \dot{\underline{x}}_1 &= -u\underline{x}_1 + \frac{\underline{a}_1\underline{x}_1\underline{x}_2}{a_2\underline{x}_1+\underline{x}_2} + k_1(\underline{y}_m - \underline{x}_1) \\ \dot{\underline{x}}_2 &= -u\underline{x}_2 - \frac{a_3\overline{a}_1\overline{x}_1\underline{x}_2}{a_2\overline{x}_1+\underline{x}_2} + u\underline{a}_4 + k_2(\underline{y}_m - \underline{x}_1). \end{cases} \tag{VI.31}$$

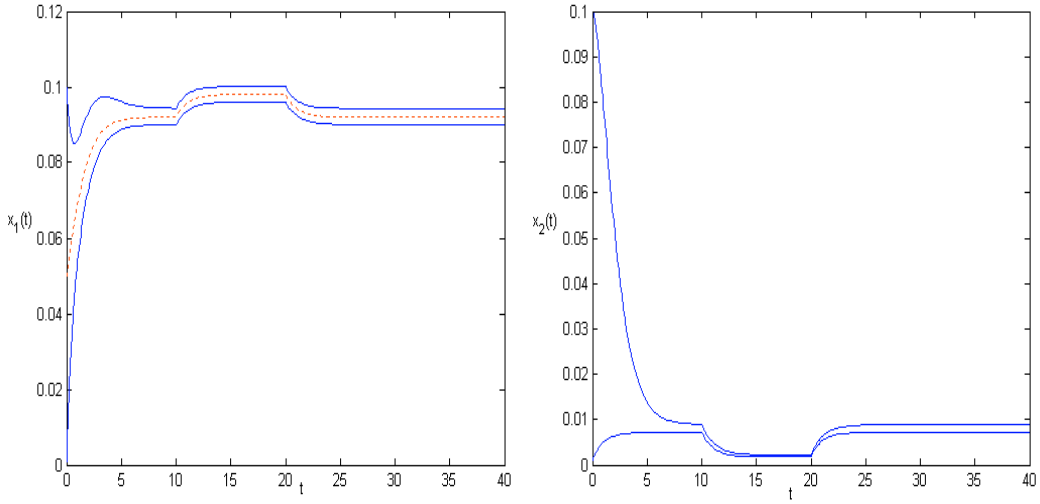
Le vecteur gain d'observation $\mathbf{K}^T = (k_1, k_2)$ doit assurer la coopérativité et la stabilité de la matrice $\mathbf{A} - \mathbf{KC}$, c'est-à-dire la positivité et la convergence de l'erreur d'observation. La matrice $\mathbf{A} - \mathbf{KC}$ est donnée ci-dessous

$$\mathbf{A} - \mathbf{KC} = \begin{bmatrix} -u - k_1 & 0 \\ -k_2 & -u \end{bmatrix} \tag{VI.32}$$

Ainsi, la condition de coopérativité et l'**Hypothèse 2** exigent que $k_2 = 0$. De plus, les valeurs propres $\lambda_1 = -(u + k_1)$ et $\lambda_2 = -u$ de la matrice (VI.32) doivent être assez négatives. Donc, k_1 doit être suffisamment positif pour assurer la convergence de l'erreur d'observation.

VI.4.1.1 Tests et résultats

Pour illustrer le comportement de cet observateur intervalle, nous réalisons plusieurs tests de simulation. La pseudo mesure est obtenue par la simulation de (VI.29) dans lequel l'état initial et les paramètres sont parfaitement connus : $x_1 = 0.05$, $x_2 = 0.05$, $a_1 = 1$ et $a_4 = 0.1$.



(a) (b)

Fig. VI.17 *: Encadrement de toutes les trajectoires d'état possibles dans le contexte de l'essai 1 : (a)* $[x_1](t)$ *; (b)* $[x_2](t)$.

Essai 1 :

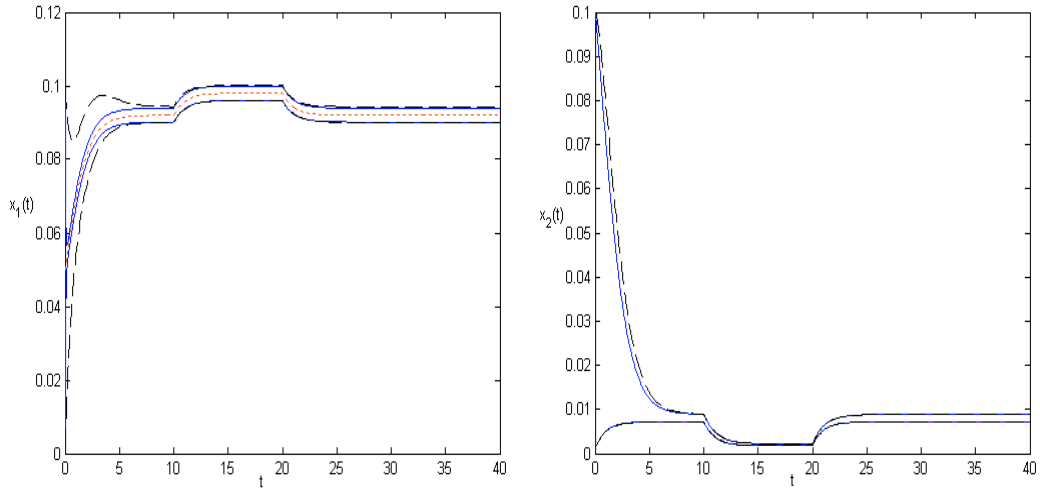
Dans ce premier cas d'étude, nous considérons les données présentées au début de la section V.4.1 et le vecteur gain d'observation $\mathbf{K}^T = (k_1, k_2) = (2, 0)$. L'observateur intervalle (VI.31) fournit les résultats tracés sur la figure VI.17.

Les courbes bleues continues représentent un encadrement garanti de toutes les trajectoires d'état générées par le système incertain (VI.29) et qui sont compatibles avec la mesure (VI.30) et ses incertitudes. Cette mesure est représentée par la courbe rouge pointillée sur la figure VI.17 (a).

Essai 2 : (Gain d'observation élevé)

Considérons dans ce cas un gain d'observation plus élevé que celui du premier essai $\mathbf{K}^T = (k_1, k_2) = (20, 0)$. La figure VI.18 montre les résultats de la simulation dans ce nouveau contexte.

Evidemment, comme la dynamique de l'erreur d'observation est beaucoup plus stable dans ce cas que dans celle du premier essai, la taille de l'encadrement est plus faible et les trajectoires convergent plus rapidement. En effet, les courbes bleues continues sur la figures VI.18 montrent l'encadrement des trajectoires d'état obtenu pour $\mathbf{K}^T = (20, 0)$, et les courbes noires discontinues représentent l'encadrement des trajectoires d'état obtenu



(a) (b)

Fig. VI.18 *: Encadrement de toutes les trajectoires d'état possibles dans le contexte du deuxième essai : (a) $[x_1](t)$; (b) $[x_2](t)$.*

pour $\mathbf{K}^T = (2, 0)$.

Essai 3 : (Grande incertitude sur l'état initial)

Pour ce troisième essai, nous augmentons l'incertitude sur l'état initial $x_1 = [0.001, 0.2]$, $x_2 = [0.001, 0.2]$ et nous utilisons le même pavé de paramètres et le même vecteur gain d'observation que ceux du deuxième essai.

Dans ce contexte, les encadrements des trajectoires d'état présentés sur la figure VI.19 (courbes bleues continues troisième essai, courbes noires discontinues deuxième essai) montrent qu'après une période de temps les deux encadrement se rejoignent. Effectivement, la taille des encadrements des trajectoires d'état en régime établi ne dépend que de la taille du pavé de paramètres et de la taille de domaine admissible de la mesure.

Essai 4 : (Incertitude importante sur les paramètres)

Maintenant, nous augmentons l'incertitude sur les paramètres $[a_1] = [0.8, 1.2]$, $a_2 = [0.08, 0.12]$ et nous observons son effet sur la taille des encadrements des trajectoires d'état admissibles.

La figure VI.20 montre bien que, contrairement au troisième essai où la taille du pavé d'état initial n'influence pas la taille de l'encadrement des trajectoires d'état en régime

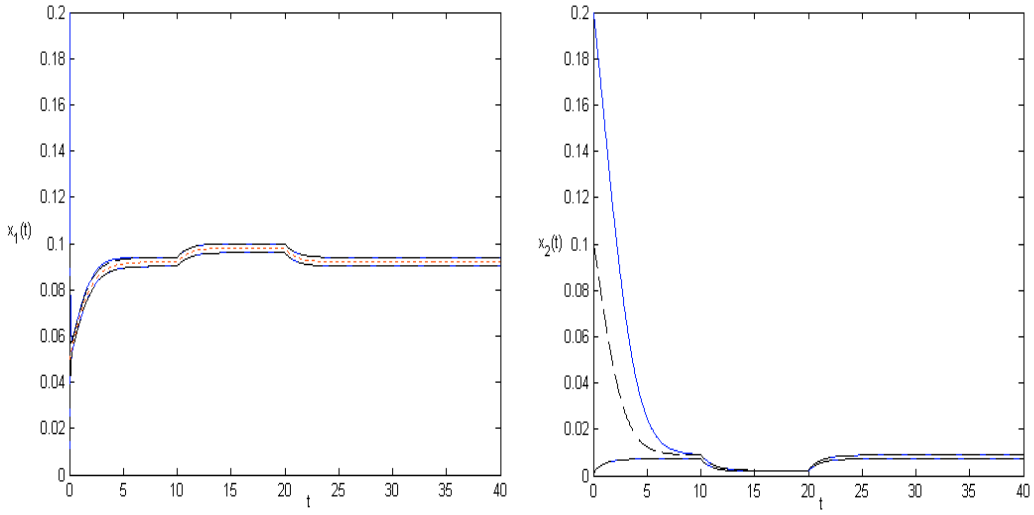


Fig. VI.19 *: Encadrement de toutes les trajectoires d'état possibles dans le contexte du troisième essai : (a) $[x_1](t)$; (b) $[x_2](t)$.*

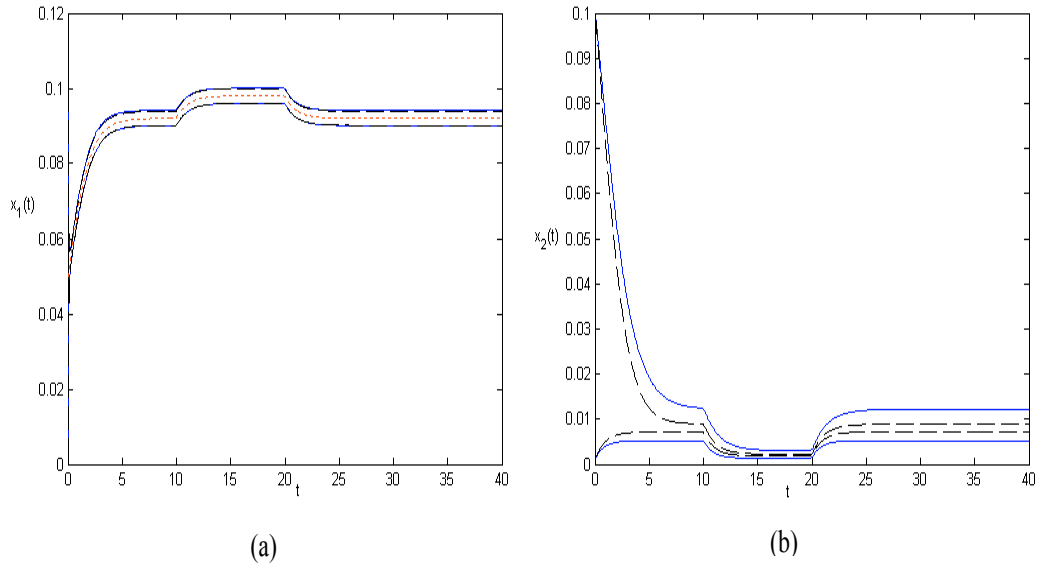


Fig. VI.20 *: Encadrement de toutes les trajectoires d'état possibles dans le contexte du quatrième essai : (a) $[x_1](t)$; (b) $[x_2](t)$.*

permanent, la taille du pavé de paramètres influence proportionnellement la taille de cet encadrement d'état. En effet, les courbes bleues continues représentent les résultats d'observation de ce quatrième essai et les courbes noires discontinues correspondent aux résultats du deuxième essai.

Enfin, malgré les incertitudes sur le système (VI.29) l'observateur intervalle (VI.31) estime de manière garantie une bande de trajectoires d'état bornée avec une taille qui dépend principalement de la taille du pavé de paramètres et de celle du domaine admissible de la mesure. Le tableau VI.3 récapitule les caractéristiques de simulation de chaque essai présenté ci-dessus.

	$[x_1] \times [x_2]$	$[a_1] \times [a_4]$	$\mathbf{K}^T$
Essai 1	$[0.001, 0.1] \times [0.001, 0.1]$	$[0.95, 1.05] \times [0.095, 0.105]$	$(2, 0)$
Essai 2	$[0.001, 0.1] \times [0.001, 0.1]$	$[0.95, 1.05] \times [0.095, 0.105]$	$(20, 0)$
Essai 3	$[0.001, 0.2] \times [0.001, 0.2]$	$[0.95, 1.05] \times [0.095, 0.105]$	$(20, 0)$
Essai 4	$[0.001, 0.1] \times [0.001, 0.1]$	$[0.8, 1.2] \times [0.08, 0.12]$	$(20, 0)$

Tab. VI.3 *: Caractéristiques de simulation de chaque essai d'observation, avec* $E(t) = \pm 2\% y_m(t)$.

Dans ce premier exemple, nous avons mis en application le principe de construction d'observateurs intervalles en boucle fermée et nous avons montré ces performances par rapport à taille du pavé d'état initial et à celle du pavé de paramètres.

VI.4.2 Exemple 2

Pour ce deuxième exemple nous reprenons le modèle de Haldane (VI.7), (VI.8) pour les bioréacteurs présenté dans la section **VI.2.3.2.2**. Donc nous avons les mêmes trois modes d'encadrement possibles (VI.13), (VI.14) et (VI.16). Par contre, les mesures sont disponibles en temps continu et leur domaine admissible est donné par

$$\mathbb{Y}(t) = [0.98X_m(t), 1.02X_m(t)] \tag{VI.33}$$

Ainsi, l'observateur intervalle proposé pour ce bioréacteur est considéré comme étant un système dynamique hybride avec trois modes continus et une condition de garde $S =$

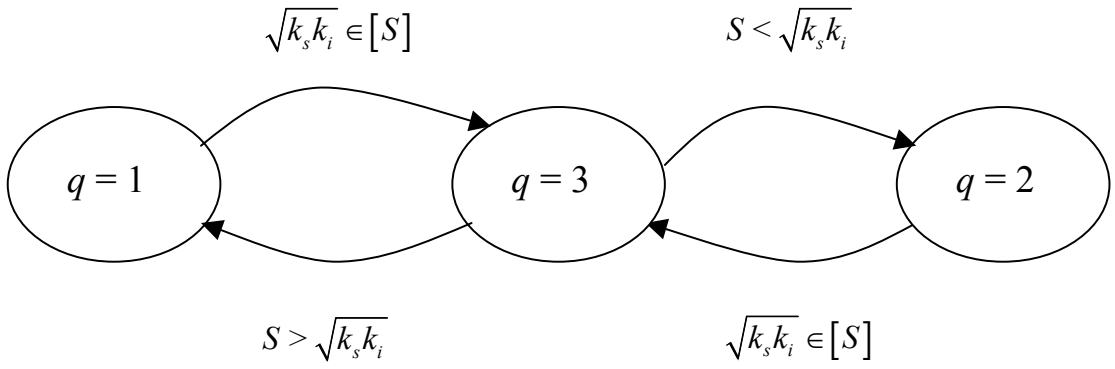


Fig. VI.21 *: Automate hybride d'observation.*

$\sqrt{k_s k_i}$. L'automate hybride qui gère cet observateur est représenté sur la figure VI.21, où $q = 1$ représente le premier mode d'encadrement, donné par la relation suivante :

$$(q=1)\begin{cases} \dot{\overline{X}}(t) = \overline{\mu}(\overline{S})\overline{X} - \alpha D\overline{X} + k_1(\overline{y}_m - \overline{X}_1) \\ \dot{\overline{S}}(t) = -k_1\underline{\mu}(\overline{S})\underline{X} + D(\overline{S}_{in} - \overline{S}) + k_2(\overline{y}_m - \overline{X}_1) \\ \dot{\underline{X}}(t) = \underline{\mu}(\underline{S})\underline{X} - \alpha D\underline{X} + k_1(\underline{y}_m - \underline{X}_1) \\ \dot{\underline{S}}(t) = -k_1\overline{\mu}(\underline{S})\overline{X} + D(\underline{S}_{in} - \underline{S}) + k_2(\underline{y}_m - \underline{X}_1) \end{cases} \tag{VI.34}$$

$q = 2$ représente le second mode donné par

$$(q=2)\begin{cases} \dot{\overline{X}}(t) = \overline{\mu}(\underline{S})\overline{X} - \alpha D\overline{X} + k_1(\overline{y}_m - \overline{X}_1) \\ \dot{\overline{S}}(t) = -k_1\underline{\mu}(\overline{S})\underline{X} + D(\overline{S}_{in} - \overline{S}) + k_2(\overline{y}_m - \overline{X}_1) \\ \dot{\underline{X}}(t) = \underline{\mu}(\overline{S})\underline{X} - \alpha D\underline{X} + k_1(\underline{y}_m - \underline{X}_1) \\ \dot{\underline{S}}(t) = -k_1\overline{\mu}(\underline{S})\overline{X} + D(\underline{S}_{in} - \underline{S}) + k_2(\underline{y}_m - \underline{X}_1). \end{cases} \tag{VI.35}$$

et le troisième mode d'encadrement $q = 3$ utilise l'encadrement trivial, il est donné par

$$(q=3)\begin{cases} \dot{\overline{X}}(t) = \overline{\mu}_0 \frac{\overline{S}}{\overline{S}+k_s+\underline{S}\overline{S}/k_i}\overline{X} - \alpha D\overline{X} + k_1(\overline{y}_m - \overline{X}_1) \\ \dot{\overline{S}}(t) = -k_1\underline{\mu}(\overline{S})\underline{X} + D(\overline{S}_{in} - \overline{S}) + k_2(\overline{y}_m - \overline{X}_1) \\ \dot{\underline{X}}(t) = \underline{\mu}_0 \frac{\underline{S}}{\underline{S}+k_s+\underline{S}\overline{S}/k_i}\underline{X} - \alpha D\underline{X} + k_1(\underline{y}_m - \underline{X}_1) \\ \dot{\underline{S}}(t) = -k_1\overline{\mu}(\underline{S})\overline{X} + D(\underline{S}_{in} - \underline{S}) + k_2(\underline{y}_m - \underline{X}_1). \end{cases} \tag{VI.36}$$

Comme dans le premier exemple, le vecteur gain d'observation $\mathbf{K}$ doit assurer la coopérativité et la stabilité de la matrice

$$\mathbf{A} - \mathbf{KC} = \begin{bmatrix} -\alpha D - k_1 & 0 \\ -k_2 & -\alpha \end{bmatrix}. \tag{VI.37}$$

Par conséquent, l'élément k_1 de ce vecteur doit être suffisamment grand et l'élément k_2 doit être nul (voir **Hypothèse 2**).

VI.4.2.1 Tests et résultats

L'évolution de cet automate hybride d'observation, pour $k_2 = 0$ et $k_1 \geq 0$, permet d'obtenir des encadrements garantis de toutes les trajectoires d'état compatibles avec le modèle et ses incertitudes ainsi qu'avec le domaine admissible de la mesure (VI.33). Dans la suite, nous réalisons plusieurs tests de simulation de cet observateur intervalle hybride. Nous considérons aussi dans ces tests que la mesure est obtenue par la simulation du système (VI.7) avec les données suivantes : $\alpha = 0.5$, $k = 42.14$, $k_s = 9.28 mmol/l$, $k_i = 256 mmol/l$, $\mu_0 = 0.74$, $X_0 = 5$, $S_0 = 40$, $S_{in}(t) = 65 + 15\cos(1/5t)$ et

$$D(t) = \begin{cases} 2 \text{ si } 0 \leq t \leq 5 \\ 0.5 \text{ si } 5 \leq t \leq 10 \\ 1.14 \text{ si } 10 \leq t \leq 20. \end{cases}$$

Essai 1 :

Considérons pour ce premier essai que l'état initial et les paramètres μ_0 et $S_{in}(t)$ sont incertains : $X_0 \in [3, 7]$, $S_0 \in [36, 44]$, $\mu_0 \in [0.69, 0.79]$ et $S_{in}(t) \in [64 + 15\cos(1/5t), 66 + 15\cos(1/5t)]$. Le vecteur gain d'observation est $\mathbf{K}^T = (2, 0)$.

Les résultats de la simulation dans ce contexte sont présentés sur la figure VI.22 où les courbes bleues continues représentent un encadrement garanti de toutes les trajectoires d'état admissibles, et la courbe rouge pointillée représente la pseudo mesure $X_m(t)$. Enfin, les transitions entre les trois modes d'encadrement de cet observateur intervalle hybride est illustrée sur la figure VI.23.

Essai 2 : (Effet du gain d'observation)

Dans ce deuxième cas d'étude, nous considérons un vecteur gain d'observation dix fois plus grand que celui du premier essai $\mathbf{K}^T = (20, 0)$. Par conséquent, comme le montre la figure VI.24, la taille de l'encadrement des trajectoires d'état associée à ce gain (courbes bleues continues) converge plus rapidement que celle obtenue dans le premier essai (courbes noires discontinues).

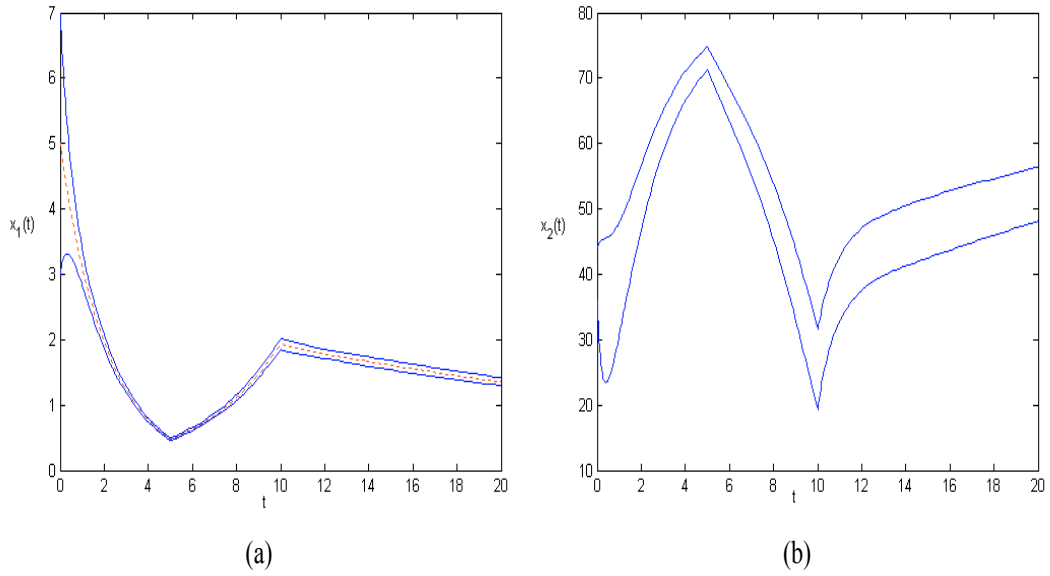


Fig. VI.22 *: Encadrement de toutes les trajectoires d'état admissibles : (a) $[x_1](t)$; (b) $[x_2](t)$.*

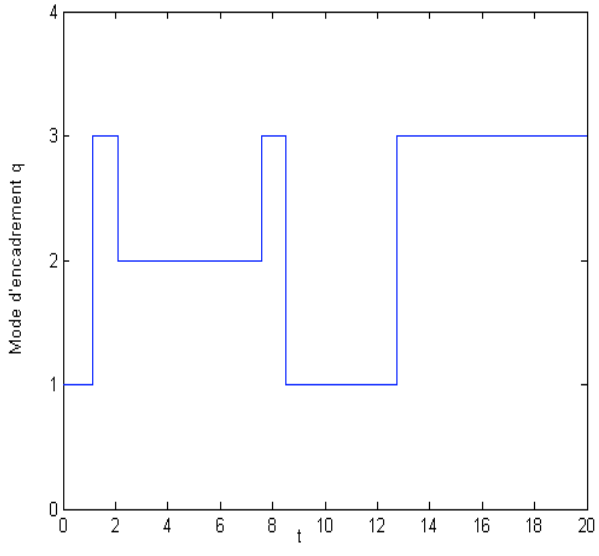


Fig. VI.23 *: Evolution du mode d'encadrement*

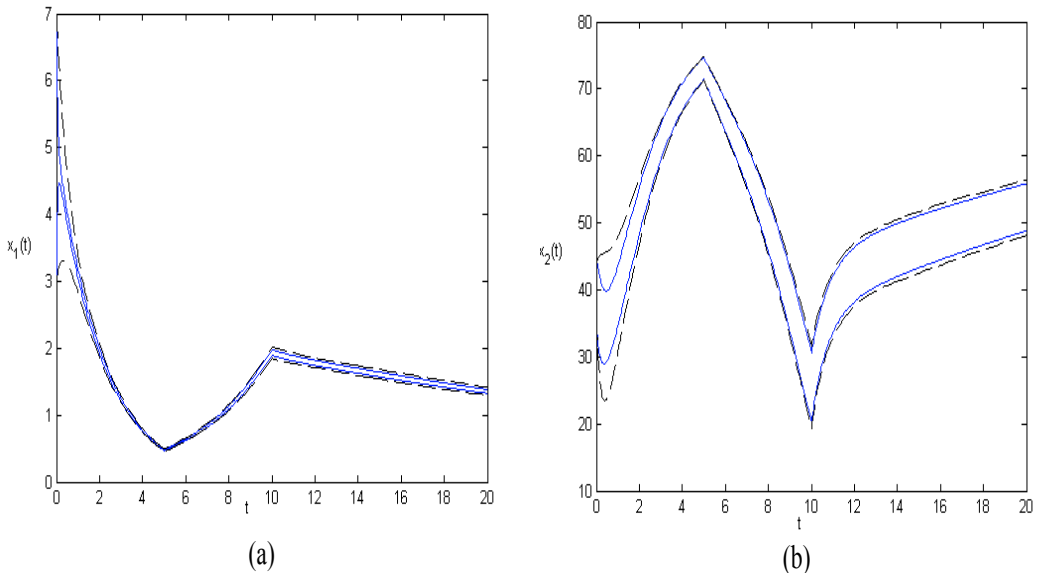


Fig. VI.24 *: Encadrement de toutes les trajectoires d'état possibles dans le contexte du deuxième essai : (a) $[x_1](t)$; (b) $[x_2](t)$.*

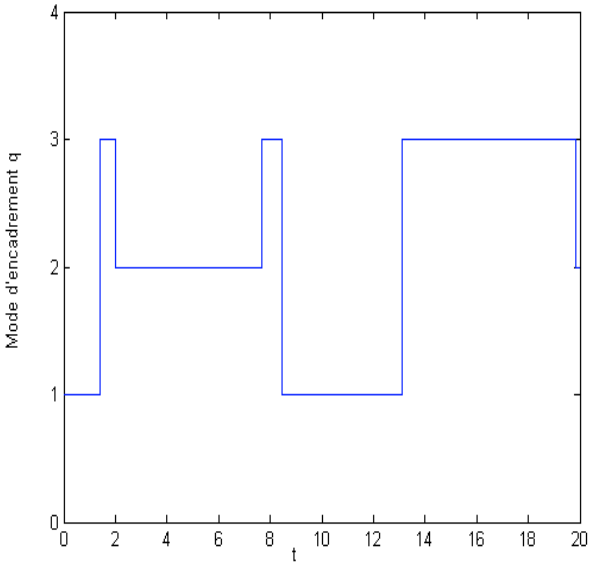


Fig. VI.25 *: Evolution du mode d'encadrement dans le contexte du deuxième essai*

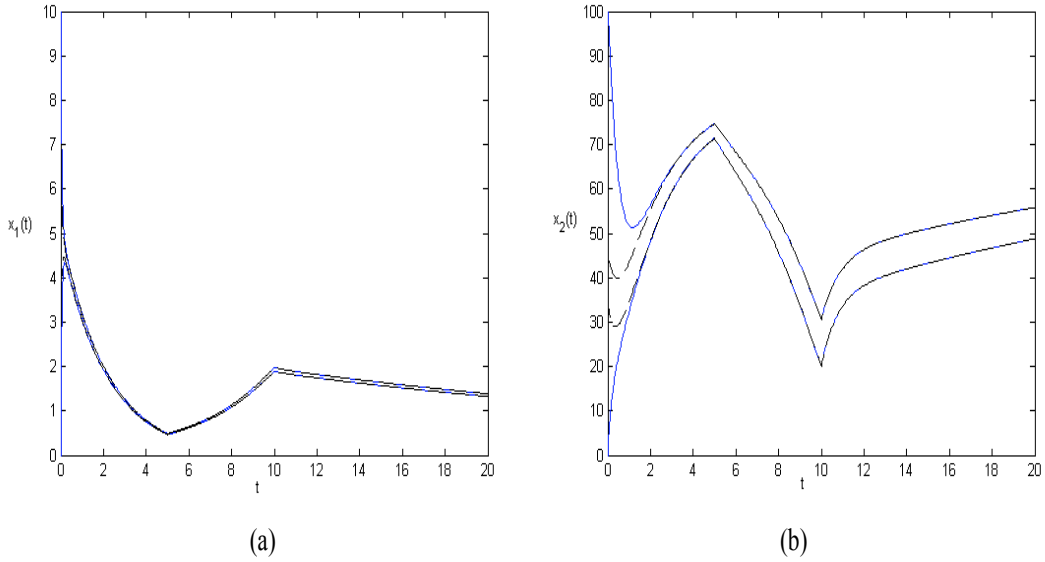


Fig. VI.26 *: Encadrement de toutes les trajectoires d'état possibles dans le contexte du troisième essai : (a)* $[x_1](t)$ *; (b)* $[x_2](t)$.

De plus, la précision de l'encadrement des trajectoires d'état obtenu dans ce deuxième essai est meilleure que celle de l'encadrement calculé dans le premier essai.

Essai 3 : (Effet de l'incertitude sur l'état initial)

L'état initial, dans ce troisième essai, est considérée fortement incertain $X_0 \in [0, 10]$, $S_0 \in [0, 100]$. Par ailleurs, son effet sur la taille de l'encadrement des trajectoires d'état est négligeable en régime permanent.

En effet, comme le montre la figure VI.26, après une période de simulation l'encadrement obtenu dans le contexte de ce troisième essai (courbes bleues continues), est confondu avec celui du deuxième essai (courbes noires discontinues).

Essai 4 : (Effet de l'incertitude sur les paramètres)

Pour ce dernier essai, nous augmentons l'incertitude sur les paramètres $\mu_0 \in [0.64, 0.84]$ et $S_{in}(t) \in [62 + 15\cos(1/5t), 68 + 15\cos(1/5t)]$ et nous observons son effet sur la taille de l'encadrement des trajectoires d'état admissibles.

Les courbes bleues continues sur la figure (VI.28) montrent que la taille de l'encadrement des trajectoires d'état dépend proportionnellement de la taille du pavé de paramètres. En effet, les courbes noires discontinues représentent l'encadrement des trajec-

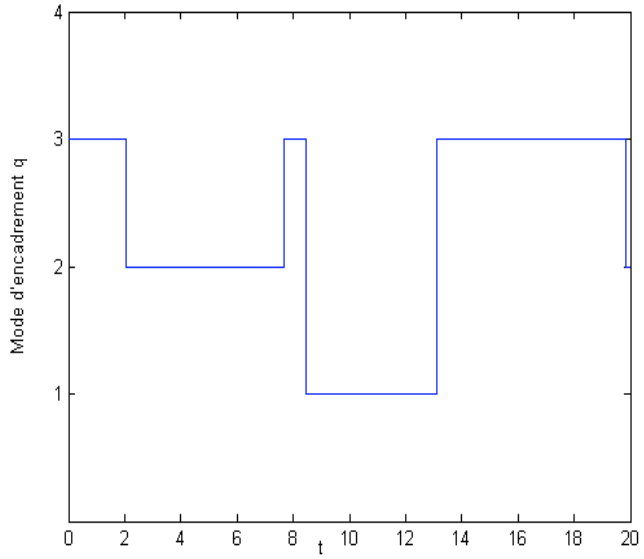


Fig. VI.27 *: Evolution du mode d'encadrement dans le contexte du troisième essai*

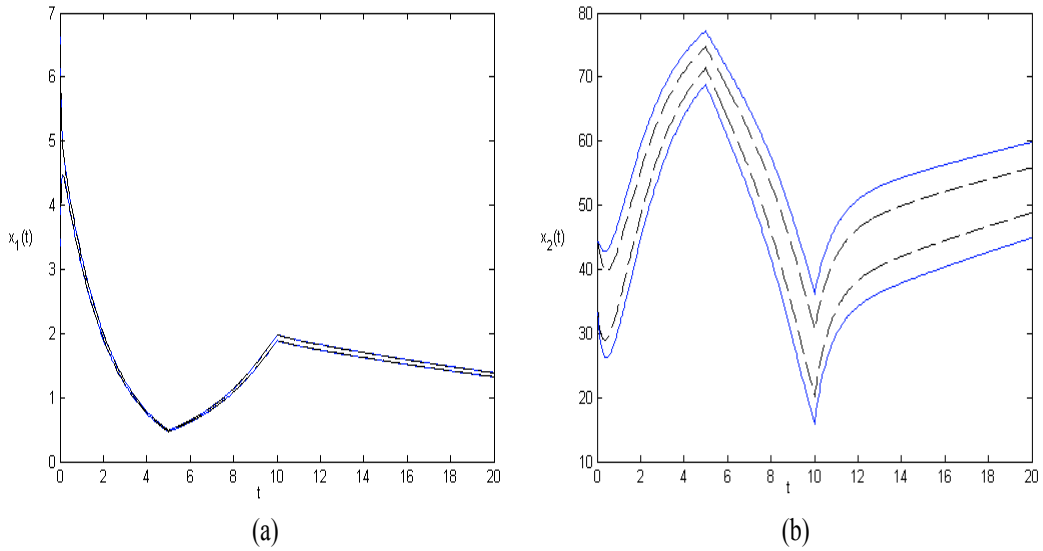


Fig. VI.28 *: Encadrement de toutes les trajectoires d'état possibles dans le contexte du quatrième essai : (a)* $[x_1](t)$ *; (b)* $[x_2](t)$*.*

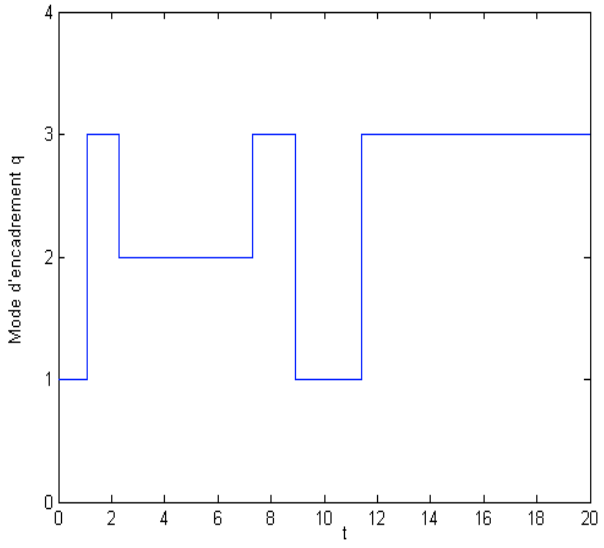


Fig. VI.29 *: Evolution du mode d'encadrement dans le contexte du quatrième essai.*

toires d'état du deuxième essai et le courbes bleues continues correspondent à l'encadrement calculé dans ce quatrième essai. Donc, de manière générale pour ce genre d'observateurs intervalles, il apparaît que la taille de l'encadrement de toutes les trajectoires d'état compatibles avec le modèle et les données expérimentales dépend de la taille du pavé de paramètres.

Le tableau VI.4 résume les caractéristiques de simulation de chaque essai arboré ci-dessus.

	$[X_0] \times [S_0]$	$[\mu_0]$	$[S_{in}](t)$	$\mathbf{K}^T$
Essai 1	$[3,7] \times [36,44]$	$[0.69, 0.79]$	$[64,66] + 15\cos(1/5t)$	$(2,0)$
Essai 2	$[3,7] \times [36,44]$	$[0.69, 0.79]$	$[64,66] + 15\cos(1/5t)$	$(20,0)$
Essai 3	$[0,10] \times [0,100]$	$[0.69, 0.79]$	$[64,66] + 15\cos(1/5t)$	$(20,0)$
Essai 4	$[3,7] \times [36,44]$	$[0.64, 0.84]$	$[62,68] + 15\cos(1/5t)$	$(20,0)$

Tab. VI.4 *: Caractéristiques de simulation de chaque essai d'observation, avec* $\mathbb{Y}(t) = [0.98X_m(t), 1.02X_m(t)]$.

Enfin, dans ce deuxième exemple nous avons voulu montrer que notre méthode hybride pour l'encadrement des solutions d'EDO est nécessaire pour résoudre de manière globale

et garantie le problème d'observation pour certain cas des systèmes dynamiques continus incertains. Cependant, l'analyse de convergence de l'erreur d'observation dans ce contexte hybride reste un problème ouvert.

VI.5 Conclusion

Dans ce chapitre, deux approches d'observation garantie des systèmes dynamiques continus non linéaires et incertains ont été présentées. La première approche est de type prédiction-correction. Elle est utilisable dans le cas où les mesures recueillies sur les systèmes sont des signaux discrets. En revanche, la seconde approche traite le cas où les mesures sont des signaux continus dans le temps. Sous certaines hypothèses, nous avons montré qu'il possible de trouver une matrice gain d'observation suffisamment grande telle que l'erreur d'observation converge vers une constante qui ne dépend que des tailles de pavés des grandeurs incertaines du système observé.

Chapitre VII

Atteignabilité des systèmes dynamiques non linéaires incertains

VII.1 Introduction

Un prérequis nécessaire à l'analyse d'atteignabilité des systèmes dynamiques hybrides (SDH) est le calcul de l'ensemble atteignable des systèmes dynamiques continus incertains, c'est-à-dire l'obtention de toutes les trajectoires d'état générées par un domaine d'état initial. La caractérisation exacte de cet ensemble est généralement impossible, à l'exception de quelques cas simples (dits décidables) [74, 1, 42] dont les dynamiques continues sont définies par $\dot{\mathbf{x}}(t) = \mathbf{c}$ où les composantes du vecteur $\mathbf{c}$ sont des constantes réelles ou intervalles. Dans la littérature il existe deux types d'approches pour résoudre ce problème. La première approche repose sur des méthodes du calcul ensembliste qui consistent à fournir une sur-approximation de l'espace d'état atteignable sous la forme d'union d'une même forme géométrique. Par exemple, des polyèdres [20], des ellipsoïdes [56, 57] ou encore des zonotopes [28]. La seconde approche consiste à partitionner l'espace d'état en plusieurs parties de même propriétés vis-à-vis la dynamique continue, puis à approcher localement la dynamique complexe par des dynamiques simples faciles à englober [7, 58, 6]. Cependant, l'application de ces méthodes est souvent limitée à cause de la dimension élevée et la complexité des modèles représentatifs des systèmes réels.

Récemment, l'analyse par intervalles [68, 92] et l'intégration numérique garantie des équations différentielles ordinaires (EDO) basée sur des modèles de Taylor intervalles

[59, 84, 69] ont été aussi utilisées pour résoudre le problème d'atteignabilité des SDH complexes ou fortement non linéaires [41, 83]. Mais la surestimation, de l'espace d'état atteignable obtenu par ces modèles, engendrée par le phénomène d'enveloppement limite leur exploitation pour résoudre certains cas réels. En effet, comme mentionné dans le chapitre II et illustré dans les chapitres V et VI, lorsque la taille du domaine initial de l'état et/ou de celui des paramètres est large, ces méthodes fournissent généralement des domaines atteignables dont le volume diverge après quelques pas d'intégration. Pour résoudre ce problème, nous proposons d'utiliser dans ce chapitre les méthodes d'intégration numérique garantie basées sur les théorèmes de la comparaison des inégalités différentielle présentées dans les chapitres III et IV, en particulier la méthode hybride pour l'encadrement des solutions d'EDO incertaines. Nous montrons également par le biais d'exemples, l'avantage de cette méthode par rapport aux méthodes basées sur les modèles de Taylor intervalles et le polynôme de Hermite-Obreschkoff.

VII.2 Définition des systèmes dynamiques hybrides

Les systèmes dynamiques hybrides *continu-discret* (SDH) [5, 104, 25] 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 [2, 1, 35].

VII.3 Analyse des systèmes dynamiques hybrides

L'analyse d'un SDH consiste à vérifier (*i*) des propriétés de sûreté, généralement spécifiées en terme de domaines de l'espace d'état hybride dans lesquels l'état doit rester ou ne jamais pénétrer, (*ii*) des propriétés de vivacité qui expriment des évolutions qui doivent être possibles ou inévitables, (*iii*) des propriétés de stabilité qui expriment que le système restera dans un ensemble invariant autour d'un point d'équilibre et (*iv*) des propriétés de performance. La vérification de ces propriétés peut être obtenue par une étude d'atteignabilité, c'est-à-dire par la détermination de la partie de l'espace d'état hybride atteignable par le système à partir d'un domaine initial [34, 35, 36].

L'étude d'atteignabilité des SDH par le biais de méthodes de simulations classiques, c'est-à-dire sans garantie, n'est pas satisfaisante car elles ne prennent pas en compte de toutes les évolutions possibles du SDH engendrées par un domaine d'état initial et/ou par les incertitudes sur les paramètres des systèmes dynamiques continus ; ce qui peut donc remettre en cause la validité de la vérification du bon comportement du système hybride. Pour surmonter cette insuffisance, deux types d'approches ont été développés pour la vérification de propriétés des SDH. Le premier type s'appuie sur une abstraction événementielle [16, 34, 35] qui consiste à transformer le modèle hybride en un modèle événementiel sur lequel il est adéquat de faire des vérifications. Le second type est basé sur le calcul d'une sur-approximation garantie de l'ensemble atteignable engendré par l'automate hybride [8, 40, 41, 7, 36, 22].

Le problème fondamental dans ces deux familles d'approches est celui du calcul de l'expansion continue d'un domaine initial, c'est-à-dire, de caractériser l'ensemble $\mathcal{R}([t_0, t]; \mathbb{X}_0)$ des trajectoires d'état générées à partir de ce domaine initial en fonction de la dynamique incertaine continue [34, 35],

$$\mathcal{R}([t_0, t]; \mathbb{X}_0) = \left\{ \begin{array}{c} \mathbf{x}(\tau),\ t_0 \leq \tau \leq t \,| \\ (\dot{\mathbf{x}}(\tau) = \mathbf{f}(\mathbf{x}, \mathbf{p}, \tau)) \wedge (\mathbf{x}(t_0) \in \mathbb{X}_0) \wedge (\mathbf{p} \in \mathbb{P}) \end{array} \right\} \tag{VII.1}$$

où $\mathbf{x}$ représente le vecteur d'état, $\mathbb{X}_0$ est le domaine fermé d'état initial et $\mathbb{P}$ est le support physique borné de paramètres incertains, $\mathbf{f}$ est le champs de vecteurs.

Dans la suite de ce chapitre, nous nous intéresserons au seul cas de l'expansion continue.

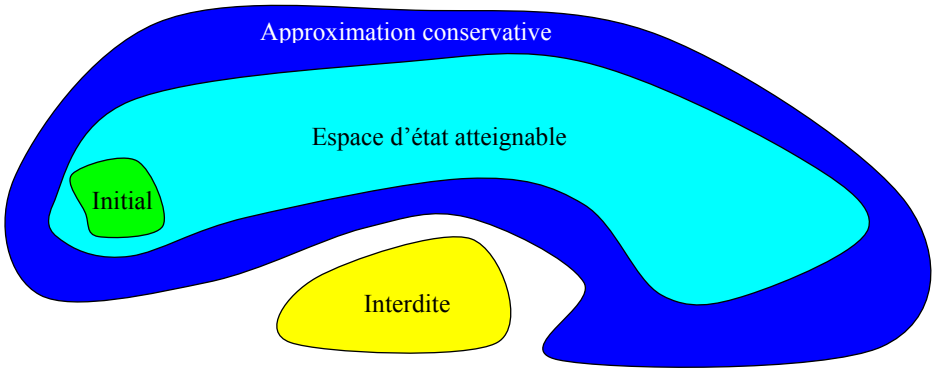


Fig. VII.1 *: Exemple de vérification de sûreté d'un système incertain*

VII.3.1 Preuve numérique de propriété

Par exemple, pour vérifier la propriété de sûreté d'un SDH, sur un horizon de temps fini, étant donné un ensemble d'état interdit ou un ensemble de pannes $\mathbb{U}$. Il faut prouver numériquement que

$$\forall t \in [t_0, T],\ \mathcal{R}([t_0, t]; \mathbb{X}_0) \cap \mathbb{U} = \emptyset \tag{VII.2}$$

où $T < \infty$.

La figure VII.1 montre le principe de cette vérification numérique. En effet, le domaine jaune représente l'ensemble d'états interdits et le domaine bleu clair représente l'espace d'état atteignable d'un système incertain engendré par l'ensemble de conditions initiales prises dans la région verte.

En pratique, compte tenu de la complexité des systèmes analysés, on cherche à calculer une approximation extérieure conservative $\overline{\mathcal{R}}([t_0, t]; \mathbb{X}_0)$ de l'espace atteignable

$$\mathcal{R}([t_0, t]; \mathbb{X}_0) \subseteq \overline{\mathcal{R}}([t_0, t]; \mathbb{X}_0), \tag{VII.3}$$

cette approximation conservative est représentée en bleu foncé sur la figure VII.1. Donc il est évident que si

$$\overline{\mathcal{R}}([t_0, t]; \mathbb{X}_0) \cap \mathbb{U} = \emptyset \Rightarrow \mathcal{R}([t_0, t]; \mathbb{X}_0) \cap \mathbb{U} = \emptyset \tag{VII.4}$$

VII.4 Approximation conservative de l'espace d'état atteignable

Dans cette section nous allons présenter une méthode de calcul d'approximations conservatives de l'espace atteignable en temps fini, qui s'appuie particulièrement sur la *solution a priori* des EDO incertaines définie dans le deuxième chapitre. Cette méthode est de type pas-à-pas. Elle caractérise l'ensemble $\overline{\mathcal{R}}([t_0,t];\mathbb{X}_0)$ par une union de vecteurs intervalles.

Pour $j=0,...,n_T-1$ et $t\in[t_j,t_{j+1}]$ nous définissons

$$[\mathbf{x}](t)=[\mathbf{x}_j]+\sum_{i=1}^{k-1}(t-t_j)^i\mathbf{f}^{[i]}([\mathbf{x}_j],[\mathbf{p}],t_j)+(t-t_j)^k\mathbf{f}^{[k]}([\xi_j],[\mathbf{p}],[t_j,t_{j+1}]) \tag{VII.5}$$

Proposition 1 :

$$Si\ [\xi_j]\supseteq[\tilde{\mathbf{x}}_j]\Rightarrow\forall t\in[t_j,t_{j+1}],\forall\mathbf{x}(t_j)\in[\mathbf{x}_j]\ \mathbf{x}(t)\in[\mathbf{x}](t),\ j=0,...,n_T-1 \tag{VII.6}$$

Preuve : Il suffit d'écrire le développement en série de Taylor au voisinage de t_j et d'utiliser la solution a priori $[\tilde{\mathbf{x}}_j]$ pour évaluer le terme reste du ce développement (voir [70]).

Ainsi pour calculer une approximation conservative de l'ensemble $\mathcal{R}([t_0,t];\mathbb{X}_0)$ on a la proposition suivante

Proposition 2 : *Une approximation conservative* $\overline{\mathcal{R}}([t_j,t];\mathbb{X}_0)$ *est donnée par,*

$$\forall t\in[t_j,t_{j+1}],\ \overline{\mathcal{R}}([t_j,t];[\mathbf{x}_j])=\cup_{\tau\in[t_j,t]}[\mathbf{x}](\tau),\ j=0,...,n_T-1 \tag{VII.7}$$

qui satisfait

$$\forall t\in[t_j,t_{j+1}],\ \overline{\mathcal{R}}([t_j,t];[\mathbf{x}_j])\subseteq[\tilde{\mathbf{x}}_j],\ j=0,...,n_T-1 \tag{VII.8}$$

Preuve : C'est évident selon la **Proposition 1**.

Définissons $\overline{\mathcal{R}}([t_0,t_0];[\mathbf{x}_0])=[\mathbf{x}_0]$. Alors on a la proposition ci-dessous

Proposition 3 : *L'approximation conservative de l'ensemble* (VII.1) *est donnée par,*

$$\forall t\in[t_j,t_{j+1}],\ \overline{\mathcal{R}}([t_0,t];[\mathbf{x}_0])=\overline{\mathcal{R}}([t_0,t_j];[\mathbf{x}_0])\cup\overline{\mathcal{R}}([t_j,t];[\mathbf{x}_j]),\ j=0,...,n_T-1 \tag{VII.9}$$

qui satisfait également

$$\forall t\in[t_j,t_{j+1}],\ \overline{\mathcal{R}}([t_0,t];[\mathbf{x}_0])\subseteq\cup_{i\in\{0,j\}}[\tilde{\mathbf{x}}_j],\ j=0,...,n_T-1 \tag{VII.10}$$

Preuve : C'est évident selon les **Propositions 1** et **2**.

Ainsi, il est pratique d'implémenter sur les calculateurs numériques la relation (VII.10) au lieu de (VII.9) pour obtenir une approximation conservative de l'espace d'état atteignable.

VII.5 Applications

Dans cette section nous montrons plusieurs exemples qui permettent d'illustrer le calcul d'approximations conservatives de l'espace d'état atteignables par des systèmes dynamiques continus incertains linéaires ou non linéaires. Pour cela, et pour chaque exemple, nous utilisons la méthode d'intégration numérique garantie la plus performante en terme de précision pour cette approximation.

VII.5.1 Exemple 1 : Système linéaire

Ce premier exemple est consacré au cas des systèmes dynamiques linéaires. Dans ce contexte, nous mettrons en évidence la capacité des méthodes d'intégration numérique garantie, basées sur les modèles de Taylor intervalles ou celles basées sur le polynôme de Hermite-Obreschkoff intervalle, à calculer une approximation conservative de taille raisonnable de l'espace d'état atteignable par des systèmes incertains suffisamment stables. Généralement, ceci est traduit par le fait que les parties réelles des valeurs propres de la matrice d'évolution du système sont très négatives. Pour cela, nous simulons le système dynamique ci-dessous (tiré de [28]) avec le logiciel VNODE [71] en utilisant la méthode de Hermite-Obreschkoff intervalle :

$$\dot{\mathbf{x}} = \begin{bmatrix} a_1 & a_2 & 0 & 0 & 0 \\ a_3 & a_4 & 1 & 0 & 0 \\ 0 & 0 & a_5 & a_6 & 0 \\ 0 & 0 & a_7 & a_8 & 0 \\ 0 & 0 & 0 & 0 & a_9 \end{bmatrix} \mathbf{x} + \begin{bmatrix} b_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \mathbf{u} \qquad \text{(VII.11)}$$

où les a_i, $i = 1, ..., 9$ et b_1 représentent les paramètres incertains du système, mais de supports physiques bornés et de bornes connues *a priori* : $a_1 = a_4 = [-1.1, -0.9]$, $a_3 =$

$-a_2 = [3.9, 4.1]$, $a_5 = a_8 = [-3.2, -2.8]$, $a_6 = -a_7 = [0.8, 1.2]$, $b_1 = [0.6, 1.2]$ et $a_9 = [-2.2, -1.8]$; le domaine d'état initial est $[x_i] = [0.8, 1.2]$, $i = 1, ..., n$; l'entrée $\mathbf{u}$ est un échelon unité. Le pas d'intégration est fixé constant $h = 0.04$, l'horizon temporel de la simulation est de 5 secondes et l'ordre de développement du polynôme de Hermite-Obreschkoff est égal à $8 + 8 + 1 = 17$.

Les figures VII.2 et VII.3 montrent respectivement deux projections de l'espace atteignable du système (VII.11) sur les plans (x_2, x_3) et (x_4, x_5). Et les figures VII.4 et VII.5 montrent l'expansion continue du domaine d'état initial relatif respectivement aux composantes x_1 et x_2 du vecteur d'état. En effet, les courbes en trait rouge pointillé représentent l'interpolation entre les solutions supérieures (resp. inférieures) du système (VII.11) aux instants $t_{i+1} = t_i + h, i = 1, ...\frac{5}{h}$; et les rectangles en trait bleu plein représentent la concaténation des solutions *a priori* $[\tilde{x}_{1i}]$ et $[\tilde{x}_{2i}]$ des variables d'état $[x_1]$ et $[x_2]$.

Remarque : La simulation de ce système par les méthodes d'intégration basées sur les modèles de Taylor intervalle donne des résultats similaires en terme de précision pour l'approximation conservative et de temps de calcul. Ces résultats sont exploitables compte tenu de la marge de stabilité assez large du système (VII.11). Néanmoins, dans le cas général des systèmes dynamiques linéaires, nous ne pouvons pas garantir que ces méthodes ne divergent pas sous l'effet de la propagation du pessimisme.

Dans les exemples suivants, nous montrons l'avantage des méthodes d'intégration numérique garantie basées sur les théorèmes de comparaison des inégalités différentielles.

VII.5.2 Exemple 2 : Système non linéaire coopératif

Pour illustrer l'application de la méthode hybride d'encadrement pour le calcul des approximations conservatives dans le cas des systèmes coopératifs, nous considérons un système non linéaire de diffusion thermique modélisé par un système d'équations différen-

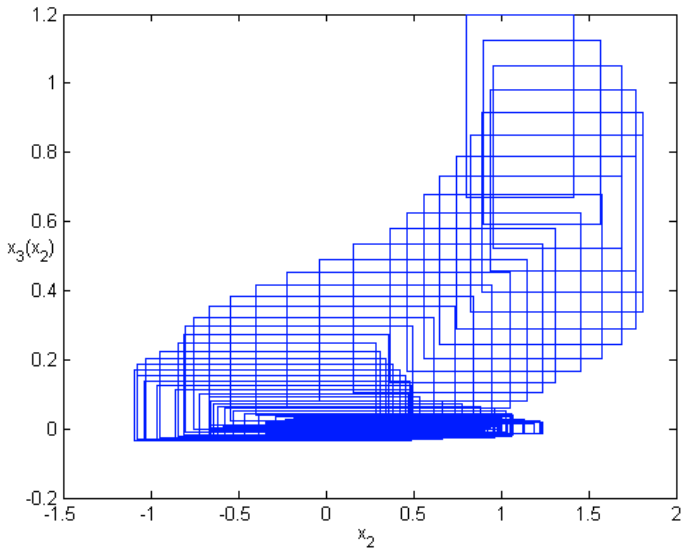


Fig. VII.2 *: Projection sur le plan (x_2, x_3) de l'espace atteignable du* (VII.11). *Simulation par la méthode de HOI.*

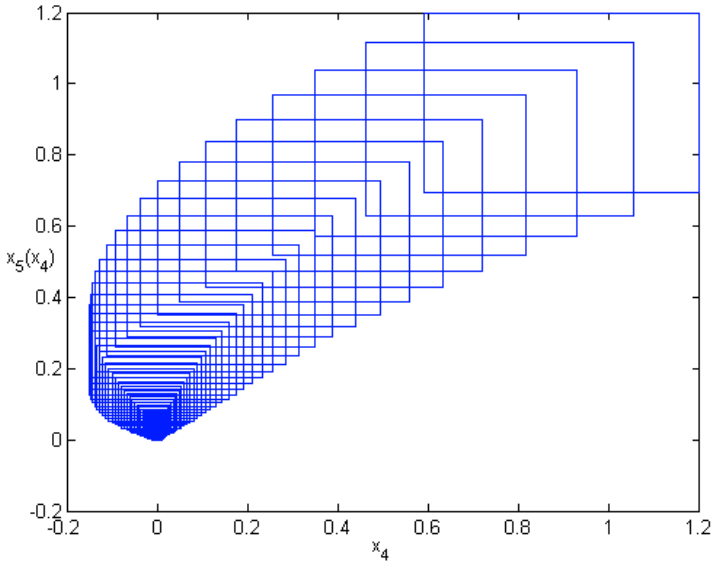


Fig. VII.3 *: Projection sur le plan (x_4, x_5) de l'espace atteignable du* (VII.11). *Temps CPU=0.5 sec (PIII 1.2MHz). Simulation par la méthode de HOI.*

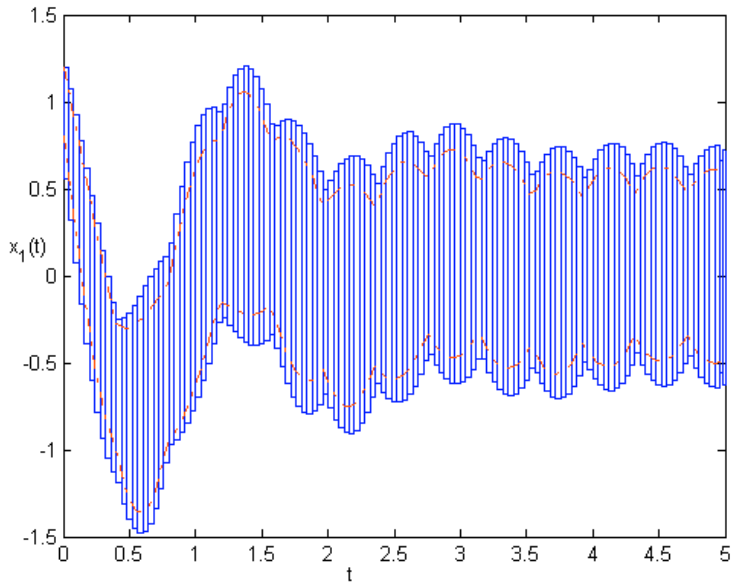


FIG. VII.4 *: L'évolution temporelle de $[x_1]$. Simulation par la méthode de HOI.*

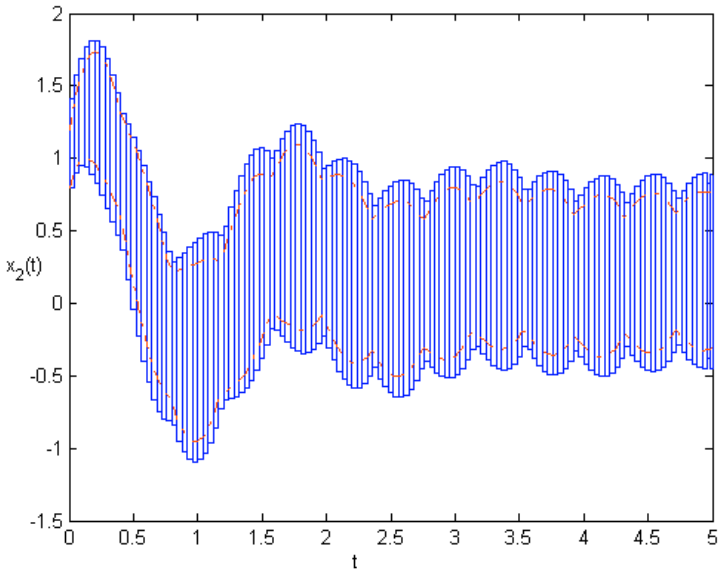


FIG. VII.5 *: L'évolution temporelle de $[x_2]$. Simulation par la méthode de HOI. Temps CPU=0.5 sec (PIII 1.2MHz)*

tielles non linéaires de dimension 13.

$$\begin{cases} \dot{x}_1 &= \alpha_1(x_2 - 2x_1 + u_0 + u(t)) \\ \dot{x}_2 &= 2\alpha_1(x_1 - (1 + \frac{\rho_1}{\rho_2})x_2 + \frac{\rho_1}{\rho_2}x_3) \\ \dot{x}_3 &= 2(p_0 + p_1x_3)(x_4 - x_3 + p_2\frac{\delta_2}{\rho_2}(x_2 - x_3)) \\ \dot{x}_i &= (p_0 + p_1x_i)(x_{i+1} - 2x_i + x_{i-1}) \qquad i = 4, \ldots, 9 \\ \dot{x}_{10} &= 2(p_0 + p_1x_{10})(x_9 - x_{10} + p_2\frac{\delta_2}{\rho_2}(x_{11} - x_{10})) \\ \dot{x}_{11} &= 2\alpha_2(x_{12} - (1 + \frac{\rho_3}{\rho_2})x_{11} + \frac{\rho_3}{\rho_2}x_{10}) \\ \dot{x}_{12} &= \alpha_2(x_{13} - 2x_{12} + x_{11}) \\ \dot{x}_{13} &= 2\alpha_2(x_{12} - (1 + \frac{\rho_3}{\rho_4})x_{13} + \frac{\rho_3}{\rho_4}u_0) \\ u(t) &= \sum_{l=1\ldots5} u_l sin(2^{l-1}\omega_0 t + \phi_0) \end{cases} \tag{VII.12}$$

où le vecteur de paramètres incertains est de dimension 3, $\mathbf{p} = (p_1,\, p_2,\, p_3)^{\mathrm{T}} \in [0.8, 1.2]\mathrm{s}^{-1} \times [0.01, 0.015]\mathrm{s}^{-1}\mathrm{K}^{-1} \times [0.35, 0.65]\mathrm{mW}^{-1}\mathrm{K}^{-1}$, l'état initial est pris dans un pavé défini par $x_{i0} \in [90,\, 110]°\mathrm{C}$, $i = 0, ..., 13$. $\mathbf{u} = (u_1,\, u_2)^{\mathrm{T}}$ est le vecteur d'entrée du système (VI.29), où u_1 est la somme de cinq sinusoïdes d'amplitudes et de fréquences différentes avec une valeur moyenne $u_0 = 100$, et $u_2 = u_0$. La période de simulation est $[T] = [0, 300]s$.

VII.5.2.1 Simulation par les modèles de Taylor intervalles

Dans les conditions de simulation décrites ci-dessus et comme les tailles du pavé d'état initial et du pavé de paramètres incertains sont considérables, toutes les tentatives de calcul d'une approximation conservative de l'espace d'état atteignable par (VII.12) par les méthodes intervalles d'intégration (modèles de Taylor intervalle, polynôme de Hermite-Obreschcoff) ont échoué. Evidemment, comme attendu, la propagation du pessimisme dans ces méthodes conduit à une divergence rapide de la taille de l'approximation conservative de l'espace d'état atteignable comme le montre les figures VII.6 et VII.7.

Sur la figure VII.6, on voit bien qu'après quelques pas d'intégration l'accumulation du pessimisme devient très importante ce qui entraîne l'arrêt de l'algorithme d'intégration. Dans la section suivante, nous montrons comment notre méthode hybride d'encadrement, en particulier l'algorithme **Hybrid_Bound_Solution** permet de contourner ce problème de pessimisme.

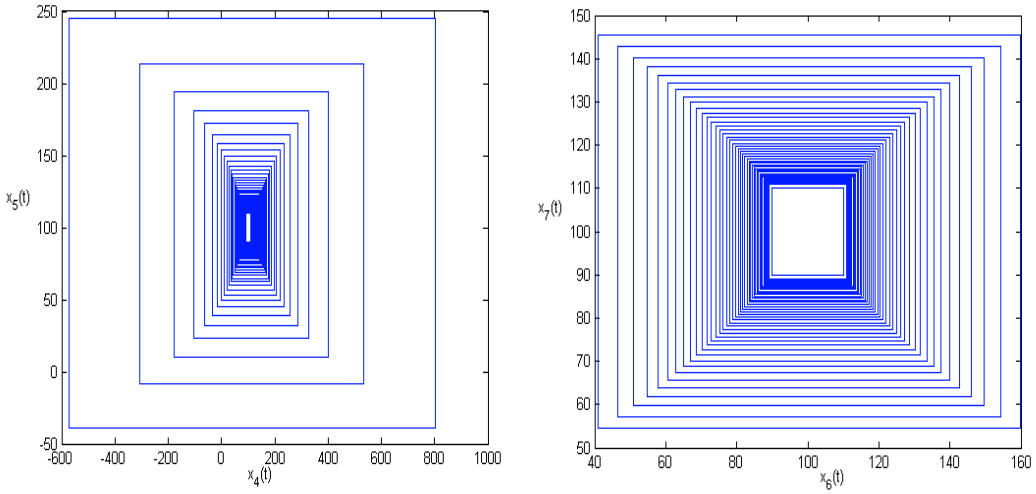


Fig. VII.6 *: Projections de l'espace atteignable de* (VII.12) *(a) sur le plan* (x_4, x_5)*, (b) sur le plan* (x_6, x_7)*. Simulation par la méthode de HOI.*

VII.5.2.2 Simulation par la méthode hybride d'encadrement

La simulation de (VII.12) avec la technique d'hybridation présentée dans l'algorithme **Hybrid_Bound_Solution**, permet de caractériser de manière garantie son espace d'état atteignable engendré par le domaine d'état initial et le pavé de paramètres incertains.

Les figures VII.8 et VII.9 représentent respectivement les projections de cet espace d'état atteignable sur les plans (x_1, x_9), (x_2, x_{10}), (x_4, x_{11}) et (x_5, x_{12}). Ces résultats sont obtenus pour une incertitude d'environ 50% sur les paramètres et de 20% sur le domaine d'état initial. Et les figures VII.10 et (VII.11) représentent respectivement les expansions continues dans le temps des domaines initiaux des variables d'état $[x_5](t)$, $[x_7](t)$ et $[x_9](t)$ selon la dynamique incertain de (VII.12). De plus, la figure VII.10 (b) représente un agrandissement de l'évolution de $[x_5](t)$ au voisinage de $t = 100$ qui montre la concaténation des solutions *a priori* $[\tilde{x}_{5i}](t)$ donnée par la relation (VII.10).

La figure VII.12 représente l'évolution du mode d'encadrement au cours de l'intégration garantie de (VII.12). Sur l'axe des ordonnées, nous avons présenté les indices des variables d'état discret qui correspondent aux modes d'encadrement locaux $q \in \mathbb{Q}$. La courbe rouge pointillée (resp. la courbe bleue en trait plein) montre les commutations successives entre les différents systèmes continus supérieurs (resp. inférieurs) au cours de l'évolution

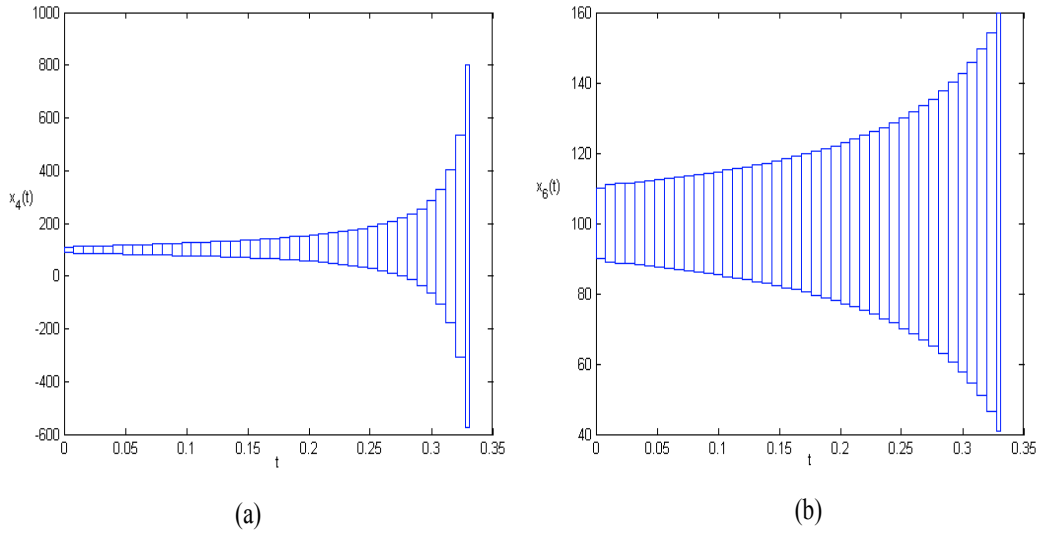


Fig. VII.7 *: L'évolution temporelle de variables d'état (a)* $[x_4](t)$*, (b)* $[x_6](t)$*. Simulation par la méthode de HOI.*

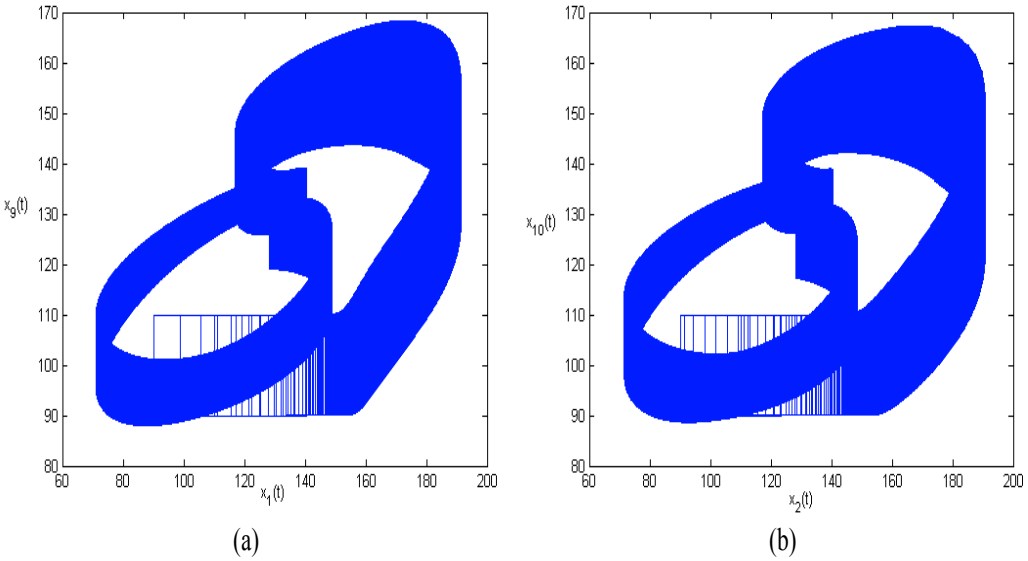


Fig. VII.8 *: Projections de l'espace atteignable de* (VII.12) *(a) sur le plan* (x_1, x_9)*, (b) sur le plan* (x_2, x_{10})*. Simulation par l'algorithme* ***Hybrid_Bound_Solution****. Temps CPU=250s.*

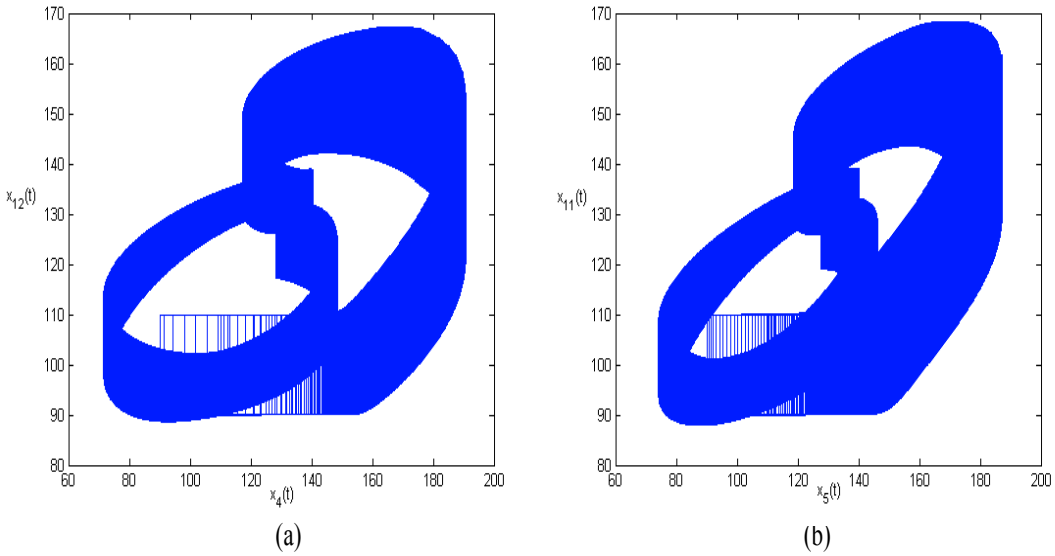


FIG. VII.9 *: Projections de l'espace atteignable de* (VII.12) *(a) sur le plan* (x_4, x_{12})*, (b) sur le plan* (x_5, x_{11})*. Simulation par l'algorithme* ***Hybrid_Bound_Solution****. Temps CPU=250s.*

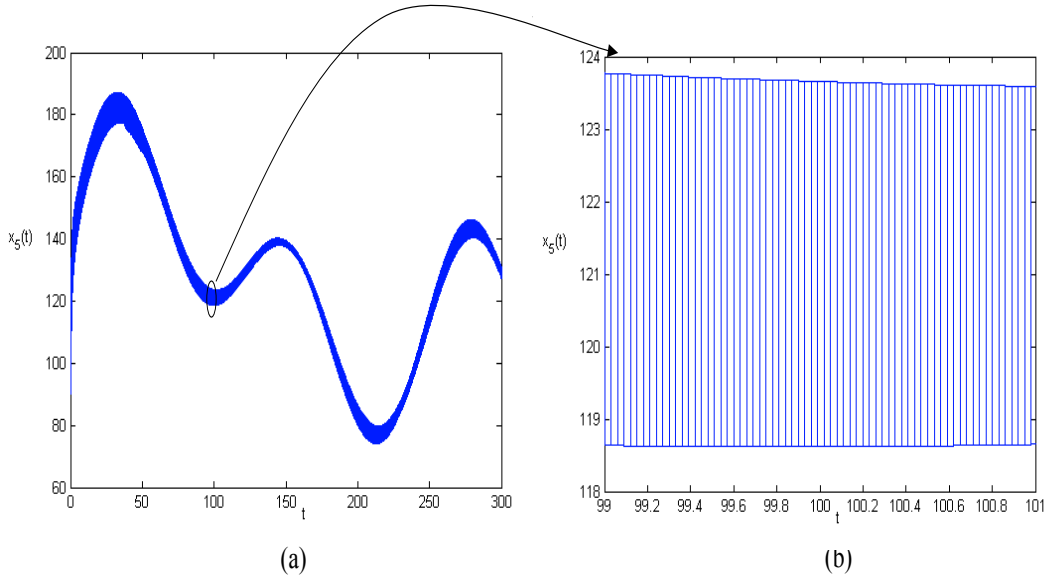


FIG. VII.10 *: L'évolution temporelle de la variable d'état* $[x_5](t)$*. Simulation par l'algorithme* ***Hybrid_Bound_Solution****. Temps CPU=250s.*

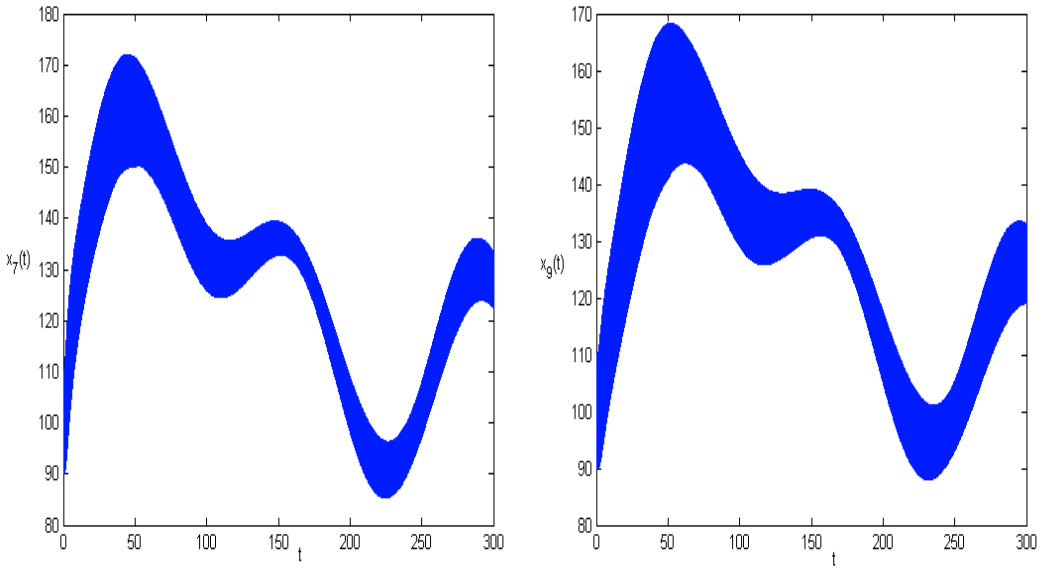


Fig. VII.11 *: L'évolution temporelle de variables d'état (a) $[x_7](t)$, (b) $[x_9](t)$. Simulation par l'algorithme* ***Hybrid_Bound_Solution****. Temps CPU=250s.*

du système incertain (VII.12). Théoriquement ce système (VII.12) admet $2^{18}+1$ modes d'encadrement, mais en pratique, dans les conditions de simulation données au début de la section VII.5.2, l'algorithme **Hybrid_Bound_Solution** n'a utilisé que $24+1$ modes d'encadrement.

VII.5.3 Exemple 3 : Système non linéaire non coopératif 1

Considérons le système dynamique non coopératif ci-dessous tiré du domaine des systèmes biologiques moléculaires [90]

$$\begin{cases} \dot{x_1} &= -\frac{v_2 x_1}{k_2+x_1} + v_0 u + v_1 \\ \dot{x_2} &= \frac{v_6(y_{tot}-x_2-x_3)}{k_6+(y_{tot}-x_2-x_3)} - \frac{v_3 x_1 x_2}{k_3+x_2} \\ \dot{x_3} &= \frac{v_4 x_1(y_{tot}-x_2-x_3)}{k_4+(y_{tot}-x_2-x_3)} - \frac{v_5 x_3}{k_5+x_3} \\ \dot{x_4} &= \frac{v_{10}(z_{tot}-x_4-x_5)}{k_{10}+(z_{tot}-x_4-x_5)} - \frac{v_7 x_3 x_4}{k_7+x_4} \\ \dot{x_5} &= \frac{v_8 x_3(z_{tot}-x_4-x_5)}{k_8+(z_{tot}-x_4-x_5)} - \frac{v_9 x_5}{k_9+x_5} \end{cases} \quad \text{(VII.13)}$$

où les paramètres sont mal connus, $v_0 \in [0.0014, 0.0015]$, $v_1 \in [0.08, 0.09]$, $v_2 \in [1.1, 1.2]$, $v_3 = v_4 \in [0.063, 0.064]$, $v_5 = v_6 = v_9 = v_{10} \in [4.9, 5]$, $v_7 = v_8 \in [0.05, 0.06]$, $y_{tot} \in [1199, 1200]$, $z_{tot} \in [299, 300]$, , $k_2 \in [199, 200]$, $k_3 = k_4 = k_5 = k_6 \in [1199, 1200]$, $k_7 = k_8 = k_9 = k_{10} \in [299, 300]$. Son vecteur d'état initial est parfaitement connu $x_1 = 165$,

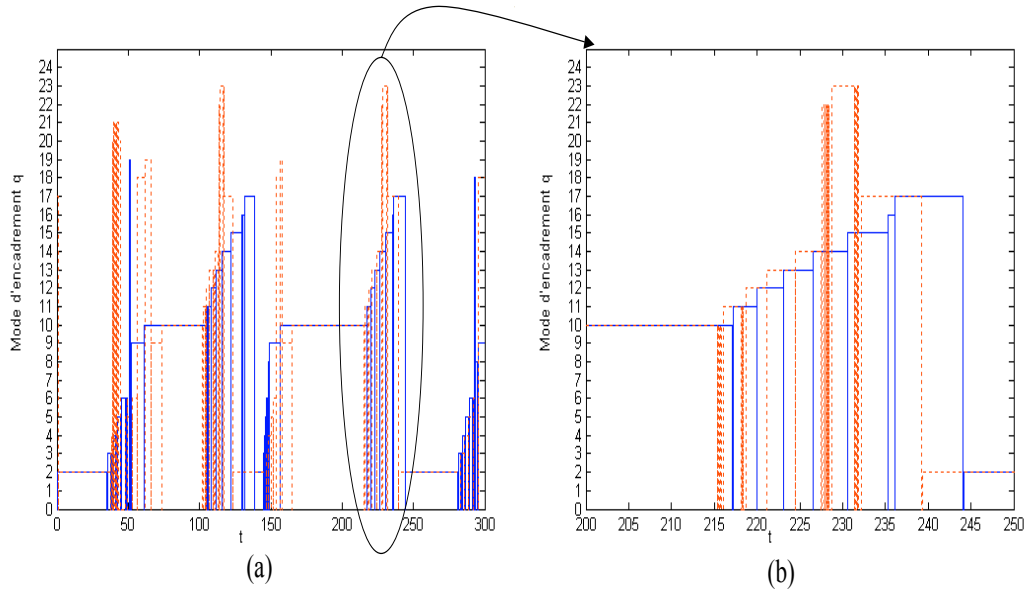


Fig. VII.12 *: L'évolution du mode d'encadrement (ligne continue : système supérieur, ligne discontinue : système inférieur). Simulation par l'algorithme* ***Hybrid_Bound_Solution****.*

$x_2 = 50$, $x_3 = 1150$, $x_4 = 5$, $x_5 = 250$, et son entrée $u = \frac{1}{0.98}x_5$. La période de simulation est $[T] = [0, 3000]$.

VII.5.3.1 Simulation par les méthodes intervalles d'intégration

Là aussi les méthodes d'intégration numérique garantie basées sur les modèles de Tylor intervalles et le polynôme de Hermite-Obreschkoff intervalle appliquées au système (VII.13) ne donnent pas de résultats satisfaisants comme le montrent les figures VII.13 à VII.16. On voit bien sur ces figures que l'effet d'enveloppement provoque l'arrêt de l'algorithme d'intégration après une courte durée de simulation. Donc, pour résoudre ce problème nous testons dans la section suivante les méthodes d'intégration garantie basées sur les théorèmes de comparaison des inégalités différentielles.

VII.5.3.2 Simulation par les méthodes d'encadrement dynamique

Comme le système (VII.13) n'est pas coopératif, nous devrons satisfaire les conditions du **Théorème 2** (théorème de Müller) pour garantir l'encadrement de toutes ses trajectoires d'état possibles. Pour ce faire, dans ce cas l'utilisation de la **Règle 2** permet d'obtenir les systèmes englobants couplés. En effet, toutes les dérivées partielles de

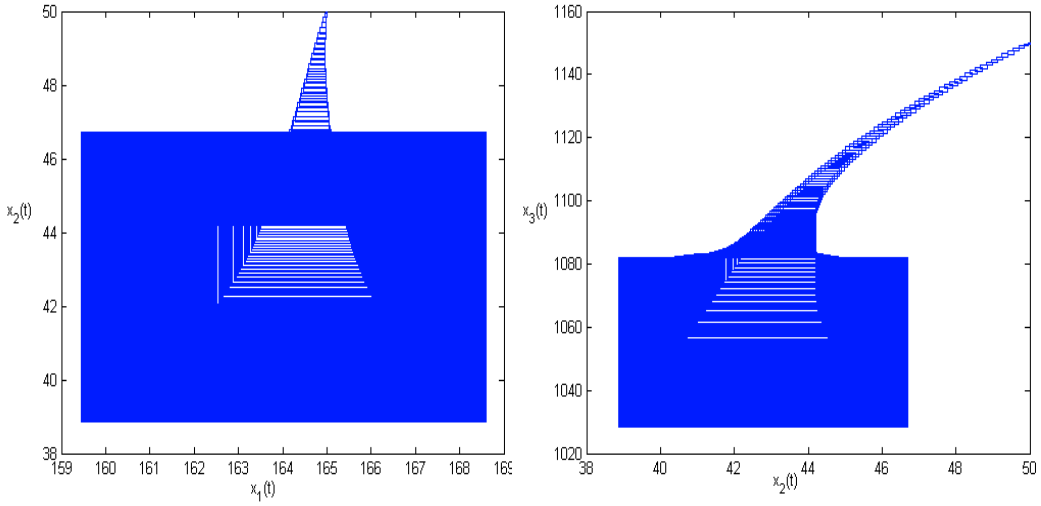


Fig. VII.13 *: Projections de l'espace atteignable de* (VII.13) *(a) sur le plan* (x_1, x_2)*, (b) sur le plan* (x_2, x_3)*. Simulation par la méthode de HOI.*

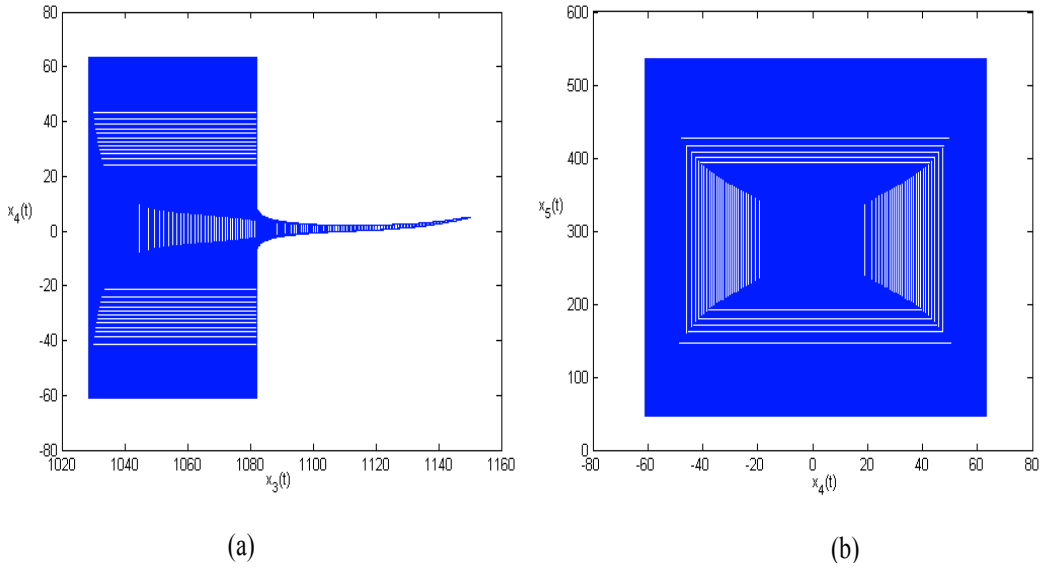


Fig. VII.14 *: Projections de l'espace atteignable de* (VII.13) *(a) sur le plan* (x_3, x_4)*, (b) sur le plan* (x_4, x_5)*. Simulation par la méthode de HOI.*

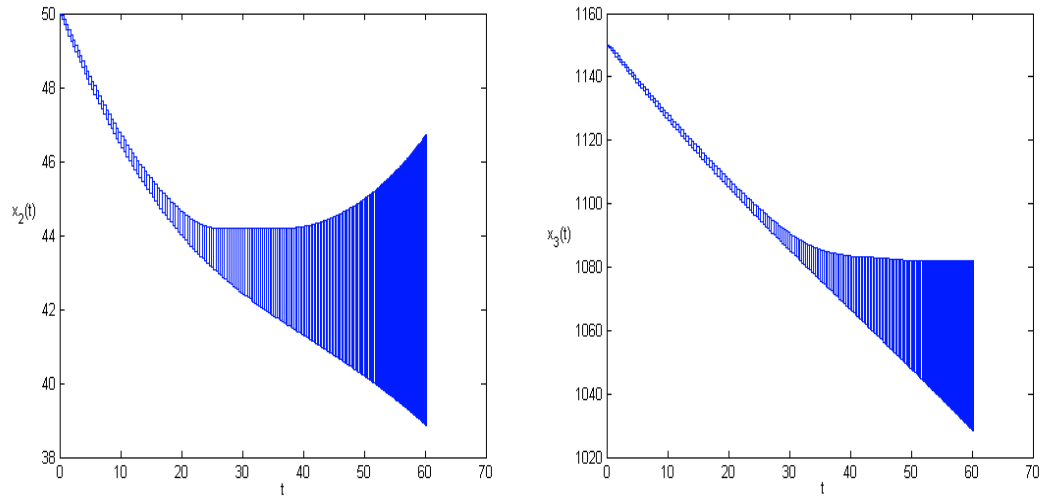


Fig. VII.15 *: L'évolution temporelle de variables d'état (a) $[x_2](t)$, (b) $[x_3](t)$. Simulation par la méthode de HOI.*

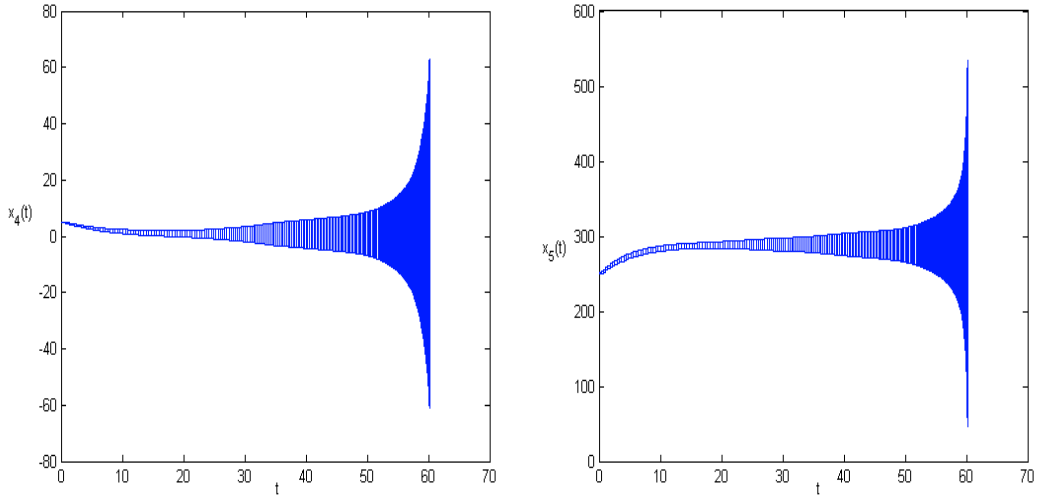


Fig. VII.16 *: L'évolution temporelle de variables d'état (a) $[x_4](t)$, (b) $[x_5](t)$. Simulation par la méthode de HOI.*

(VII.13) par rapport aux variables d'état et celles par rapport aux paramètres incertains sont de signes constants sur toute la période de simulation $[T]$. Il n'est donc pas nécessaire d'utiliser les techniques d'hybridation présentées dans le chapitre IV. Ainsi, les deux systèmes couplés sont donnés par la relation suivante :

$$\begin{cases} \dot{\underline{x}}_1 = -\frac{\overline{v}_2\underline{x}_1}{\underline{k}_2+\underline{x}_1} + \underline{v}_0\underline{u} + \underline{v}_1 \\ \dot{\underline{x}}_2 = \frac{\underline{v}_6(\underline{y}_{tot}-\underline{x}_2-\overline{x}_3)}{\overline{k}_6+(\underline{y}_{tot}-\underline{x}_2-\overline{x}_3)} - \frac{\overline{v}_3\overline{x}_1\underline{x}_2}{\underline{k}_3+\underline{x}_2} \\ \dot{\underline{x}}_3 = \frac{\underline{v}_4\underline{x}_1(\underline{y}_{tot}-\overline{x}_2-\underline{x}_3)}{\overline{k}_4+(\underline{y}_{tot}-\overline{x}_2-\underline{x}_3)} - \frac{\overline{v}_5\underline{x}_3}{\underline{k}_5+\underline{x}_3} \\ \dot{\underline{x}}_4 = \frac{\underline{v}_{10}(\underline{z}_{tot}-\underline{x}_4-\overline{x}_5)}{\overline{k}_{10}+(\underline{z}_{tot}-\underline{x}_4-\overline{x}_5)} - \frac{\overline{v}_7\overline{x}_3\underline{x}_4}{\underline{k}_7+\underline{x}_4} \\ \dot{\underline{x}}_5 = \frac{\underline{v}_8\underline{x}_3(\underline{z}_{tot}-\overline{x}_4-\underline{x}_5)}{\overline{k}_8+(\underline{z}_{tot}-\overline{x}_4-\underline{x}_5)} - \frac{\overline{v}_9\underline{x}_5}{\underline{k}_9+\underline{x}_5} \\ \dot{\overline{x}}_1 = -\frac{\underline{v}_2\overline{x}_1}{\overline{k}_2+\overline{x}_1} + \overline{v}_0\overline{u} + \overline{v}_1 \\ \dot{\overline{x}}_2 = \frac{\overline{v}_6(\overline{y}_{tot}-\overline{x}_2-\underline{x}_3)}{\underline{k}_6+(\overline{y}_{tot}-\overline{x}_2-\underline{x}_3)} - \frac{\underline{v}_3\underline{x}_1\overline{x}_2}{\overline{k}_3+\overline{x}_2} \\ \dot{\overline{x}}_3 = \frac{\overline{v}_4\overline{x}_1(\overline{y}_{tot}-\underline{x}_2-\overline{x}_3)}{\underline{k}_4+(\overline{y}_{tot}-\underline{x}_2-\overline{x}_3)} - \frac{\underline{v}_5\overline{x}_3}{\overline{k}_5+\overline{x}_3} \\ \dot{\overline{x}}_4 = \frac{\overline{v}_{10}(\overline{z}_{tot}-\overline{x}_4-\underline{x}_5)}{\underline{k}_{10}+(\overline{z}_{tot}-\overline{x}_4-\underline{x}_5)} - \frac{\underline{v}_7\underline{x}_3\overline{x}_4}{\overline{k}_7+\overline{x}_4} \\ \dot{\overline{x}}_5 = \frac{\overline{v}_8\overline{x}_3(\overline{z}_{tot}-\underline{x}_4-\overline{x}_5)}{\underline{k}_8+(\overline{z}_{tot}-\underline{x}_4-\overline{x}_5)} - \frac{\underline{v}_9\overline{x}_5}{\overline{k}_9+\overline{x}_5} \end{cases} \quad \text{(VII.14)}$$

Ainsi, la simulation de (VII.14), permet d'obtenir un encadrement garanti de l'ensemble du flot d'état généré par le système incertain (VII.13). Notons que dans ce contexte, nous avons considéré aussi l'état initial de (VII.13) est incertain $x_1 \in [160, 170]$, $x_2 \in [0, 100]$, $x_3 \in [1100, 1200]$, $x_4 \in [0, 10]$, $x_5 \in [200, 300]$.

Les courbes sur les figures VII.17 et VII.18 montrent respectivement la concaténation des solutions *a priori* des variables d'état x_2, x_3, x_4 et x_5. En effet, ces figures traduisent l'application de la formule de calcul d'approximation conservative de l'espace d'état atteignable (VII.10) au système (VII.13). Par ailleurs, les figures VII.19 et'VII.20 montrent respectivement les projections de cette approximation conservative sur les plans (x_1, x_2), (x_2, x_3), (x_3, x_4) et (x_4, x_5).

VII.5.4 Exemple 4 : Système non linéaire non coopératif 2

Pour montrer l'efficacité du calcul d'atteignabilité avec hybridation dans le cas des systèmes non linéaires non coopératifs, nous reprenons dans cet exemple le modèle de Haldane pour les bioréacteurs (VI.7), (VI.8) avec les données suivantes : $\alpha = 0.5$, $k = 42.14$, $k_s = 9.28 mmol/l$, $k_i = 256 mmol/l$, $S_{in}(t) = S^0_{in} + 15\cos(1/5t))$ et $D(t) = 2$.

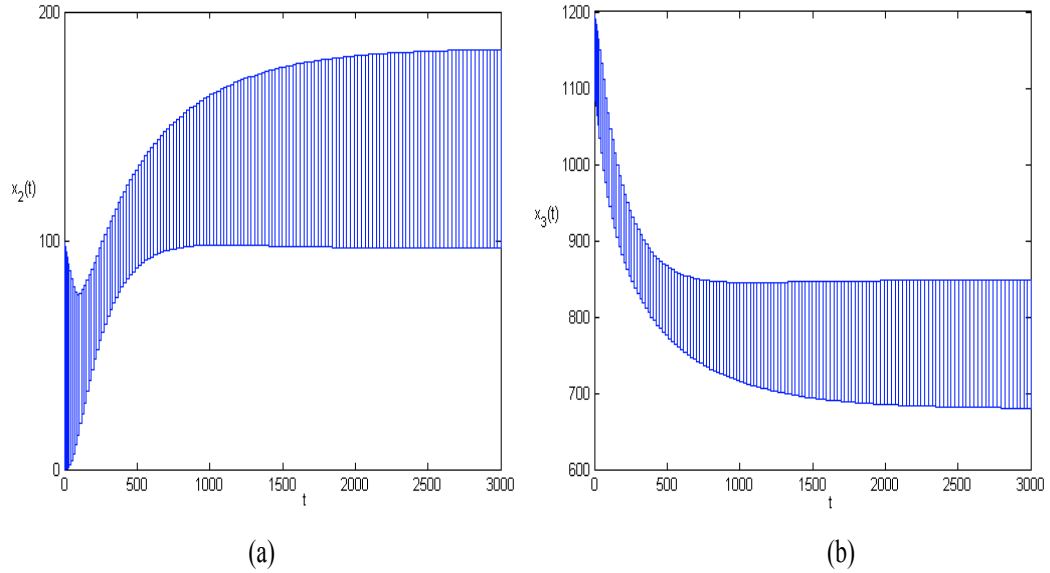


Fig. VII.17 *: L'évolution temporelle de variables d'état (a) $[x_2](t)$, (b) $[x_3](t)$. Simulation par l'utilisation de la* ***Règle 2****. Temps CPU=13s.*

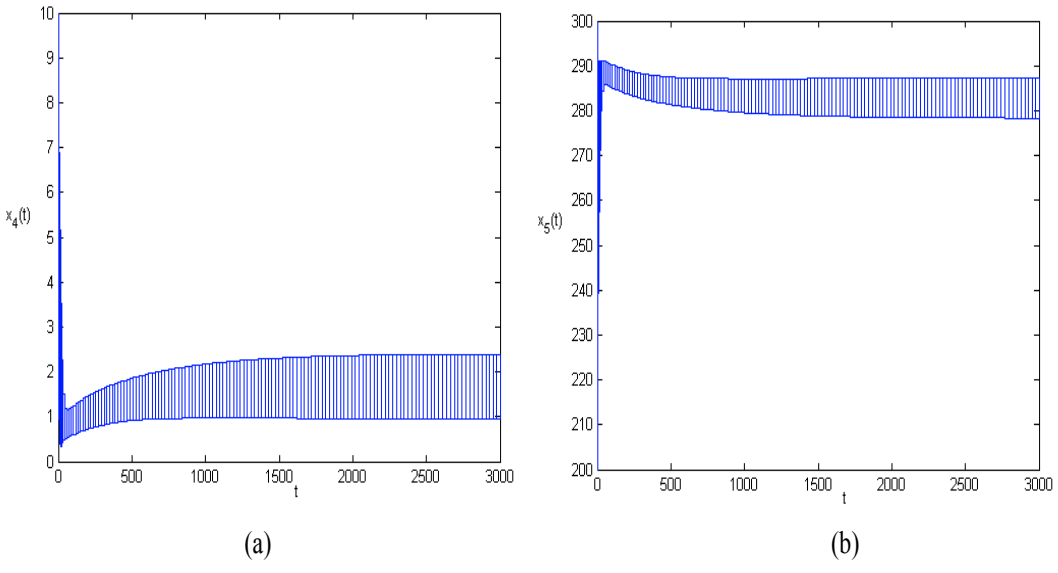


Fig. VII.18 *: L'évolution temporelle de variables d'état (a) $[x_4](t)$, (b) $[x_5](t)$. Simulation par l'utilisation de la* ***Règle 2****. Temps CPU=13s.*

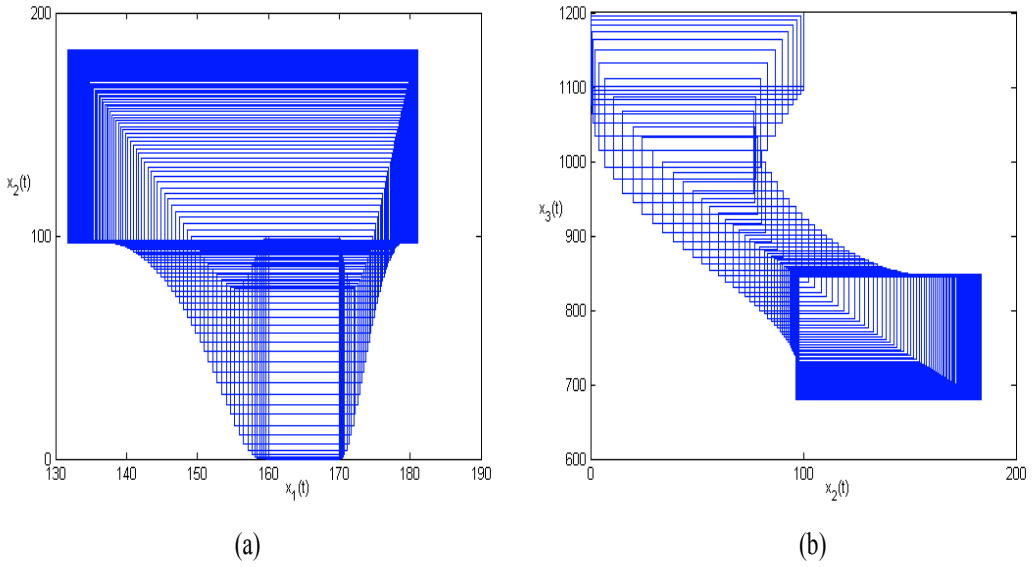


Fig. VII.19 *: Projections de l'espace atteignable de* (VII.13) *(a) sur le plan* (x_1, x_2)*, (b) sur le plan* (x_2, x_3)*. Simulation par l'utilisation de la* ***Règle 2****. Temps CPU=13s.*

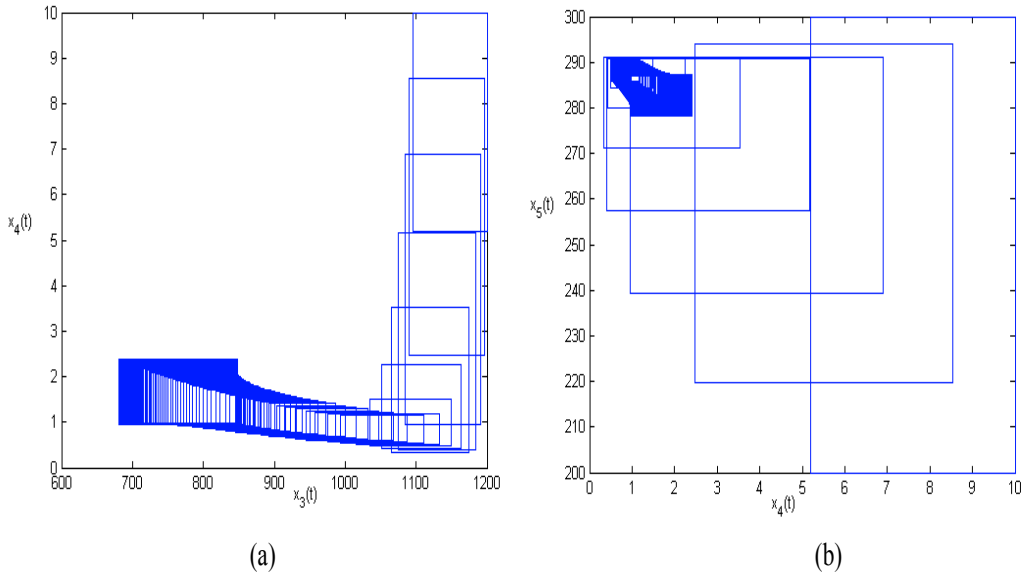


Fig. VII.20 *: Projections de l'espace atteignable de* (VII.13) *(a) sur le plan* (x_3, x_4)*, (b) sur le plan* (x_4, x_5)*. Simulation par l'utilisation de la* ***Règle 2****. Temps CPU=13s.*

Les coefficients μ_0 et S_{in}^0 sont incertains : $\mu_0 = 0.75$ avec une erreur relative de $\pm 1\%$ et $S_{in}^0 = 65$ avec une erreur relative de $\pm 1.5\%$. L'état initial est aussi incertain, dont la valeur exacte est incluse dans $X_0 \times S_0 \in [9.5, 10.5] \times [36, 44]$.

Remarque : Même pour ce système, les méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalle et le polynôme de Hermite-Obreschcoff intervalle, ne peuvent par contrôler la propagation du pessimisme. Par conséquent la taille de l'approximation conservative diverge au bout de quelques pas d'intégration et provoque l'arrêt de l'algorithme de calcul.

Donc, comme nous l'avons montré dans le chapitre VI, ce système admet trois modes d'encadrement. Les modes $q = 1$ et $q = 2$ sont donnés par les équations VI.13 et VI.14, par contre pour le mode $q = 0$ nous considérons contrairement au chapitre VI le système original (VI.7).

VII.5.4.1 Premier cas : Sans bissection

La simulation de tous les comportements possibles de ce système incertain par l'algorithme **Hybrid-Bounding** sans aucune bissection de l'espace d'état ni du domaine des paramètres permet d'obtenir une approximation conservative de l'espace d'état atteignable. Les figures VII.21 et VII.22 montrent respectivement des encadrements garantis de toutes les trajectoires possibles de la biomasse et le substrat générées par l'ensemble d'incertitude présent dans le modèle (VI.7).

VII.5.4.2 Deuxième cas : Avec bissection

Par ailleurs, nous pouvons améliorer la précision de ces encadrements en procédant à la bissection de l'espace d'état au voisinage de l'ensemble d'état défini par la condition de garde. Cet ensembles est représenté par la droite rouge dans la figure VII.22. Pour cela, pour ce deuxième cas nous simulons le système (VI.7) avec l'algorithme **Hybrid-Bounding-Partition** qui réalise des bissections de l'espace d'état afin de réduire le pessimisme dû à l'effet d'enveloppement lié à l'utilisation du mode $q = 0$ pour franchir les conditions de garde. Sur la figure VII.24 on voit bien l'amélioration de la précision apportée à l'encadrement des trajectoires de la biomasse, en particulier au voisinage de l'ensemble d'état défini par la condition de garde.

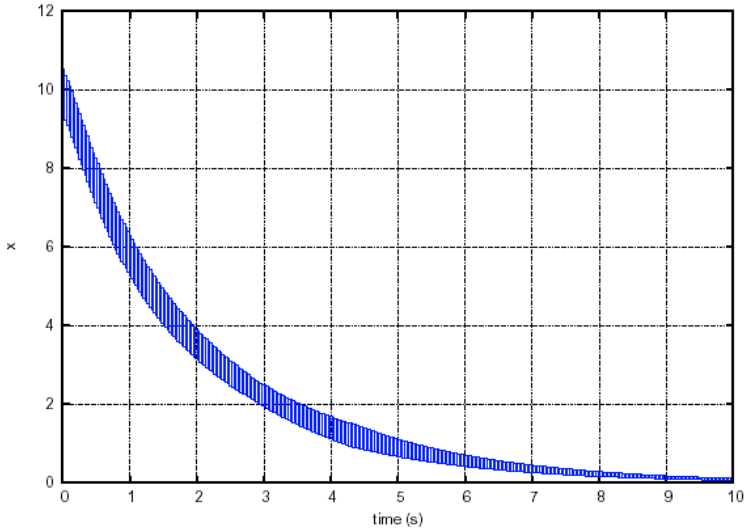


Fig. VII.21 *: Biomasse, simulation par l'algorithme* ***Hybrid-Bounding****.*

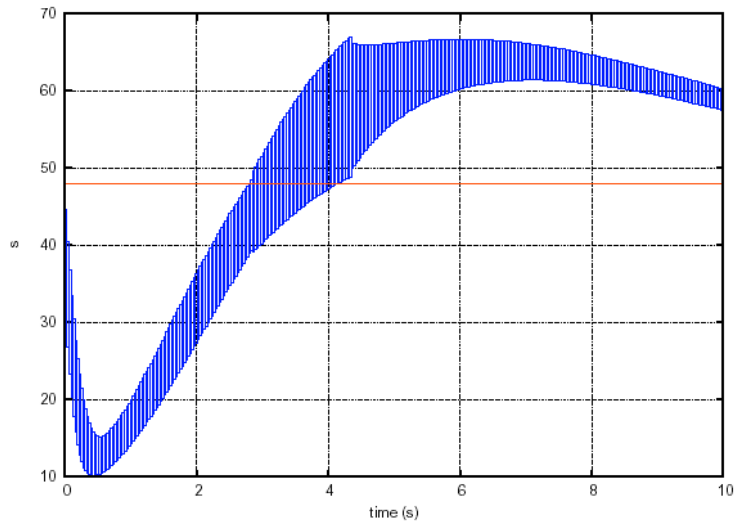


Fig. VII.22 *: Substrat, simulation par l'algorithme* ***Hybrid-Bounding****. La droite rouge parallèle à l'axe des abscisses représente la condition de garde* $S = \sqrt{k_s k_i}$.

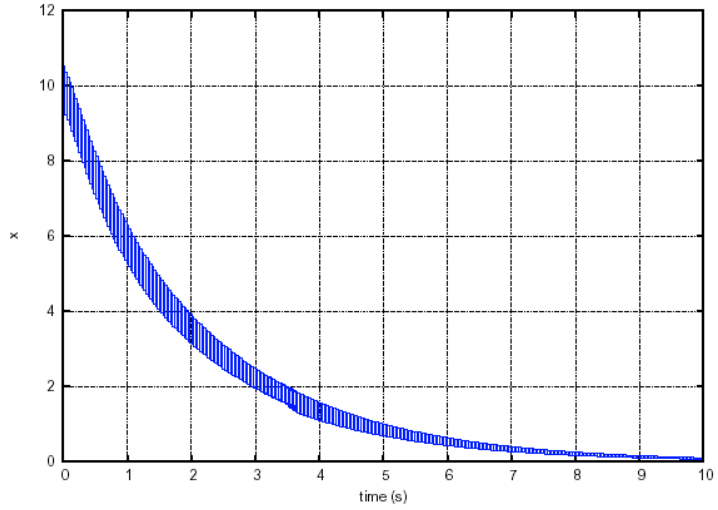


FIG. VII.23 *: Biomasse, simulation par l'algorithme* ***Hybrid-Bounding-Partition****.*

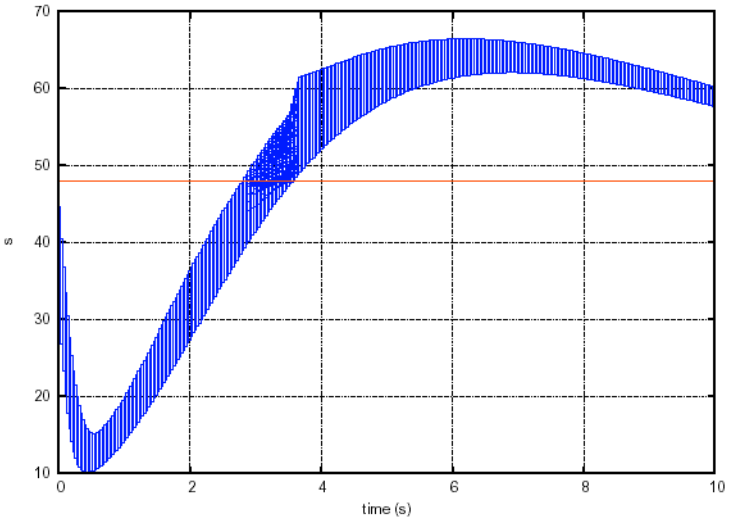


FIG. VII.24 *: Substrat, simulation par l'algorithme* ***Hybrid-Bounding-Partition****. La droite rouge parallèle à l'axe des abscisses représente la condition de garde* $S = \sqrt{k_s k_i}$.

VII.6 Conclusion

Dans ce chapitre, nous avons, en premier lieu, présenté et formulé le problème du calcul d'approximations conservatives de l'espace d'état atteignable par des systèmes dynamiques continus incertains dans le contexte de l'analyse par intervalles. La méthode proposée pour résoudre ce problème repose principalement sur les méthodes d'intégration numérique garantie des EDO introduites dans les chapitres III et IV. Plusieurs exemples illustratifs ont été donnés. Dans chaque exemple, nous avons utilisé la méthode d'intégration garantie la plus appropriée en terme de la précision de l'approximation conservative de l'espace d'état atteignable calculée. Globalement, nous avons constaté que les méthodes d'intégration garantie basées sur les théorèmes de comparaison des inégalités différentielles et une modélisation hybride des systèmes englobants fournissent des meilleurs résultats par rapport aux méthodes basées sur les modèles de Taylor intervalle ou le polynôme de Hermite-Obreschkoff intervalle, en terme de précision et temps de calcul.

Conclusion générale et perspectives

L'objectif de ce travail a été d'élargir l'application des méthodes de calcul par intervalle, en particulier l'intégration numérique garantie d'EDO incertaines, à l'estimation d'état et de paramètres des systèmes dynamiques non linéaires continus complexes, ainsi qu'au calcul de leur espace d'état atteignable. Pour cela, nous avons présenté dans le premier chapitre les principes de base de calcul par intervalle. Ensuite, dans le second chapitre, nous avons introduit les différentes méthodes d'intégration numérique garantie basées sur les modèles de Taylor intervalle et le polynôme de Hermite-Obserchkoff intervalle. Cependant, nous avons mentionné les limites de ces méthodes pour résoudre des problèmes pratiques à cause de la propagation du pessimisme associé à l'effet d'enveloppement. Dans le but de remédier à cet effet d'enveloppement, nous avons consacré le troisième chapitre aux méthodes d'intégration numérique garantie fondées sur la théorie de comparaison des inégalités différentielles. Avant de présenter le principe fondamental de ce type de méthodes d'intégration garantie d'EDO incertaines, nous avons classé les EDO incertaines en deux catégories : EDO coopératives et EDO non coopératives. Pour ces deux catégories, nous avons présenté les théorèmes et les règles permettant de construire les EDO englobantes déterministes dont les solutions encadrent de manière garantie toutes les solutions possibles de l'EDO incertaine initiale. Dans ce contexte, les règles de constructions des EDO englobantes que nous avons proposé supposent que tous les signes des dérivées partielles de l'EDO incertaine, par rapport à ses variables et par rapport à ses paramètres incertains, sont constants sur toute la période de simulation. Pour étendre l'utilisation de ces règles aux EDO pour lesquelles les signes des dérivées partielles changent au cours de l'intégration, nous avons proposé dans le quatrième chapitre une méthode hybride d'intégration garantie qui consiste à définir les EDO englobantes comme étant des systèmes dynamiques hybrides continu-discret dans lesquels les conditions de

garde sont les instants de changement de signe des dérivées partielles. Cette idée d'hybridation représente la contribution de cette thèse à l'intégration numérique garantie des EDO incertaines.

Les trois derniers chapitres de ce travail sont consacrés à l'utilisation de l'intégration numérique garantie pour la résolution globale et garantie de trois types de problèmes de l'automatique dans un contexte ensembliste. Dans le cinquième chapitre, nous avons aborder le problème de l'identification des paramètres de systèmes dynamiques continus dans le contexte à erreur-inconnue-mais-bornée de bornes connues *a priori*. Pour cela, nous avons étudié un dispositif d'identification des propriétés thermo-physiques de matériaux composites. Nous avons montré, en première lieu, les limites des méthodes d'intégration numérique garanties basées sur les modèles de Taylor intervalles et celles fondées sur le polynôme de Hermite-Obreschkoff intervalle à résoudre ce problème. En revanche, nous avons montré comment les méthodes d'intégration garantie basées sur les théorèmes de comparaison des inégalités différentielles permettent de résoudre avec succès ce problème, en particulier notre méthode hybride dans le cas d'excitation de ce dispositif avec des signaux harmoniques. Le sixième chapitre a été destiné à la construction de deux types d'observateurs intervalles pour les systèmes dynamiques continus en présence d'incertitudes. Le premier type d'observateurs est proposé pour résoudre le problème d'observation dans le cas où les mesures sont des signaux discrets et le second type pour le cas où les mesures sont des signaux continus. Pour ces deux types d'observateurs intervalles, nous avons montré l'avantage d'utiliser les méthodes d'intégration basées sur les théorèmes de comparaison des inégalités différentielles par rapport à ceux fondées sur les modèles de Taylor intervalle et le polynôme de Hermite-Obreschkoff intervalle. Enfin le septième chapitre a été dédié au calcul d'approximation conservative de l'espace d'état atteignable par des systèmes dynamiques continus incertains à partir d'un domaine d'état initial borné. Tout d'abord nous avons formulé ce problème dans le cas de l'analyse par intervalle. Ensuite, par le biais de plusieurs exemples, nous avons montré l'intérêt des théorèmes de comparaison des inégalités différentielles pour résoudre ce genre de problèmes en terme de précision de l'approximation conservative de l'espace d'état atteignable fournie et le temps de calcul nécessaire pour l'obtenir. En parallèle, nous avons montré les limites des méthodes d'intégration basées sur les modèles de Taylor intervalle et le polynôme de

Hermite-Obreschkoff.

Le travail réalisé laisse ouvertes un certain nombre de questions et ouvre la voie à de nouvelles applications. Dans cette optique, nous proposons comme suite de ce travail de thèse deux directions de recherches. La première direction porte sur *l'amélioration des performances* des méthodes d'intégration numérique garantie et *l'analyse de la convergence* de la taille d'encadrement des solutions d'EDO fourni par ces méthodes :

1. Pour les méthodes d'intégration basées sur le développement en série de Taylor intervalle ou celles basées sur le polynôme de Hermite-Obreschkoff, nous suggérons de faire des choix particuliers et alternés des matrices de pré-conditionnement bien adaptées pour chaque EDO traitée. Par exemple, sur les intervalles de temps où l'EDO est coopérative on utilise des matrices identité, et sur les intervalles de temps où cette ODE est non coopérative on utilise la factorisation **QR** ;

2. Dans le cas de la théorie de comparaison des inégalités différentielles, nous souhaitons trouver des transformations géométriques réversibles des coordonnées de variables de l'EDO de la sorte que la nouvelle EDO soit coopérative dans la nouvelle base de coordonnées de variables ;

3. L'analyse de la stabilité des EDO englobantes par l'approche de Lyapunov même dans le cas où les EDO englobantes sont des systèmes dynamiques hybrides continu-discret ;

4. Quantifier le pessimisme introduit par le fait que dans les systèmes englobant on trouve simultanément les bornes supérieure et inférieure du même paramètre, ce qui est physiquement irréalisable ...

La seconde direction de recherche consiste à *élargir le champ d'application* de l'étude d'atteignabilité à base de calcul par intervalle et l'intégration numérique garantie d'EDO pour d'autres problèmes de l'automatique continue :

1. Construction des lois de commande robustes. Par exemple, trouver l'ensemble de commandes qui permettent de faire évoluer un système dynamique incertain d'un domaine d'état initial vers un domaine d'état final défini selon un certain cahier des charges quelle que soit l'incertitude sur le modèle de ce système. Cette évolution peut se faire en une seule étape ou en plusieurs étapes ;

2. Stabilité et stabilisation, c'est-à-dire de caractériser le domaine (état et paramètre) d'attraction d'un système dynamique incertain au tour de ses points d'équilibre, de caractériser la marge de gain dans le cas des systèmes linéaires en boucle fermée qui assure la stabilité ;
3. Identification ensembliste des systèmes décrits pas de équations différentielles à dérivées partielles (EDP). Dans ce contexte, il est intéressant de trouver une méthode de discrétisation spatiale garantie des EDP afin de pouvoir appliquer les méthodes d'identification ensembliste développées pour les EDO ;
4. Etendre les méthodes de calcul d'approximations conservatives de l'espace d'état atteignable par les systèmes dynamiques continus au cas des systèmes dynamiques hybrides SDH. Pour ce contexte, il faut trouver des méthodes garanties pour le franchissement des condition de gardes ...

Annexes

Annexe A

Méthode de Hermite-Obreschkoff intervalle

A.1 Polynôme de Hermite-Obreschkoff : Cas ponctuel

Le polynôme ou le développement de Hermite-Obreschkoff est conçu à partir de l'intégrale définie ci-dessous,

$$(-1)^{(p+q)} \int_0^1 P_{p,q}(s)\mathbf{g}^{p+q+1}(s)ds \tag{A.1}$$

où

$$\begin{aligned} P_{p,q}(s) &= \tfrac{s^q(s-1)^p}{(p+q)!}, \\ c_i^{q,p} &= \tfrac{q!(q+p-i)!}{(p+q)!(q-i)!}, \\ \mathbf{g}_i(s) &= \tfrac{\mathbf{g}^{(i)}(s)}{i!}, \end{aligned} \tag{A.2}$$

avec $p \geq 0,\ q \geq 0,\ 0 \leq i \leq q$ et $\mathbf{g}(t)$ représente toute fonction $(p+q+1)$ différentiable.

D'une part, en intégrant successivement par partie la formule (A.1), on obtient,

$$(-1)^{(p+q)} \int_0^1 P_{p,q}(s)\mathbf{g}^{p+q+1}(s)ds = \sum_{i=0}^{q}(-1)^i c_i^{q,p}\mathbf{g}_i(1) - \sum_{i=0}^{p} c_i^{p,q}\mathbf{g}_i(0), \tag{A.3}$$

par la suite, si on pose $\mathbf{g}(s) = \mathbf{x}(t_j + sh_j)$, tel que $\mathbf{x}(t_j)$ et la solution de $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x})$ à l'instant t_j. Alors

$$\begin{aligned} \mathbf{g}^{p+q+1}(s) &= h_j^{p+q+1}\mathbf{x}^{p+q+1}(t_j + sh_j), \\ \mathbf{g}_i(0) &= \tfrac{\mathbf{g}^{(i)}(0)}{i!} = h_j^i \tfrac{\mathbf{y}^{(i)}(t_j)}{i!} = h_j^i \mathbf{f}^{[i]}(\mathbf{x}_j), \\ \mathbf{g}_i(1) &= \tfrac{\mathbf{g}^{(i)}(1)}{i!} = h_j^i \tfrac{\mathbf{y}^{(i)}(t_j+h_j)}{i!} = h_j^i \mathbf{f}^{[i]}(\mathbf{x}_{j+1}), \end{aligned} \tag{A.4}$$

où $\mathbf{x}_{j+1} = \mathbf{x}(t_j + h_j)$ et les coefficients $\mathbf{f}^{[i]}$ sont définis dans (II.4). Ainsi, l'équation (A.3) devient,

$$(-1)^{(p+q)} \int_0^1 P_{p,q}(s)\mathbf{g}^{p+q+1}(s)ds = \sum_{i=0}^{q}(-1)^i c_i^{q,p} h_j^i \mathbf{f}^{[i]}(\mathbf{x}_{j+1}) - \sum_{i=0}^{p} c_i^{p,q} h_j^i \mathbf{f}^{[i]}(\mathbf{x}_j). \quad (A.5)$$

Et d'autre part, si on applique le théorème de la valeur moyenne au (A.1), on obient

$$\begin{aligned} &(-1)^{p+q} \textstyle\int_0^1 P_{p,q}(s)\mathbf{g}^{(p+q+1)}(s)ds \\ &\qquad = (-1)^{p+q} h_j^{p+q+1} \textstyle\int_0^1 P_{p,q}(s)\mathbf{x}^{(p+q+1)}(t_j + sh_j)ds \\ &\qquad = (-1)^q \tfrac{q!p!}{(p+q)!} h_j^{p+q+1} \tfrac{\mathbf{x}^{(p+q+1)}(t;t_j,t_{j+1})}{(p+q+1)!}, \\ &\qquad = (-1)^q \tfrac{q!p!}{(p+q)!} h_j^{p+q+1} \mathbf{f}^{[p+q+1]}(\mathbf{x}(\tau)), \end{aligned} \quad (A.6)$$

où le vecteur $\mathbf{x}(\tau) \in [\tilde{\mathbf{x}}_j]$.

Enfin, de (A.3) et (A.5) on obtient le polynôme de Hermite-Obreschkoff suivant,

$$\sum_{i=0}^{q}(-1)^i c_i^{q,p} h_j^i \mathbf{f}^{[i]}(\mathbf{x}_{j+1}) = \sum_{i=0}^{p} c_i^{p,q} h_j^i \mathbf{f}^{[i]}(\mathbf{x}_j) + (-1)^q \frac{q!p!}{(p+q)!} h_j^{p+q+1} \mathbf{f}^{[p+q+1]}(\mathbf{x}(\tau)). \quad (A.7)$$

Ainsi, pour $\mathbf{x}_j$ donnée, si nous résolvons les système d'équations non linéaires (cas général),

$$\sum_{i=0}^{q}(-1)^i c_i^{q,p} h_j^i \mathbf{f}^{[i]}(\mathbf{x}_{j+1}) = \sum_{i=0}^{p} c_i^{p,q} h_j^i \mathbf{f}^{[i]}(\mathbf{x}_j) \quad (A.8)$$

nous obtiendrons une approximation de la solution $\mathbf{x}_{j+1}$ du problème $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x})$ à l'instant t_{j+1}, avec $\mathbf{x}_j$ comme condition initiale. De plus, il est possible de borner localement l'erreur de troncature, si nous disposions d'une solution *a priori* $[\tilde{\mathbf{x}}_j]$. **Remarques** :

1. Si $p > 0$ et $q = 0$, alors nous obtiendrons le développement en série de Taylor explicite défini par (II.12).
2. Si $p = 0$ et $q > 0$, alors nous obtiendrons le schéma d'intégration implicite de Taylor défini par (II.36).

En conséquent, nous considérons le polynôme de Hermite-Obreschkoff comme une généralisation du développement en série de Taylor.

A.2 Extension aux intervalles

Comme nous venons de le constater, dans la formule (A.7), pour réaliser un pas d'intégration d'un système d'équations différentielles à l'aide du polynôme de Hermite-Obreshkoff, nous avons besoin que des coefficients du développement de Taylor du seconde

membre du système d'équations différentielles en question. De plus, il est facile de borner d'une manière garantie, l'erreur de troncature locale, puisque nous savons comment générer les coefficients de Taylor intervalles, en particulier le dernier terme $\mathbf{f}^{[p+q+1]}([\tilde{\mathbf{x}}_j])$ qui encadre extérieurement cette erreur.

En conclusion, il est intéressant d'étendre la méthode de Hermite-Obreshkoff aux cas intervalles. L'extension de cette méthode proposée Nedialkov [69, 70] est organisée en deux phases. La première, dite phase de *prédiction* permet de calculer un encadrement extérieur de la solution $[\mathbf{x}_{j+1}]_p$ en utilisant un développement en série de Taylor. La seconde, dite phase de *correction* consiste à réduire $[\mathbf{x}_{j+1}]_p$ avec un contracteur de type Newton tel que $[\mathbf{x}_{j+1}]_c \subseteq [\mathbf{x}_{j+1}]_p$.

Remarque : Il faut pas confondre entre les étapes *prédiction-correction* qui constituent le principe fondamental des méthodes d'intégration numérique garantie, et ceux de la méthode de Hermite-Obreshkoff étendue aux intervalles.

A.2.1 Phase de prédiction

A ce jour, l'idée cruciale pour limiter le pessimisme dû à l'effet d'enveloppement dans le cadre général de l'intégration numérique garantie à base de la représentation et l'arithmétique d'intervalles, consiste à faire propager l'encadrement de l'ensemble solution sous forme de parallélépipède $\mathbf{x}(t_j; t_0, [\mathbf{x}_0]) \subseteq \{\hat{\mathbf{x}}_j + \mathbf{A}_j\mathbf{r}_j \mid \mathbf{r}_j \in [\mathbf{r}_j]\}$ au lieu de vecteur intervalles $[\mathbf{x}_j]$. Compte tenu du fait que, l'inclusion ci dessous est toujours vraie

$$\forall j \geq 0 \;\; \{\hat{\mathbf{x}}_j + \mathbf{A}_j\mathbf{r}_j \mid \mathbf{r}_j \in [\mathbf{r}_j]\} \subseteq [\mathbf{x}_j].$$

où $\hat{\mathbf{x}}_j = m([\mathbf{x}_j])$, $\mathbf{A}_j \in \mathbb{R}^{n\times n}$ matrice non-singulière et $[\mathbf{r}_j] \in \mathbb{IR}^n$.

En effet, cette phase de prédiction est réalisée par l'utilisation du un algorithme basé sur la méthode de Lohner II.3.2.4 avec un développement en série de Taylor d'ordre $q+1$. Par contre, la préparation du prochain pas d'intégration se fait dans la phase de correction, c'est-à-dire le calcul de $\mathbf{A}_{j+1}$ et de $[\mathbf{r}_{j+1}]$. Soient h_j et $[\tilde{\mathbf{x}}_j]$, respectivement le futur pas d'intégration et la solution *a priori* sur l'intervalle $[t_j, t_{j+1}]$ calculés par **Algorithme I** et soient $\mathbf{A}_j$, $[\mathbf{r}_j]$ et $[\mathbf{x}_j]$ les données obtenues à l'instant t_j, alors l'algorithme qui décrit la phase de prédiction est le suivant,

AlgorithmeII : (Phase de prédiction)

1. Entrées : $q,\ [\tilde{\mathbf{x}}_j], h_j, [\mathbf{x}_j], \hat{\mathbf{x}}_j, \mathbf{A}_j, [\mathbf{r}_j]$;
2. $[\mathbf{z}_{j+1}] := h_j^{q+1}\mathbf{f}^{[q+1]}([\tilde{\mathbf{x}}_j])$;
3. $\mathbf{f}_{j,i}^{+} := h_j^i\mathbf{f}^{[i]}(\hat{\mathbf{x}}_j)$, pour $i = 1, ..., q$;
4. $[\mathbf{u}_{j+1}] := \hat{\mathbf{x}}_j + \sum_{i=1}^{q}\mathbf{f}_{j,i}^{+} + [\mathbf{z}_{j+1}]$;
5. $[\mathbf{F}_{j,i}^{+}] := h_j^i\mathbf{J}(\mathbf{f}^{[i]}, [\mathbf{x}_j])$, pour $i = 1, ..., q$;
6. $[\mathbf{S}_j] := I + \sum_{i=1}^{q}[\mathbf{F}_{j,i}^{+}]$;
7. $[\mathbf{x}_{j+1}]_p := [\mathbf{u}_{j+1}] + ([\mathbf{S}_j]\mathbf{A}_j)[\mathbf{r}_j]$;
8. Sorties :

   $[\mathbf{x}_{j+1}]_p$;

   $\mathbf{f}_{j,i}^{+},\ [\mathbf{F}_{j,i}^{+}]$, pour $i = 1, ..., q$.

Remarque : Il est possible d'utiliser directement, dans la phase de correction, $[\tilde{\mathbf{x}}_j]$ au lieu de calculer $[\mathbf{x}_{j+1}]_p$. Par contre, étant donné que la taille de $[\tilde{\mathbf{x}}_j]$ est importante, nous serions contraint d'itérer plusieurs fois l'algorithme de la phase de correction afin d'améliorer la précision de l'encadrement de solutions. Ce qui augmentera considérablement le temps de calcul.

A.2.2 Phase de correction

L'objectif de cette phase consiste à corriger ou à contracter $[\mathbf{x}_{j+1}]_p \supseteq [\mathbf{x}_{j+1}]_c$ et de préparer le prochain pas d'intégration en calculant $[\mathbf{x}_{j+1}]$, $\hat{\mathbf{x}}_{j+1}$, $\mathbf{A}_{j+1}$ et $[\mathbf{r}_{j+1}]$.

Soient

$$\mathbf{x}_j = \mathbf{x}(t_j; t_0, \mathbf{x}_0) \ \text{ et } \ \mathbf{x}_{j+1} = \mathbf{x}(t_{j+1}; t_0, \mathbf{x}_0)$$

pour un certain $\mathbf{x}_0 = [\mathbf{x}_0]$, et on pose $\hat{\mathbf{x}}_{j+1,p} = m([\mathbf{x}_{j+1}]_p)$. Par ailleurs, puisque

$$\mathbf{x}_{j+1},\ \hat{\mathbf{x}}_{j+1,p} \in [\mathbf{x}_{j+1}]_p \ \text{ et } \ \mathbf{x}_j,\ \hat{\mathbf{x}}_j \in [\mathbf{x}_j],$$

En appliquant le théorème de la valeur moyenne pour les différents coefficients du développement de Taylor de l'équation (A.7) nous obtiendrons

$$\begin{aligned}
&\textstyle\sum_{i=0}^{q}(-1)^i c_i^{q,p} h_j^i \mathbf{f}^{[i]}(\hat{\mathbf{x}}_{j+1,p}) + \big(\sum_{i=0}^{q}(-1)^i c_i^{q,p} h_j^i \mathbf{J}(\mathbf{f}^{[i]}; \mathbf{x}_{j+1}, \hat{\mathbf{x}}_{j+1,p})\big)(\mathbf{x}_{j+1} - \hat{\mathbf{x}}_{j+1,p}) \\
&\quad = \textstyle\sum_{i=0}^{p} c_i^{p,q} h_j^i \mathbf{f}^{[i]}(\hat{\mathbf{x}}_j) + \big(\sum_{i=0}^{p} c_i^{p,q} h_j^i \mathbf{J}(\mathbf{f}^{[i]}; \mathbf{x}_j, \hat{\mathbf{x}}_j)\big)(\mathbf{x}_j - \hat{\mathbf{x}}_j) \\
&\qquad + (-1)^q \tfrac{q!p!}{(p+q)!} h_j^{p+q+1} \mathbf{f}^{[p+q+1]}(\mathbf{x}(\tau)).
\end{aligned} \tag{A.9}$$

où $\mathbf{J}(\mathbf{f}^{[i]};\mathbf{x}_{j+1},\hat{\mathbf{x}}_{j+1,p})$ est le Jacobien de $\mathbf{f}^{[i]}$ dont les l^{ieme} ligne sont évaluées aux points $\mathbf{x}_{j+1}+\theta_{il}(\hat{\mathbf{x}}_{j+1,p}-\mathbf{x}_{j+1})$ pour un certain $\theta_{il}\in[0,1]$, et $\mathbf{J}(\mathbf{f}^{[i]};\mathbf{x}_j,\hat{\mathbf{x}}_j)$ est le Jacobien de $\mathbf{f}^{[i]}$ dont les l^{ieme} ligne sont évaluées aux points $\mathbf{x}_j+\eta_{il}(\hat{\mathbf{x}}_j-\mathbf{x}_j)$ pour un certain $\eta_{il}\in[0,1]$, $l=1,...,n$.

Ainsi, la résolution de (A.9) permet de contracter $[\mathbf{x}_{j+1}]_p$ et de calculer le pavé $[\mathbf{x}_{j+1}]_c\subseteq[\mathbf{x}_{j+1}]_p$. Pour montrer comment résoudre ce problème, nous considérons les changements des variables suivants,

$$\begin{array}{lcl}
\mathbf{S}_{j+1,-} &=& \sum_{i=0}^{q}(-1)^i c_i^{q,p}h_j^i\mathbf{J}(\mathbf{f}^{[i]};\mathbf{x}_{j+1},\hat{\mathbf{x}}_{j+1,p})\\
&\in& \sum_{i=0}^{q}(-1)^i c_i^{q,p}h_j^i\mathbf{J}(\mathbf{f}^{[i]};[\mathbf{x}_{j+1}]_p)\equiv[\mathbf{S}_{j+1,-}],\\
\hat{\mathbf{S}}_{j+1,-} &=& m([\mathbf{S}_{j+1,-}]),\\
\mathbf{S}_{j,+} &=& \sum_{i=0}^{p}(-1)^i c_i^{p,q}h_j^i\mathbf{J}(\mathbf{f}^{[i]};\mathbf{x}_j,\hat{\mathbf{x}}_j)\\
&\in& \sum_{i=0}^{p}(-1)^i c_i^{p,q}h_j^i\mathbf{J}(\mathbf{f}^{[i]};[\mathbf{x}_j])\equiv[\mathbf{S}_{j,+}],\\
[\mathbf{B}_j] &=& (\hat{\mathbf{S}}_{j+1,-}^{-1}[\mathbf{S}_{j,+}])\mathbf{A}_j,\\
[\mathbf{C}_j] &=& \mathbf{I}-\hat{\mathbf{S}}_{j+1,-}^{-1}[\mathbf{S}_{j+1,-}],\\
[\mathbf{v}_j] &=& [\mathbf{x}_{j+1}]_p-\hat{\mathbf{x}}_{j+1,p},\\
\epsilon_{j+1} &=& (-1)^q\frac{q!p!}{(p+q)!}h_j^{p+q+1}\mathbf{f}^{[p+q+1]}(\mathbf{x}(\tau)),\\
&\in& (-1)^q\frac{q!p!}{(p+q)!}h_j^{p+q+1}\mathbf{f}^{[p+q+1]}([\tilde{\mathbf{x}}_j])\equiv[\epsilon_{j+1}],\\
\mathbf{g}_{j+1} &=& \sum_{i=0}^{p}c_i^{p,q}h_j^i\mathbf{f}^{[i]}(\hat{\mathbf{x}}_j)-\sum_{i=0}^{q}(-1)^i c_i^{q,p}h_j^i\mathbf{f}^{[i]}(\hat{\mathbf{x}}_{j+1,p}),\quad\text{et}\\
\delta_{j+1} &=& \mathbf{g}_{j+1}+\epsilon_{j+1}\in\mathbf{g}_{j+1}+[\epsilon_{j+1}]\equiv[\delta_{j+1}].
\end{array}\tag{A.10}$$

D'où, avec ces nouvelles notations (A.10), nous récrivons l'équation (A.9) sous cette forme,

$$\hat{\mathbf{S}}_{j+1,-}(\mathbf{x}_{j+1}-\hat{\mathbf{x}}_{j+1,p})=\mathbf{S}_{j,+}(\mathbf{x}_j-\hat{\mathbf{x}}_j)+\delta_{j+1}-(\mathbf{S}_{j+1,-}-\hat{\mathbf{S}}_{j+1,-})(\mathbf{x}_{j+1}-\hat{\mathbf{x}}_{j+1,p}).\tag{A.11}$$

De plus, puisque

$$\mathbf{x}_j-\hat{\mathbf{x}}_j\in\{\mathbf{A}_j\mathbf{r}_j\mid\mathbf{r}_j\in[\mathbf{r}_j]\},$$

alors il existe $\mathbf{r}_j\in[\mathbf{r}_j]$ tel que $\mathbf{x}_j-\hat{\mathbf{x}}_j=\mathbf{A}_j\mathbf{r}_j$. Par la suite nous transformons (A.11) en,

$$\hat{\mathbf{S}}_{j+1,-}(\mathbf{x}_{j+1}-\hat{\mathbf{x}}_{j+1,p})=(\mathbf{S}_{j,+}\mathbf{A}_j)\mathbf{r}_j+\delta_{j+1}-(\mathbf{S}_{j+1,-}-\hat{\mathbf{S}}_{j+1,-})(\mathbf{x}_{j+1}-\hat{\mathbf{x}}_{j+1,p}).\tag{A.12}$$

En pratique, pour des pas h_j assez petits, il est possible d'inverser la matrice $\hat{\mathbf{S}}_{j+1,-}$. Dès

lors, de (A.12) et les notations du (A.10) on obtient,

$$\begin{aligned}
\mathbf{x}_{j+1} - \hat{\mathbf{x}}_{j+1,p} &= \big((\hat{\mathbf{S}}^{-1}_{j+1,-}\mathbf{S}_{j,+})\mathbf{A}_j\big)\mathbf{r}_j + \hat{\mathbf{S}}^{-1}_{j+1,-}\delta_{j+1} + (\mathbf{I} - \hat{\mathbf{S}}^{-1}_{j+1,-}\mathbf{S}_{j+1,-})(\mathbf{x}_{j+1} - \hat{\mathbf{x}}_{j+1,p}) \\
&\in \big((\hat{\mathbf{S}}^{-1}_{j+1,-}[\mathbf{S}_{j,+}])\mathbf{A}_j\big)[\mathbf{r}_j] + \hat{\mathbf{S}}^{-1}_{j+1,-}[\delta_{j+1}] \\
&\quad +(\mathbf{I} - \hat{\mathbf{S}}^{-1}_{j+1,-}[\mathbf{S}_{j+1,-}])([\mathbf{x}_{j+1}]_p - \hat{\mathbf{x}}_{j+1,p}) \\
&= [\mathbf{B}_j][\mathbf{r}_j] + [\mathbf{C}_j][\mathbf{v}_j] + \hat{\mathbf{S}}^{-1}_{j+1,-}[\delta_{j+1}]
\end{aligned} \tag{A.13}$$

Ainsi, étant donné que l'appartenance ci-dessous,

$$\mathbf{x}_{j+1} = \mathbf{x}(t_{j+1}; t_0, \mathbf{x}_0) \in \hat{\mathbf{x}}_{j+1,p} + [\mathbf{B}_j][\mathbf{r}_j] + [\mathbf{C}_j][\mathbf{v}_j] + \hat{\mathbf{S}}^{-1}_{j+1,-}[\delta_{j+1}]$$

et toujours vraie pour tout $\mathbf{x}_0 \in [\mathbf{x}_0]$, alors

$$\mathbf{x}(t_{j+1}; t_0, [\mathbf{x}_0]) \in \hat{\mathbf{x}}_{j+1,p} + [\mathbf{B}_j][\mathbf{r}_j] + [\mathbf{C}_j][\mathbf{v}_j] + \hat{\mathbf{S}}^{-1}_{j+1,-}[\delta_{j+1}].$$

Enfin, le vecteur intervalles qui contient toutes les solutions possibles du (II.1) à l'instant t_{j+1} est calculé par la relation suivante

$$[\mathbf{x}_{j+1}] = [\mathbf{x}_{j+1}]_c = \big(\hat{\mathbf{x}}_{j+1,p} + [\mathbf{B}_j][\mathbf{r}_j] + [\mathbf{C}_j][\mathbf{v}_j] + \hat{\mathbf{S}}^{-1}_{j+1,-}[\delta_{j+1}]\big) \cap [\mathbf{x}_{j+1}]_p, \tag{A.14}$$

et nous propageons pour le prochain pas d'intégration

$$\hat{\mathbf{x}}_{j+1} = m([\mathbf{x}_{j+1}]), \tag{A.15}$$

$\mathbf{A}_{j+1}$ qui est la matrice $\mathbf{Q}$ de la factorisation $\mathbf{QR}$ de $m([\mathbf{B}_j])$ et le vecteur intervalle

$$[\mathbf{r}_{j+1}] = (\mathbf{A}^{-1}_{j+1}[\mathbf{B}_j])[\mathbf{r}_j] + (\mathbf{A}^{-1}_{j+1}[\mathbf{C}_j])[\mathbf{v}_j] + (\mathbf{A}^{-1}_{j+1}\hat{\mathbf{S}}^{-1}_{j+1,-})[\delta_{j+1}] + \mathbf{A}^{-1}_{j+1}(\hat{\mathbf{x}}_{j+1,p} - \hat{\mathbf{x}}_{j+1}). \tag{A.16}$$

Pour résumer cette phase de correction de la méthode de Hermite-Obreschkoff, nous reprenons l'algorithme présenté par Nedialkov dans sa thèse [69].

AlgorithmeII : (Méthode de la valeur moyenne étendue)

1. Entrées :

   p, q, $c_i^{p,q}$ pour $i = 0, ..., p$, $c_i^{q,p}$ pour $i = 0, ..., q$, ;

   h_j, $\hat{\mathbf{x}}_j$, $\mathbf{A}_j$, $[\mathbf{r}_j]$, $[\tilde{\mathbf{x}}_j]$, $[\mathbf{x}_{j+1}]_p$, ;

   $\mathbf{f}_{j,i}$, $[\mathbf{F}_{j,i}]$, pour $i = 0, ..., q$, ;

2. $[\mathbf{z}_{j+1}] := h_j^{p+q+1}\mathbf{f}^{[p+q+1]}([\tilde{\mathbf{x}}_j])$;

3. $\hat{\mathbf{x}}_{j+1,p} = m([\mathbf{x}_{j+1}]_p)$;

4. $\mathbf{f}^-_{j+1,i} = h^i_j \mathbf{f}^{[i]}(\hat{\mathbf{x}}_{j+1,p})$, pour $i = 1, ..., q$;
5. $\mathbf{g}_{j+1} = \hat{\mathbf{x}}_j - \hat{\mathbf{x}}_{j+1,p} + \sum_{i=1}^{p} c_i^{p,q} \mathbf{f}_{j,i} - \sum_{i=1}^{q} (-1)^i c_i^{q,p} \mathbf{f}^-_{j+1,i}$;
6. $\gamma = \frac{(-1)^q !p!}{(p+q)!}$;
7. $[\delta_{j+1}] = \mathbf{g}_{j+1} + \gamma[\mathbf{z}_{j+1}]$; item $[\mathbf{S}_{j,+}] = \mathbf{I} + \sum_{i=1}^{p} c_i^{p,q} [\mathbf{F}_{j,i}])$;
8. $[\mathbf{F}^-_{j+1,i}] = h^i_j \mathbf{J}(\mathbf{f}^{[i]}; [\mathbf{x}_{j+1}]_p)$, pour $i = 1, ..., q$;
9. $[\mathbf{S}_{j+1,-}] = \mathbf{I} + \sum_{i=1}^{q} (-1)^i c_i^{q,p} [\mathbf{F}^-_{j+1,i}])$;
10. $\hat{\mathbf{S}}_{j+1,-} = m([\mathbf{S}_{j+1,-}])$;
11. $[\mathbf{B}_j] = (\hat{\mathbf{S}}^{-1}_{j+1,-}[\mathbf{S}_{j,+}])\mathbf{A}_j$;
12. $[\mathbf{C}_j] = \mathbf{I} - \hat{\mathbf{S}}^{-1}_{j+1,-}[\mathbf{S}_{j+1,-}]$;
13. $[\mathbf{v}_j] = [\mathbf{x}_{j+1}]_p - \hat{\mathbf{x}}_{j+1,p}$;
14. $[\mathbf{x}_{j+1}]_c = \left(\hat{\mathbf{x}}_{j+1,p} + [\mathbf{B}_j][\mathbf{r}_j] + [\mathbf{C}_j][\mathbf{v}_j] + \hat{\mathbf{S}}^{-1}_{j+1,-}[\delta_{j+1}]\right) \cap [\mathbf{x}_{j+1}]_p$;
15. $[\mathbf{x}_{j+1}] = [\mathbf{x}_{j+1}]_c$;
16. $\hat{\mathbf{A}}_{j+1} = m([\mathbf{B}_j])$;
17. $\mathbf{A}_{j+1} = \mathbf{Q}$, la matrice orthogonale de la factorisation $\mathbf{QR}$ de $\hat{\mathbf{A}}_{j+1}$;
18. $\hat{\mathbf{x}}_{j+1} = m([\mathbf{x}_{j+1}])$;
19. $[\mathbf{r}_{j+1}] = (\mathbf{A}^{-1}_{j+1}[\mathbf{B}_j])[\mathbf{r}_j] + (\mathbf{A}^{-1}_{j+1}[\mathbf{C}_j])[\mathbf{v}_j] + (\mathbf{A}^{-1}_{j+1}\hat{\mathbf{S}}^{-1}_{j+1,-})[\delta_{j+1}] + \mathbf{A}^{-1}_{j+1}(\hat{\mathbf{x}}_{j+1,p} - \hat{\mathbf{x}}_{j+1})$.
20. Sorties :

    $\hat{\mathbf{x}}_{j+1}$, $\mathbf{A}_{j+1}$, $[\mathbf{r}_{j+1}]$, $[\mathbf{x}_{j+1}]$.

Les coefficients $c_i^{q,p}$ peuvent être calculer récursivement par la formule,

$$\begin{aligned} c_0^{q,p} &= 1, \\ c_i^{q,p} &= c_{i-1}^{q,p} \tfrac{q-i+1}{q+p-i+1}. \end{aligned}$$

Idem, les coefficient $c_i^{p,q}$ sont calculés de la même manière.

Liste des tableaux

II.1 Comparaison entre les Tailles des deux méthodes. 57

V.1 Temps CPU sur un PIII 1.2MHz 128
V.2 Temps CPU sur un PIII 1.2MHz 131
V.3 Temps CPU sur un PIII 1.2MHz 135
V.4 Temps CPU sur un PIII 1.2MHz. 141

VI.1 Caractéristiques de simulation de chaque essai 154
VI.2 Caractéristiques de simulation de chaque test d'observation 164
VI.3 Caractéristiques de simulation de chaque essai d'observation, avec $E(t) = \pm 2\% y_m(t)$. 173
VI.4 Caractéristiques de simulation de chaque essai d'observation, avec $\mathbb{Y}(t) = [0.98X_m(t), 1.02X_m(t)]$. 180

Table des figures

I.1 Exemple de fonctions d'inclusion. $[\mathbf{f}^*]([\mathbf{x}])$ fonction d'inclusion minimale . 19

I.2 Exemple d'inversion ensembliste par SIVIA (a) seuil de la bissection μ_1 (b) seuil de la bissection $\mu_2 < \mu_1$ 24

II.1 Principe d'intégration numérique garantie 35

II.2 L'effet d'enveloppement 41

II.3 Factorisation **QR** [69]. 49

II.4 Comparaisons entre la méthode directe et la méthode de factorisation **QR** dans le cas des EDO coopératives, encadrement de la variable $x_1(t)$. 53

II.5 Comparaisons entre la méthode directe et la méthode de factorisation **QR** dans le cas des EDO coopératives, encadrement de la variable $x_2(t)$. 53

II.6 Exemple d'intégration avec la méthode de Hermite-Obreschkoff : Première composante du vecteur solution $\mathbf{x}(t)$. 59

II.7 Exemple d'intégration avec la méthode de Hermite-Obreschkoff : Deuxième composante du vecteur solution $\mathbf{x}(t)$. 59

III.1 Graphe d'incidence d'un système non préservateur d'ordre. 64

III.2 Graphe d'incidence d'un système préservateur d'ordre 65

III.3 Encadrement de toutes les valeurs possibles de $x_1(t)$. Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillées méthode de Hermite-Obreschkoff intervalle avec pas d'intégration variable. 71

III.4 Encadrement de toutes les valeurs possibles de $x_2(t)$. Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle avec pas d'intégration variable 72

III.5 Encadrement de toutes les valeurs possibles de $x_1(t)$.Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle. 74

III.6 Encadrement de toutes les valeurs possibles de $x_2(t)$. Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle. 74

III.7 Encadrement de toutes les valeurs possibles de $x_3(t)$. Courbes bleues continues méthodes basées sur le Théorème 1, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle. 75

III.8 Graphe d'incidence de (III.30) 75

III.9 Graphe d'incidence de (III.46) 81

III.10 Encadrement de toutes les valeurs possibles de $x_1(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle. 82

III.11 Encadrement de toutes les valeurs possibles de $x_2(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle. 82

III.12 Graphe d'incidence de (III.48) 83

III.13 Encadrement de toutes les valeurs possibles de $x_1(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle. 84

III.14 Encadrement de toutes les valeurs possibles de $x_2(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle. 85

III.15 Encadrement de toutes les valeurs possibles de $x_3(t)$. Courbes bleues continues méthodes basées sur le théorème de Müller, courbes rouges pointillés méthode de Hermite-Obreschkoff intervalle. 85

IV.1 Les différents modes d'encadrement possibles de (IV.10) 93

IV.2 Automate hybride 93

IV.3 Encadrement de toutes les solutions possibles de $x_1(t)$: courbes rouges pointillées méthode de Hermite-Obreschkoff ; courbes bleues continues méthode hybride d'encadrement. 100
IV.4 Encadrement de toutes les solutions possibles de $x_2(t)$: courbes rouges pointillées méthode de Hermite-Obreschkoff ; courbes bleues continues méthode hybride d'encadrement. 100
IV.5 Encadrement de toutes les solutions possibles de $x_3(t)$: courbes rouges pointillées méthode de Hermite-Obreschkoff ; courbes bleues continues méthode hybride d'encadrement. 101
IV.6 Les différents modes d'encadrement utilisés au cours de l'intégration garantie (IV.14). 101
IV.7 Les différents modes d'encadrement inférieurs possibles de (IV.18) 105
IV.8 Les différents modes d'encadrement supérieurs possibles de (IV.18) 106
IV.9 Automate hybride inférieur 106
IV.10 Automate hybride supérieur 107
IV.11 Encadrement de toutes les solutions possibles de $x_5(t)$: (a) Méthode hybride d'encadrement ; (b) Méthode de Hermite-Obreschkoff intervalle. Arrêt de l'algorithme de calcul après 0.35 seconde d'intégration. 110
IV.12 Encadrement de toutes les solutions possibles de $x_7(t)$: (a) Méthode hybride d'encadrement ; (b) Méthode de Hermite-Obreschkoff intervalle. Arrêt de l'algorithme de calcul après 0.35 seconde d'intégration. 111
IV.13 Encadrement de toutes les solutions possibles de $x_{11}(t)$: (a) Méthode hybride d'encadrement ; (b) Méthode de Hermite-Obreschkoff intervalle. Arrêt de l'algorithme de calcul après 0.35 seconde d'intégration. 111
IV.14 Les différents modes d'encadrement utilisés sur la période d'intégration garantie de (IV.21). La courbe rouge pointillée représente l'évolution de l'état discret du système hybride d'encadrement inférieur. La courbe bleue continue représente l'évolution de l'état discret du système hybride d'encadrement supérieur. 112

V.1 Principe d'identification à erreurs bornées 115
V.2 Banc d'essai thermique 120

V.3 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.981] \times [0.5, 0.501]$. Temps CPU=131s . 123

V.4 L'encadrement de toutes les sorties possibles $[y](t)$: (b) le pavé de paramètres est $[\mathbf{p}] = [0.98, 0.982] \times [0.5, 0.502]$, temps CPU=145s ; (c) le pavé de paramètres est $[\mathbf{p}] = [0.98, 0.983] \times [0.5, 0.503]$, temps CPU=157s. . . . 124

V.5 L'encadrement de toutes les sorties possibles $[y](t)$: (d) le pavé de paramètres est $[\mathbf{p}] = [0.98, 0.984] \times [0.5, 0.504]$, temps CPU=170s ; (e) le pavé de paramètres est $[\mathbf{p}] = [0.98, 0.985] \times [0.5, 0.505]$, temps CPU=190s. . . . 124

V.6 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.981] \times [0.5, 0.501]$. Courbes bleues continues méthode HOIS, temps CPU=42.7s ; Courbes rouges discontinues méthode HOI. 125

V.7 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.982] \times [0.5, 0.502]$. Courbes bleues continues méthode HOIS, temps CPU=43.4s ; Courbes rouges discontinues méthode HOI. 126

V.8 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.983] \times [0.5, 0.503]$. Courbes bleues continues méthode HOIS, temps CPU=44.8s ; Courbes rouges discontinues méthode HOI. 126

V.9 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.984] \times [0.5, 0.504]$. Courbes bleues continues méthode HOIS, temps CPU=45.6s ; Courbes rouges discontinues méthode HOI. 127

V.10 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.985] \times [0.5, 0.505]$. Courbes bleues continues méthode HOIS, temps CPU=48.1s ; Courbes rouges discontinues méthode HOI. 127

V.11 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.99] \times [0.5, 0.51]$. Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS, temps CPU=164s 130

V.12 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.991] \times [0.5, 0.511]$. Courbes bleues continues méthode HOIS, temps CPU=47s ; Courbes rouges discontinues méthode HOI, temps CPU=173s. 131

V.13 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.992] \times [0.5, 0.512]$. Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS. temps CPU=219s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.993] \times [0.5, 0.513]$. Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS. temps CPU= 304s. 132

V.14 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.994] \times [0.5, 0.514]$. Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS. temps CPU=368s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.995] \times [0.5, 0.515]$. Courbes bleues continues méthode TCID, temps CPU=47s ; Courbes rouges discontinues méthode HOIS. temps CPU= 482s. 132

V.15 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 1.48] \times [0.5, 1]$; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 1.98] \times [0.5, 1.5]$. Temps CPU=47s. 133

V.16 Approximation intérieure (pavés en rouge) de l'ensemble solution. Couche d'incertitude (pavés en noir), TC=6h. 134

V.17 Approximation intérieure (pavés en rouge) de l'ensemble solution. Couche d'incertitude (pavé en noir), TC=15h. 134

V.18 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.981] \times [0.5, 0.501]$. Courbes bleues continues méthode HOIS, temps CPU=98s ; Courbes rouges discontinues méthode HOI, temps CPU=245s. 136

V.19 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.982] \times [0.5, 0.502]$. Courbes bleues continues méthode HOIS, temps CPU=120s ; Courbes rouges discontinues méthode HOI, temps CPU=265s. 136

V.20 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes à un pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.983] \times [0.5, 0.503]$. Courbes bleues continues méthode HOIS, temps CPU=178s ; Courbes rouges discontinues méthode HOI, temps CPU=282s. 137

V.21 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.984] \times [0.5, 0.504]$. Courbes bleues continues méthode HOIS, temps CPU=213s ; Courbes rouges discontinues méthode HOI. temps CPU=300s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.985] \times [0.5, 0.505]$. Courbes bleues continues méthode HOIS, temps CPU=240s ; Courbes rouges discontinues méthode HOI. temps CPU= 329s. 137

V.22 L'évolution temporelle de la dérivée partielle $g_{5,1}(.)$ du système (V.7) . . 138

V.23 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.99] \times [0.5, 0.51]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU=311s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.991] \times [0.5, 0.511]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU= 348s. 139

V.24 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.992] \times [0.5, 0.512]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU=383s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.993] \times [0.5, 0.513]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU=433s. 139

V.25 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.994] \times [0.5, 0.514]$. Courbes bleues continues méthode TCID, temps CPU=176s ; Courbes rouges discontinues méthode HOIS. temps CPU=500s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 0.995] \times [0.5, 0.515]$. Courbes bleues continues méthode TCID, temps CPU=240s ; Courbes rouges discontinues méthode HOIS. temps CPU=578s. 140

V.26 L'encadrement de toutes les sorties possibles $[y](t)$ correspondantes : (a) pavé de paramètres $[\mathbf{p}]^T = [0.98, 1.48] \times [0.5, 1]$, temps CPU=200s ; (b) pavé de paramètres $[\mathbf{p}]^T = [0.98, 1.98] \times [0.5, 1.5]$, temps CPU=200s ; Courbes rouges discontinues méthode HOI. temps CPU= 329s. 140

V.27 Evolution du mode d'encadrement q utilisé par la méthode TCID. Le pavé de paramètres est $[\mathbf{p}]^T = [0.98, 1.98] \times [0.5, 1.5]$. 142

V.28 Approximation intérieure (pavés en bleu) de l'ensemble solution. Couche d'incertitude (pavé en noir), TC=24h. 142

VI.1 Phases prédiction et correction de la première approche d'estimation ensembliste 148

VI.2 Schéma de principe d'un bioréacteur. F_{in} et F_{out} représentent respectivement le débit de la solution d'entrée et le débit de la solution de sortie. . 150

VI.3 Arrêt de l'algorithme d'estimation d'état après quelques pas d'intégration (essai 1). 152

VI.4 Divergence de la taille des encadrements des variables d'état représentées par les courbes bleues continues (essai 2). La courbe rouge pointillée représente le signal discret de mesure. 153

VI.5 Divergence de la taille des encadrements des variables d'état, même dans le cas où les mesures sont précises (essai 3). 154

VI.6 Divergence de la taille des encadrements des variables d'état sous l'effet de l'accumulation du pessimisme (essai 4). 155

VI.7 Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 1 : (a) Biomasse ; (b) Substrat. 157

VI.8 Evolution du mode d'encadrement dans le contexte de l'essai 1. 158

VI.9 Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 2 : (a) Biomasse ; (b) Substrat. 159
VI.10 Evolution du mode d'encadrement dans le contexte de l'essai 2. 159
VI.11 Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 3 : (a) Biomasse ; (b) Substrat. 160
VI.12 Evolution du mode d'encadrement dans le contexte de l'essai 3. 161
VI.13 Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 4 : (a) Biomasse ; (b) Substrat. 161
VI.14 Evolution du mode d'encadrement dans le contexte de l'essai 4. 162
VI.15 Encadrement de toutes les valeurs possibles des deux variables d'état dans le contexte de l'essai 5 : (a) Biomasse ; (b) Substrat. 163
VI.16 Evolution du mode d'encadrement dans le contexte de l'essai 5. 163
VI.17 Encadrement de toutes les trajectoires d'état possibles dans le contexte de l'essai 1 : (a) $[x_1](t)$; (b) $[x_2](t)$. 170
VI.18 Encadrement de toutes les trajectoires d'état possibles dans le contexte du deuxième essai : (a) $[x_1](t)$; (b) $[x_2](t)$. 171
VI.19 Encadrement de toutes les trajectoires d'état possibles dans le contexte du troisième essai : (a) $[x_1](t)$; (b) $[x_2](t)$. 172
VI.20 Encadrement de toutes les trajectoires d'état possibles dans le contexte du quatrième essai : (a) $[x_1](t)$; (b) $[x_2](t)$. 172
VI.21 Automate hybride d'observation. 174
VI.22 Encadrement de toutes les trajectoires d'état admissibles : (a) $[x_1](t)$; (b) $[x_2](t)$. 176
VI.23 Evolution du mode d'encadrement 176
VI.24 Encadrement de toutes les trajectoires d'état possibles dans le contexte du deuxième essai : (a) $[x_1](t)$; (b) $[x_2](t)$. 177
VI.25 Evolution du mode d'encadrement dans le contexte du deuxième essai . . 177
VI.26 Encadrement de toutes les trajectoires d'état possibles dans le contexte du troisième essai : (a) $[x_1](t)$; (b) $[x_2](t)$. 178
VI.27 Evolution du mode d'encadrement dans le contexte du troisième essai . . 179

VI.28 Encadrement de toutes les trajectoires d'état possibles dans le contexte du quatrième essai : (a) $[x_1](t)$; (b) $[x_2](t)$. 179

VI.29 Evolution du mode d'encadrement dans le contexte du quatrième essai. . 180

VII.1 Exemple de vérification de sûreté d'un système incertain 186

VII.2 Projection sur le plan (x_2, x_3) de l'espace atteignable du (VII.11). Simulation par la méthode de HOI. 190

VII.3 Projection sur le plan (x_4, x_5) de l'espace atteignable du (VII.11). Temps CPU=0.5 sec (PIII 1.2MHz). Simulation par la méthode de HOI. 190

VII.4 L'évolution temporelle de $[x_1]$. Simulation par la méthode de HOI. 191

VII.5 L'évolution temporelle de $[x_2]$. Simulation par la méthode de HOI. Temps CPU=0.5 sec (PIII 1.2MHz) 191

VII.6 Projections de l'espace atteignable de (VII.12) (a) sur le plan (x_4, x_5), (b) sur le plan (x_6, x_7). Simulation par la méthode de HOI. 193

VII.7 L'évolution temporelle de variables d'état (a) $[x_4](t)$, (b) $[x_6](t)$. Simulation par la méthode de HOI. 194

VII.8 Projections de l'espace atteignable de (VII.12) (a) sur le plan (x_1, x_9), (b) sur le plan (x_2, x_{10}). Simulation par l'algorithme **Hybrid_Bound_Solution**. Temps CPU=250s. 194

VII.9 Projections de l'espace atteignable de (VII.12) (a) sur le plan (x_4, x_{12}), (b) sur le plan (x_5, x_{11}). Simulation par l'algorithme **Hybrid_Bound_Solution**. Temps CPU=250s. 195

VII.10 L'évolution temporelle de la variable d'état $[x_5](t)$. Simulation par l'algorithme **Hybrid_Bound_Solution**. Temps CPU=250s. 195

VII.11 L'évolution temporelle de variables d'état (a) $[x_7](t)$, (b) $[x_9](t)$. Simulation par l'algorithme **Hybrid_Bound_Solution**. Temps CPU=250s. . 196

VII.12 L'évolution du mode d'encadrement (ligne continue : système supérieur, ligne discontinue : système inférieur). Simulation par l'algorithme **Hybrid_Bound_Solution**. 197

VII.13 Projections de l'espace atteignable de (VII.13) (a) sur le plan (x_1, x_2), (b) sur le plan (x_2, x_3). Simulation par la méthode de HOI. 198

VII.14 Projections de l'espace atteignable de (VII.13) (a) sur le plan (x_3, x_4), (b) sur le plan (x_4, x_5). Simulation par la méthode de HOI. 198
VII.15 L'évolution temporelle de variables d'état (a) $[x_2](t)$, (b) $[x_3](t)$. Simulation par la méthode de HOI. 199
VII.16 L'évolution temporelle de variables d'état (a) $[x_4](t)$, (b) $[x_5](t)$. Simulation par la méthode de HOI. 199
VII.17 L'évolution temporelle de variables d'état (a) $[x_2](t)$, (b) $[x_3](t)$. Simulation par l'utilisation de la **Règle 2**. Temps CPU=13s. 201
VII.18 L'évolution temporelle de variables d'état (a) $[x_4](t)$, (b) $[x_5](t)$. Simulation par l'utilisation de la **Règle 2**. Temps CPU=13s. 201
VII.19 Projections de l'espace atteignable de (VII.13) (a) sur le plan (x_1, x_2), (b) sur le plan (x_2, x_3). Simulation par l'utilisation de la **Règle 2**. Temps CPU=13s. 202
VII.20 Projections de l'espace atteignable de (VII.13) (a) sur le plan (x_3, x_4), (b) sur le plan (x_4, x_5). Simulation par l'utilisation de la **Règle 2**. Temps CPU=13s. 202
VII.21 Biomasse, simulation par l'algorithme **Hybrid-Bounding**. 204
VII.22 Substrat, simulation par l'algorithme **Hybrid-Bounding**. La droite rouge parallèle à l'axe des abscisses représente la condition de garde $S = \sqrt{k_s k_i}$. 204
VII.23 Biomasse, simulation par l'algorithme **Hybrid-Bounding-Partition**. . . 205
VII.24 Substrat, simulation par l'algorithme **Hybrid-Bounding-Partition**. La droite rouge parallèle à l'axe des abscisses représente la condition de garde $S = \sqrt{k_s k_i}$. 205

Bibliographie

[1] R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical computer Science*, 138 :3–34, 1995.

[2] R. Alur, C. Courcoubetis, T.A. Henzinger, and P.-H. Ho. Hybrid automata : an algorithmic approach to the specification an verification of hybrid systems. *in : R.L. Grossman, A. Nerode, A.P. Ravn and H.Rischel, eds, Workshop on theory of hybrid systems, Lecture Notes in Computer Science, Springer, Berlin*, 736 :209–229, 1993.

[3] D. Angeli and E.D Sontag. Monotone control systems. *IEEE Transaction on automatic control*, 48(10) :1684– 1698, 2003.

[4] D. Angeli and E.D Sontag. A note on multistability and monotone i/o systems. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 67–72, Maui, Hawaii USA, December 2003.

[5] P.J. Antsaklis and A. Nerode. Special issue on hybrid systems. *IEEE Transaction on automatic control*, 43(4), 1998.

[6] E. Asarin, T. Dang, G. Frehse, A. Girard, C. L. Guernic, and O. Maler. Recent progress in continuous and hybrid reachability analysis. *In : Proceedings of IEEE Symp. on Computer-Aided Control Systems Design*, 2006.

[7] E. Asarin, T. Dang, and A. Girard. Reachability analysis of nonlinear systems using conservative approximation. In *Hybrid Systems : Computation and Control : 6th International Workshop, HSCC*, pages 20–35, Prague, 2003.

[8] E. Asarin, O. Maler, and A. Pnueli. Reachability analysis of dynamical systems

having piecewise-constant derivatives. *Theoretical Computer Science*, 138 :35–65, 1995.

[9] O. Bernard and J.L. Gouzé. Closed loop observers bundle for uncertain biotechnological models. *Journal of process control*, 14 :765–774, 2004.

[10] M. Berz and K. Makino. Verified integration of odes and flows using differential algebraic methods on high-order taylor models. *Reliable Computing*, 4(4) :361–369, 1998.

[11] M. Berz and K. Makino. Performance of taylor model methods for validated integration of odes. *Lecture Notes in Computer Science*, 3732(3732) :65–74, 2005.

[12] A. Boudenne. *Etude expérimentale et thermique des propriétés thermophysiques de matériaux composites a matrice polymère.* Thèse de doctorat université paris 12 val de marne, crétiel edition, 2003.

[13] F.L Chernousko. Ellipsoidal state estimation for dynamical systems. *Nonlinear analysis*, 2005.

[14] F.L Chernousko and D. Ya. Rokityanskii. Ellipsoidal bounds on reachable sets of dynamical systems with matrices subjected to uncertain perturbations. *Journal of optimization theory and applications*, 104 :1–19, 2000.

[15] L. Chisci, A. Garulli, and G. Zappa. Recursive state bounding by parallelotopes. *Automatica*, 32 :1049–1055, 1996.

[16] A. Chutinan and B. H. Krogh. Computational techniques for hybrid systems verification. *IEEE T. Automatic Control*, 48, 2003.

[17] C. Combastel. A state bounding observer for uncertain nonlinear continuous-time systems based on zonotopes. In *44th IEEE Conference on decision and control and European control conference ECC 2005*, 2005.

[18] D. Contois. Kinetics of bacterial growth relationship between population density and specific growth rate of continuous cultures. *J. Genetic Macrobiol*, 21 :40–50, 1959.

[19] G. Corliss and R. Rihm. Validating an a priori enclosure using high-ordre taylor series. *In : Alefeld, G., Formmer, A. (Eds.), Scientific computing, Computer Arithmetic, and Validated Numerics. Akademie Velag*, pages 228–238, 1996.

[20] P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *Proceedings of fifth annual symposium on principles of programming languages*, NY, 1978. ACM Press.

[21] Y. Deville, M. Janssen, and P.V. Hentenryck. Consistency techniques in ordinary differential equations. *2002 Kluwer Academic Publishers*, 7 :289–315, 2002.

[22] L. Doyen, T.A. Henzinger, and J.F. Raskin. Automatic rectangular refinement of affine hybrid systems. In *Proceedings of the Third International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS)*, pages 144–161. Springer, 2005.

[23] S.V. Drakunov. An adaptive quasioptimal filter with discontinuous parametres. *Automation ans remote Control*, 44 :76–86, 1983.

[24] P. Eijgenraam. *The solution of initial value problems using interval arithmetic.* Number 144. Amsterdamm : Mathematisch Centrum, 1981.

[25] S. Engell, S. Kowalewski, and J. Zaytoon Ed. Special issue on automatic of hybrid dynamical systems. *JESA*, 35(4), 2001.

[26] J.P. Gauthier, H. Hammouri, and S. Othman. Simple observer for nonlinear systems applications to bioreactors. *Transactions on Automatic Control*, 37 :875–880, 1992.

[27] J.C. Gilbert, G. Le Vey, and J. Masse. La différentiation automatique de fonctions représentées par des programmes. *Rapport de recherche de l'INRIA-Rocquencourt, Equipe : PROMATH 68 pages - Novembre 1991.*

[28] A. Girard. Reachability of uncertain linear systems using zonotopes. In *in M. Morari, L. Thiele (eds), Hybrid Systems : Computation and Control : 8th International Workshop, HSCC2005*, pages 291–305, Zurich, Switzerland, march 2005. LNCS 3414, Springer.

[29] A. Glumineau. New algebro-geometric conditions for the linearization by input-output injection. *Transactions on Automatic Control*, 41 :598–602, 1996.

[30] J.-L. Gouzé and K.P. Hadeler. Monotone flows and order intervals. *Nonlinear World*, 1 :23–34, 1994.

[31] J.L. Gouzé, A. Rapaport, and Z. M. Hadj-Sadok. Interval observers for uncertain biological systems. *Journal of Ecological Modelling*, 133 :45–56, 2000.

[32] A. Griewank. Ode solving via automatic differentiation and rational prediction. *Numerical Analysis*, 344 :37–56, 1995.

[33] A. Griewank, G.F. Corliss, P. Henneberger, G. Kirlinger, F.A. Potra, and H.J. Stetter. High-order stiff ode solvers via automatic differentiation and rational prediction. In *In lecture notes in Comput. Sci., 1196*, pages 114–125, Springer, Berlin, 1997.

[34] H. Guéguen and J. Zaytoon. Abstractions événementielles et approximations pour la vérification des systèmes dynamiques hybrides. In *Conférence Internationale Francophone d'Automatique CIFA04*, Douze, Tunisie, novembre 2004.

[35] H. Guéguen and J. Zaytoon. On the formal verification of hybrid systems. *Control Engineering Practice*, 12 :1253–1267, 2004.

[36] H. Guéguen and J. Zaytoon. Sur le calcul de l'espace atteignable des systèmes dynamiques hybrides. In *Conférence Internationale Francophone d'Automatique CIFA04*, Douze, Tunisie, novembre 2004.

[37] M.Z. Hadj-Sadok and J.L. Gouzé. Estimation of uncertain models of activated sludge processes with interval observers. *Journal of Process Control*, 11 :299–310, 2001.

[38] E. R. Hansen. Global optimization using interval analysis. New York, 1992. Marcel Dekker.

[39] G.I. Hargreaves. *Interval analysis in MATLAB*. Numerical analysis report no. 416, Manchester Centre for Computational Mathematics, 2002.

[40] T.A. Henzinger, P.H. Ho, and H. Wong-Toi. Algorithmic analysis of nonlinear hybrid systems. *IEEE Transaction on automatic control*, 43(4) :540 – 554, 1998.

[41] T.A. Henzinger, B. Horowitz, R. Majumdar, and H. Wong-Toi. Beyond hytech : Hybrids systems analysis using interval numerical methods. *In : Hybrid systems : Computation and control*, 1790 :130–144, 2000.

[42] T.A. Henzinger, P.W. Kopke, A. Puri, and P. Varaiya. What's decidable about hybrid automata ? the algorithmic analysis of hybrid systems. In *Proceedings of 27th. annual ACM Symposium on Theory of computing*, pages 373–382, 1995.

[43] R. Hermann. On the accessibility problem in control theory. In *Int. Symp. on*

nonlinear differential equations and nonliear mechanics, pages 325–332, New York, 1963. Academic Press.

[44] R. Hermann and A. J.Krener. Nonlinear controllability and observability. *Transactions on Automatic Control*, AC-22 :728–740, 1977.

[45] M. Hirsch and H.L. Smith. Monotone dynamical systems. In A. Canada, P. Drabek, and A. Fonda, editors, *Handbook of Differential Equations, Ordinary Differential Equations*, volume 2, chapter 4. Elsevier, 2005.

[46] M.W Hirsch and H.L Smith. Competitive and cooperative systems : a mini-review. *Lecture notes in control and information sciences*, 294 :183–190, 2004.

[47] J. Hoefkens, M. Berz, and K. Makino. Computing validated solutions of implicit differential equations. *Reliable Computing*, 19(19) :231–253, 2003.

[48] A. Isidori. *Nonlinear control systems*. Springer-Verlag, London, 1995.

[49] L. Jaulin. Nonlinear bounded-error state estimation of continuous-time systems. *Automatica*, 38 :10791–1082, 2002.

[50] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied Interval Analysis*. Springer, 2001.

[51] L. Jaulin and E. Walter. Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29(4) :1053–1064, 1993.

[52] R.E. Kalman. On a new approach to filtering and prediction problems. *ASME J.Basic Enginneering*, 28 :1960, 1960.

[53] M. Kieffer and E. Walter. Guaranteed nonlinear state estimation for continuous-time dynamical models from discrete-time measurements. In *proceedings 6th IFAC Symposium on robust control*, Toulouse, 2006.

[54] M. Kieffer, E. Walter, and I. Simeonov. Guaranteed nonlinear parameter estimation for continuous-time dynamical models. In *Proceedings 14th IFAC Symposium on System Identification*, pages 843–848, Newcastle, Australia, 2006.

[55] H. Kunze and D. Siegel. A graph theoretical approach to monotonicity with respect to initial conditions ii. *Nonlinear Analysis*, 35 :1–20, 1999.

[56] A. B. Kurzhanski and P. Varaiya. Ellipsoidal techniques for reachability analysis. *Springer*, pages 202–214, 2004.

[57] A. B. Kurzhanski and P. Varaiya. Ellipsoidal techniques for hybrid dynamics : the reachability problem. *In : Dayawansa, W., Lindquist, A., Zhou, Y.(Eds.), New Directions and Applications in Control Theory, Lecture Notes in Control and Information Sciences*, 65 :1150–1167, 2005.

[58] M.-A Lefebvre and H. Guéguen. Hybrid abstractions of affine systems. *Nonlinear Analysis*, 321 :93–205, 2005.

[59] R. J Lohner. Enclosing the solutions of ordinary initial and boundary value problems. *In : Wiley (Ed.), Computer Arithmetic : Scientific Computation and Programming Languages*, pages 255–286, 1987.

[60] R. J Lohner. *Einschließung der Lösung gewöhnlicher Anfangs- und Randwertaufgaben und Anwendungen.* Thèse de doctorat, Universität Karlsruhe, 1988.

[61] D.G. Luenberger. An introduction to observers. *Transactions on Automatic Control*, AC-16, 1971.

[62] K. Makino and M. Berz. Cosy infinity version 9. *Nuclear Instruments and Methods in Physics Research A*, 558(558) :346–350, 2006.

[63] C. Marcelli and P. Rubbioni. A new extension of classical müller theorem. *Nonlinear Analysis, Theory, Methods and Applications*, 28(11) :1759–1767, 1997.

[64] M. Milanese, J.P. Norton, H. Piet-Lahanier, and E. Walter. *Bounding approaches to system identification.* New York, plenum press edition, 1996.

[65] E.A. Misawa and J.K. Hedrick. Nonlinear observers-a state -of-the-art survey. *Journal of dynamic systems, measurement, and control*, 111 :344–352, 1989.

[66] M. Müller. Über das fundamentaltheorem in der theorie der gewohnlichen differentialgleinchungen. *Math.Z*, 26 :619–645, 1926.

[67] M. Moisan and O. Bernard. Interval observers for non monotone systems. application to bioprocess models. In *proceedings of the 16th IFAC world conference*, Prague, 2005.

[68] R. E. Moore. *Interval Analysis.* Englewood Cliffs, prentice-hall edition, 1966.

[69] N.S. Nedialkov. *Computing rigorous bounds on the solution of an initial value problem for an ordinary differential equation.* Thèse de doctorat, Univ. of Toronto, 1999.

[70] N.S. Nedialkov and K. Jackson. An interval Hermite-Obreschkoff method for computing rigorous bounds on the solution of an initial value problem for an ordinary differential equation. *Reliable Computing*, 5 :289–310, 1999.

[71] N.S. Nedialkov and K. Jackson. The design and implementation of a validated object-oriented solver for ivps for odes. In *Tech. Rep. 6, Software Quality Research Laboratory, Department of computing and Software, L8S 4L7*, Hamilton, Canada, 2002. McMaster University.

[72] N.S. Nedialkov, K. Jackson, and J. Pryce. An effective high-order interval method for validating existence and uniqueness of the solution of an ivp for an ode. *Reliable Computing*, 7(6) :449–465, 2001.

[73] R. Ortega and L. Sánchez. Abstract competitive systems and orbital stability in $\mathbb{R}^3$. *Proc. Amer. Math. Soc.*, 128 :2911–2919, 2000.

[74] A. Puri and P. Varaiya. Decidability of hybrid systems with rectangular differential inclusions. In *Proceedings of 6th Int. Conference on computer Aided-Verification, LNCS-818.*, 1994.

[75] T. Raïssi, N. Ramdani, and Y. Candau. Parameter estimation for non-linear continuous-time systems in a bounded error context. In *Proceedings 42nd IEEE Conference on Decision and Control*, pages 2240–2245, Hawaii, 2003.

[76] T. Raïssi, N. Ramdani, and Y. Candau. Set membership state and parameter estimation for systems described by nonlinear differential equations. *Automatica*, 40(10) :1771–1777, 2004.

[77] T. Raïssi, N. Ramdani, and Y. Candau. Bounded-error moving horizon state estimator for non-linear continuous-time systems : application to a bioprocess system. *Journal of Process Control*, 15(5) :537–545, 2005.

[78] L.B. Rall and G. F. Corliss. Introduction to automatic differentiation. In *Computational Differentiation : Techniques, applications, and tools. SIAM*, pages 1–18, Philadelphia, 1996.

[79] N. Ramdani, N. Meslem, T. Raïssi, and Y. Candau. Set-membership identification of continuous-time systems. In *Proceedings 14th IFAC Symposium on System Identification*, pages 446–451, Newcastle, Aus 2006.

[80] A. Rapaport and D. Dochain. Interval observers for biochemical processes with uncertain kinetics and inputs. *Mathematical biosciences*, 193 :235–253, 2005.

[81] A. Rapaport and J.L. Gouzé. Parallelotopic and practical observers for nonlinear uncertain systems. *International journal of control*, 76(3) :237–251, 2002.

[82] T. Raïssi. Méthodes ensembliste pour l'estimation d'état et de paramètres. *Thèse de doctorat université Paris 12 Val de Marne, Crétiel*, 2004.

[83] A. Rauh, M. Kletting, H. Aschemann, and E.P Hofer. Interval methods for simulation of dynamical systems with state-dependent switching characteristics. In *Proceedings of the 2006 IEEE International Conference on Control Applications*, pages 355–360, Munich, 2006.

[84] R. Rihm. Interval methods for initial value problems in odes. In *Computational Mathematics, E. S. (Ed.), Validated computations : proceedings of the IMACS-GAMM International Workshop on Validated Computations*, pages 173–208, Amsterdam, 1994.

[85] R. Rihm. Implicit methods for enclosing solutions of odes. *Journal of Universal Computer Science*, 42(6) :202–209, 1998.

[86] A.B Singer and P.I Barton. Bounding the solutions of parameter dependent nonlinear ordinary differential equations. *SIAM Journal on Scientific Computing*, 27(6) :2167–2182, 2006.

[87] J.J.E Slotine, J.K Hedrick, and E.A Misawa. On sliding observers. In *American control conference*, Seattle, 1986.

[88] H.L. Smith. *Monotone dynamical systems : An introduction to the theory of competitive and cooperative systems.* AMS. Providence, RI, 1995.

[89] H.L. Smith and P. Waltman. *The theory of chemostat.* American Mathematical Society, 1995.

[90] E.D. Sontag. Molecular systems biology and control. *European journal of control*, 11 :396–435, 2005.

[91] H.W. Sorenson. Kalman filtering : Theory and application. *IEEE Press*, 1985.

[92] T. Sunaga. Theory of interval algebra and its application to numerical analysis. *RAAG Memoirs, Ggujutsu Bunken Fukuy-kai*, 2 :547–564, 1958.

[93] R. Uhl. Ordinary differential inequalities and quasimonotonicity in ordered topological vector spaces. *Proc. Amer. Math. Soc.*, 126 :1999–2003, 1998.

[94] E. Walter and M. Kieffer. Interval analysis for guaranteed nonlinear parameter estimation. In *Proceedings 13th IFAC Symposium on System Identification*, pages 259–270, Rotterdam, 2003.

[95] E. Walter and M. Kieffer. Guaranteed optimisation of the parameters of continuous-time knowledge-based models. In *C. Commault and N. Marchand, editors, Positive systems - LNCIS 341*, pages 137–144, Heidelberg, 2006. Springer.

[96] E. Walter and M. Kieffer. Sur quelques idées fausses ayant des conséquences en identification. In *JIME'2006*, Poitiers, Novembre 2006.

[97] E. Walter and L. Pronzato. *Identification of parametric models from experimental data.* Springer-Verlag, London, 1997.

[98] W. Walter. *Differential and integral inequalities.* Springer-Verlag, Berlin, 1970.

[99] W. Walter. Differential inequalities and maximum principles : theory, new methods and applications. *Nonlinear analysis, Theory, Methods and Applications*, 30(8) :4695–4711, 1997.

[100] G. Wanner. On the integration of stiff differential equations. In *Proceedings of the colloquium on Numerical Analysis, volume 37 of Internat. Ser. Numer. Math.*, pages 209–226, Basel, Birkhäuser, 1977.

[101] G. Wanner. On the integration of stiff differential equations. In *Technical report*, Université de Genève, Section de Mathematique, 1211, Suisse, October 1979.

[102] N. Wiener. A contribution to the theory of relative position. In *Proc. Cambridge Philos. Soc*, pages 441–449, Cambridge Philos, 1914. Vol.

[103] N. Wiener. A new theory of measurement : a study in the logic of mathematics. In *Proceeding Mathematical Society*, London, 1921.

[104] J. Zaytoon. Special issue on hybrid systems. *JESA*, 32(9-10), 1998.

Résumé : Cette thèse porte sur le calcul d'une sur-approximation conservative pour les solutions d'équations différentielles ordinaires en présence d'incertitudes et sur son application à l'estimation et l'analyse de systèmes dynamiques à temps continu. L'avantage principal des méthodes et des algorithmes de calculs présentés dans cette thèse est qu'ils apportent une preuve numérique de résultats. Cette thèse est organisée en deux parties. La première partie est consacrée aux outils mathématiques et aux méthodes d'intégration numérique garantie des équations différentielles incertaines. Ces méthodes permettent de caractériser de manière garantie l'ensemble des trajectoires d'état engendrées par un système dynamique incertain dont les incertitudes sont naturellement représentées par des intervalles bornés. Dans cette optique, nous avons développé une méthode d'intégration hybride qui donne de meilleurs résultats que les méthodes d'intégration basées sur les modèles de Taylor intervalles. La seconde partie aborde les problèmes de l'identification et de l'observation dans un contexte à erreurs bornées ainsi que le problème d'atteignabilité continue pour la vérification de propriétés des systèmes dynamiques hybrides.
Mots clés : Equations différentielles ordinaires, systèmes dynamiques continus, systèmes dynamiques hybrides, analyse par intervalles, erreurs bornées, identification, observation, atteignabilité.

Abstract : This thesis addresses the computation of conservative over-approximation of the solutions of uncertain ordinary differential equations and its application to the estimation and the analysis of uncertain continuous-time dynamical systems. The main feature of the methods and algorithms presented in this thesis is the fact that they are numerically verified and hence can be used to obtain numerical proof of properties. This thesis is organized in two parts. The first part is devoted to the mathematical tools and the guaranteed numerical integration methods for uncertain ordinary differential equations. These methods make it possible to characterize in a guaranteed way all the state trajectories generated by an uncertain dynamical system whose uncertainties are in a natural way described by bounded boxes. Accordingly, we have developed a hybrid integration method which gives better results than the integration methods based on interval Taylor models. The second part is dedicated to the resolution of identification and observation issues in a bounded error context. It also deals with continuous reachability computation for the verification of the properties of hybrid dynamical systems.
Keywords : Ordinary differential equations, continuous-time dynamical systems, hybrid dynamical systems, interval analysis, bounded uncertainties, identification, observation, reachability.