

MASTER RECHERCHE SMIS-EEAS

**Spécialité : Systèmes Automatiques,
Informatiques et Décisionnels,**

Année Universitaire 2005-2006

**ESTIMATION D'ETAT ET DE PARAMETRES
EN UTILISANT L'ANALYSE PAR INTERVALLES**

RIBOT Pauline

Groupe : DISCO

Responsable de stage : C. Jauberthie

Remerciements

Le travail présenté dans ce rapport a été effectué au Laboratoire d'Analyse et d'Architecture des Systèmes (*LAAS*) du Centre National de la Recherche Scientifique (*CNRS*), au sein du groupe Diagnostic, supervision et conduite qualitatifs (*DISCO*), en vue de l'obtention du Diplôme de Master Recherche Systèmes Automatiques, Informatiques et décisionnels *S.A.I.D* de l'Université Paul Sabatier.

Je remercie Monsieur Malik Ghallab, directeur du *LAAS* pour m'avoir permis d'effectuer ce stage et Madame Louise Travé-Massuyès pour m'avoir accueillie dans le groupe *DISCO*.

Je remercie Monsieur Jean-Louis Calvet pour m'avoir acceptée dans sa formation ainsi que tous mes professeurs.

J'exprime également mes plus sincères remerciements à ma responsable de stage Carine Jauberthie pour avoir assuré l'encadrement de ce travail et m'avoir judicieusement prodigué ses encouragements.

Merci aussi aux permanents et aux doctorants du groupe *DISCO* particulièrement Aïmed, Siegfried et Hervé pour leur soutien moral.

Mille mercis aux doctorants et stagiaires du groupe *MAC* pour m'avoir supportée tout au long des pauses café! Enfin, un merci tout particulier à Christophe pour son aide précieuse et efficace.

Résumé

Deux problèmes sont étudiés dans ce rapport: l'estimation d'état et de paramètres d'un modèle dynamique à partir de données expérimentales. Généralement, ces problèmes sont résolus par des méthodes stochastiques dans lesquelles les bruits et perturbations sont assimilées à des variables aléatoires suivant une loi de probabilité connue a priori. Cependant, cette hypothèse est rarement vérifiée en pratique.

Nous préférons donc adopter une approche ensembliste en considérant que les variables appartiennent à des domaines admissibles. Ces domaines sont caractérisés par des bornes acceptables ou connues a priori. Le problème d'estimation est alors abordé dans un contexte ensembliste où seules les bornes d'erreur résultant des bruits de mesure sont connues. Les résultats obtenus ne sont plus des valeurs ponctuelles mais des ensembles garantis contenir la solution.

La première étape pour résoudre ce problème consiste à développer une méthode d'intégration du modèle incertain. La méthode d'intégration choisie est basée sur des développements en série de Taylor et sur l'arithmétique par intervalles. Nous vérifions ensuite dans une deuxième étape l'appartenance des variables estimées au domaine d'admissibilité dont la valeur est déterminée par les bornes d'erreurs connues. Des algorithmes ont été développés afin d'appliquer cette méthode. Leur efficacité est illustrée par un modèle de comportement en vol longitudinal d'un avion.

Table des matières

Introduction	5
I L'estimation dans un contexte ensembliste	7
I.1 Contexte à erreurs bornées	7
I.2 Problème d'estimation dans un contexte ensembliste	9
I.2.1 Estimation paramétrique ensembliste	9
I.2.2 Estimation d'état ensembliste	11
II Analyse par intervalles	13
II.1 Définitions	13
II.1.1 Intervalle	13
II.1.2 Pavé et sous-pavage	14
II.2 Calcul par intervalles	14
II.2.1 Arithmétique par intervalles	15
II.2.2 Fonctions élémentaires	15
II.3 Fonction d'inclusion	16
II.4 Image réciproque d'un ensemble	19
II.4.1 Tests d'inclusion	20
II.4.2 Bissection	20
II.4.3 Algorithme SIVIA	20
II.5 Image directe d'un ensemble	21
III Intégration numérique garantie des équations différentielles ordinaires	23
III.1 Intégration numérique garantie basée sur les séries de Taylor	23
III.2 Existence et unicité de la solution	25
III.2.1 Théorème du point fixe	25
III.2.2 Opérateur de Lindelöf	26
III.2.3 Recherche d'une solution a priori	26
III.3 Réduction de la solution	28
III.3.1 Méthode directe	28
III.3.2 Effet d'enveloppement	30

III.3.3	Méthode de la valeur moyenne étendue	31
IV	Estimation d'état et de paramètres	33
IV.1	Estimation d'état	33
IV.1.1	Estimateur causal	34
IV.1.2	Estimateur non causal	35
IV.2	Estimation de paramètres	36
V	Application	38
V.1	Modèle de comportement en vol d'un avion	38
V.2	Estimation d'état du modèle	40
	Conclusion	43
	Références bibliographiques	44

Introduction

La modélisation des processus peut parfois être rendue difficile par la présence de bruits et de perturbations. Lorsque des hypothèses peuvent être faites sur la nature de ces bruits et perturbations, ils peuvent être assimilés à des lois de probabilité. Il est alors possible de les modéliser et d'inclure ce modèle dans celui du processus étudié.

Quand la structure du modèle est connue, l'état du processus peut être reconstruit. Dans un contexte statistique, le problème d'estimation de paramètres nécessite le choix d'un critère et l'optimisation de celui-ci (comme la minimisation du critère des moindres carrés). En réalité les variables de bruits ou perturbations ne suivent pas des lois de probabilité, les hypothèses stochastiques faites sur les bruits et les perturbations sont donc rarement satisfaites. Une alternative à l'approche probabiliste consiste à considérer que les seules informations disponibles sur les variables incertaines sont les ensembles de valeurs auxquelles elles appartiennent. Il s'agit de l'approche ensembliste. Les ensembles sont habituellement caractérisés par leurs bornes. Les grandeurs incertaines sont donc supposées varier entre des bornes connues ou acceptables.

Il est donc naturel de se concentrer sur des estimateurs ensemblistes, dans un cadre déterministe à erreurs bornées. Les résultats obtenus par ces méthodes ensemblistes ne sont plus des valeurs ponctuelles mais des ensembles de valeurs qui contiennent à coup sûr toutes les solutions possibles au problème. C'est pourquoi ils sont dits garantis ou validés. Les outils ensemblistes ont tout d'abord été développés pour l'arithmétique et l'informatique dans le cadre de l'arithmétique par intervalles. En Automatique, ils sont utilisés pour résoudre des problèmes d'estimation d'état ou de paramètres dans un contexte à erreurs bornées. L'estimation ensembliste a pour objectif de caractériser l'ensemble des valeurs du vecteur des paramètres ou des variables d'état qui sont admissibles, c'est-à-dire qui correspondent à des erreurs inconnues mais variant entre des bornes d'erreur connues a priori. N'importe quel point de l'ensemble de valeurs admissibles est une solution potentielle du problème d'estimation.

Le but de ce stage, effectué dans le groupe Diagnostic, Supervision et Conduite qualitatifs *DISCO* du LAAS-CNRS, est dans un premier temps d'étudier les techniques récemment développées de l'analyse par intervalles pour ensuite mettre en oeuvre des algorithmes sous Matlab permettant de reconstruire l'état et d'identifier les paramètres d'un modèle

incertain.

Dans un premier chapitre, nous présentons les problèmes d'estimation d'état et de paramètres dans un contexte ensembliste. Les ensembles de valeurs admissibles pour les paramètres et les variables d'état sont caractérisés à partir des bornes de l'ensemble d'erreur connu et des mesures bruitées qui sont disponibles à tout instant. Le deuxième chapitre reprend les bases du calcul par intervalles. Des méthodes permettant de manipuler les ensembles et les sous-ensembles y sont présentées. Pour reconstruire l'état du système, le modèle dépendant de variables incertaines doit être intégré. Nous présentons dans le Chapitre III une méthode d'intégration garantie basée sur le développement en série de Taylor. Le quatrième chapitre donne une solution au problème d'estimation d'état et de paramètres en utilisant les techniques ensemblistes présentées dans le Chapitre II et les algorithmes mis en oeuvre dans le Chapitre III. Enfin, l'ensemble de ces méthodes sont appliquées au problème d'estimation de l'état d'un modèle de comportement en vol longitudinal d'un avion dans le dernier chapitre.

Chapitre I

L'estimation dans un contexte ensembliste

Les problèmes d'estimation de grandeurs d'un modèle (état ou paramètres) à partir de données expérimentales sont habituellement résolus par des méthodes probabilistes [WAL 94] où les bruits de mesure ou perturbations sont assimilés à des lois de probabilité. Contrairement à cette approche, les erreurs de mesure et de modélisation seront supposées inconnues mais variant entre des bornes connues ou acceptables a priori.

Le problème d'estimation de grandeurs est donc traité dans un contexte à erreurs bornées.

I.1 Contexte à erreurs bornées

L'erreur entre les sorties mesurées et les sorties prédites résulte entre autre des bruits de mesure dûs à la précision limitée des capteurs, des perturbations diverses pouvant agir sur le système, et de l'erreur structurelle qui traduit notre capacité limitée à modéliser les systèmes physiques. Dans notre cas, seuls les bruits de mesure seront pris en compte, les autres sources d'erreur n'étant pas quantifiables. Si cette erreur, notée e , est supposée additive sur la sortie du modèle y , les mesures expérimentales y_{exp} sont alors reliées aux sorties prédites y par la relation

$$y_{exp} = y + e. \quad (\text{I.1})$$

Dans un contexte à erreurs bornées, l'erreur e est supposée varier entre une borne inférieure e_{min} et une borne supérieure e_{max} . Ces bornes sont supposées acceptables. Un ensemble

d'erreur admissible \mathbb{E} peut alors être défini par un ensemble de contraintes de la manière suivante :

$$\mathbb{E} = \{e \mid e_{min} \leq e \leq e_{max}\}. \quad (I.2)$$

Les valeurs des bornes sont établies à partir des spécifications relevées sur les instruments de mesure (ou capteurs). Ces spécifications sont fournies par le constructeur.

Dans ce rapport, les bornes sur l'erreur sont supposées constantes à tout instant. Cependant le cas d'erreurs variant avec le temps peut aussi être envisagé avec ces méthodes ensemblistes.

Nous étudions un système (voir le schéma bloc de la Figure I.1 dont la condition initiale réelle x_0^{reel} est inconnue mais appartient à un intervalle connu, cependant les sorties mesurées sur le procédé physique y_{exp} sont disponibles à chaque instant. La condition initiale de notre modèle x_0 est donc aussi inconnue mais nous supposons qu'elle appartient à un ensemble connu a priori \mathbb{X}_0 . Puisque la condition initiale du modèle appartient à un ensemble, la sortie y appartient également à un ensemble noté $[y]$ et l'erreur e à un ensemble noté $[e]$. L'ensemble $[e]$ doit être compris dans le domaine d'admissibilité \mathbb{E} .

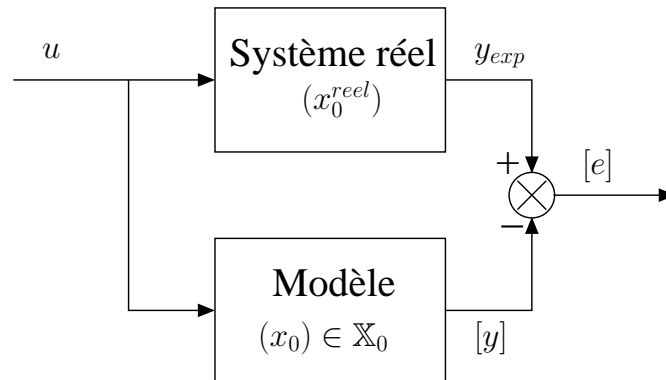


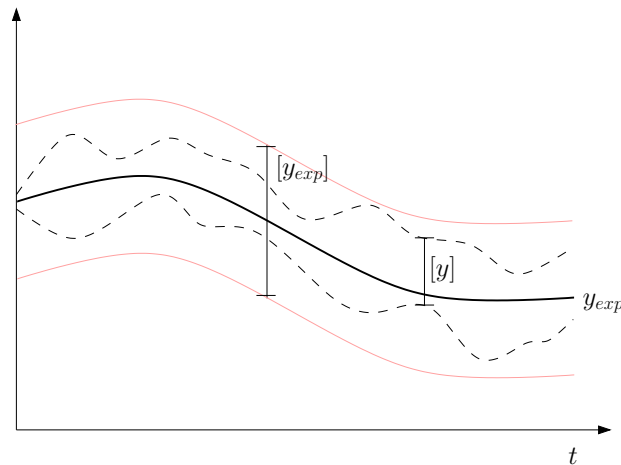
FIG. I.1 – Schéma bloc

De la même façon que pour l'erreur $[e]$, nous pouvons définir un domaine admissible \mathbb{Y} (voir Figure I.1) pour les sorties du modèle $[y]$ tel que :

$$\mathbb{Y} = \{[y] \mid [y] \subset [y_{exp}] = [y_{exp} - e_{max}, y_{exp} - e_{min}]\}. \quad (I.3)$$

où y_{exp} sont les mesures expérimentales, et où e_{min} et e_{max} sont les bornes d'erreur connues.

De nombreuses approches ont été explorées dans le cas d'un modèle linéaire : des méthodes ellipsoïdales [LES 03] [MIL 91] [DUR 01], parallélotopiques et zonotopes [ALA 05]. Nous nous intéressons ici au cas où le modèle est non-linéaire. Nous allons utiliser l'analyse par intervalles qui permet de rejeter les modèles non consistants avec les données expérimentales et les bornes d'erreurs prédéfinies.

FIG. I.2 – Détermination du domaine admissible \mathbb{Y}

Dans la suite de ce chapitre nous présentons l'approche à erreurs bornées pour résoudre un problème d'estimation d'état et de paramètres d'un modèle.

I.2 Problème d'estimation dans un contexte ensembliste

Dans un contexte à erreurs bornées, l'estimation d'une grandeur physique (état ou paramètre) ne nous amène plus à une valeur ponctuelle, comme dans le cas des méthodes probabilistes, mais à un ensemble de solutions ou une union de sous-ensembles disjoints de solutions qui sont consistantes avec la structure du modèle, les données expérimentales et les bornes d'erreur prédéfinies.

I.2.1 Estimation paramétrique ensembliste

Le problème d'estimation paramétrique ensembliste consiste à déterminer l'ensemble des valeurs du vecteur de paramètres d'un modèle à partir de mesures expérimentales, de telle sorte que toutes les erreurs soient comprises entre des bornes prédéfinies.

Considérons le modèle d'état paramétrique continu non-linéaire décrit par les équations suivantes:

$$\begin{cases} \dot{x}(t,p) = f(x(t),u(t),p), \\ y(t,p) = g(x(t),u(t),p), \\ x_0 \in \mathbb{X}_0, \\ p \in \mathbb{P}_0, \end{cases} \quad (\text{I.4})$$

où

- f et g , sont des fonctions continues non-linéaires connues,
- $x(t) \in \mathbb{R}^n$ est le vecteur d'état à l'instant t ,
- $u(t) \in \mathbb{R}^m$ est le vecteur d'entrée à l'instant t ,
- $y(t) \in \mathbb{R}^r$ est le vecteur de sortie à l'instant t ,
- \mathbb{X}_0 est l'ensemble connu a priori qui contient la condition initiale x_0 inconnue,
- \mathbb{P}_0 est l'ensemble de recherche connu a priori supposé contenir toutes les valeurs recherchées du vecteur de paramètres p .

Un vecteur de paramètres p est dit acceptable si, et seulement si, l'erreur entre les mesures y_{exp} et les sorties du modèle $[y]$ est comprise entre des bornes connues a priori. Pour estimer les paramètres d'un système dans un contexte à erreurs bornées, il faut alors déterminer \mathbb{P} l'ensemble de tous les paramètres p contenu dans l'ensemble de recherche a priori \mathbb{P}_0 afin que l'erreur entre les données expérimentales et les sorties du modèle, notée

$$[e(p)] = y_{exp} - [y(p)], \quad (\text{I.5})$$

appartienne à l'ensemble d'erreur admissible \mathbb{E} dont les bornes e_{min} et e_{max} sont connues:

$$\mathbb{P} = \{p \in \mathbb{P}_0 \mid e(p) \in \mathbb{E}\} = \{p \in \mathbb{P}_0 \mid e_{min} \leq [e(p)] \leq e_{max}\} \quad (\text{I.6})$$

La caractérisation de l'ensemble \mathbb{P} peut être défini comme un problème d'inversion ensembliste :

$$\mathbb{P} = [e^{-1}](\mathbb{E}). \quad (\text{I.7})$$

Une approximation garantie de \mathbb{P} peut être obtenue avec l'analyse par intervalles et l'algorithme SIVIA développé par Luc Jaulin [JAU 93b] et présenté dans le Chapitre II.

Le problème d'estimation de paramètres peut aussi se formuler de la manière suivante :

$$\mathbb{P} = \{p \in \mathbb{P}_0 \mid [y(p)] \subset [y_{exp}]\} \quad (\text{I.8})$$

où $[y(p)]$ représente la sortie de notre modèle et $[y_{exp}]$ le domaine admissible pour la sortie définie dans (I.3).

Sur la Figure I.3, l'estimation du vecteur de paramètres p est schématisée. Dans cet exemple, nous cherchons à estimer deux paramètres p_1 et p_2 à partir de mesures y_{exp} et de bornes d'erreur définissant un domaine admissible $[y_{exp}]$ pour les sorties du modèle $[y(p)]$. Les valeurs de y_{exp} proviennent des capteurs à des instants précis, nous avons donc des mesures discrètes.

Lorsque le système est décrit par des équations différentielles ordinaires, l'évaluation de la sortie du modèle continu aux instants correspondant aux mesures doit être réalisée

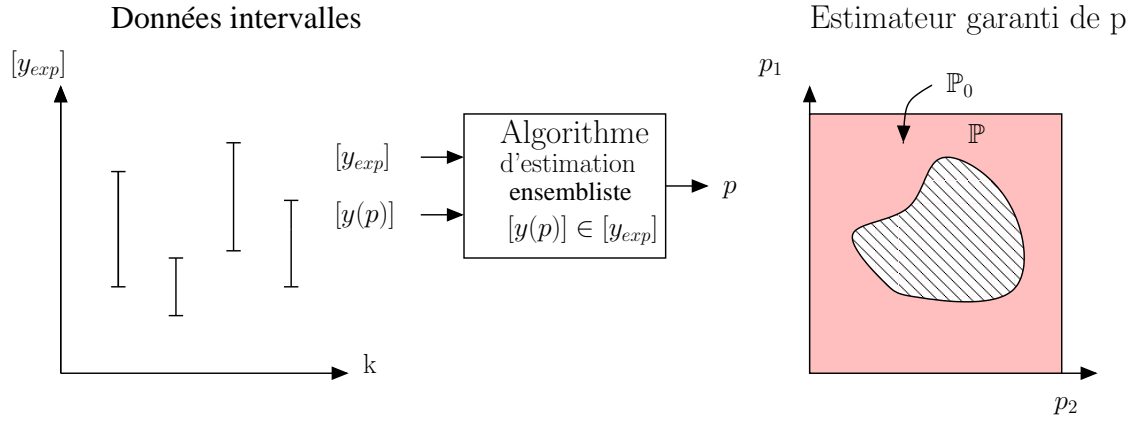


FIG. I.3 – Estimation de paramètres dans un contexte ensembliste

sous forme ensembliste. Il existe plusieurs méthodes d'intégration garanties pour évaluer la sortie du modèle. Celle que nous avons choisie d'utiliser ici est une méthode d'intégration numérique garantie par modèles de Taylor intervalles [COR 94][NED 99][RAM 05]. Cette méthode d'intégration est développée dans le Chapitre III.

I.2.2 Estimation d'état ensembliste

L'estimation d'état est un problème moins complexe à résoudre que l'estimation de paramètres. Les méthodes ensemblistes utilisées sont les mêmes mais le vecteur de paramètres p est ici connu. Nous cherchons à estimer l'état d'un système dynamique décrit par les équations suivantes:

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)), \\ y(t) = g(x(t), u(t)), \\ x_0 \in \mathbb{X}_0, \end{cases} \quad (\text{I.9})$$

où

- $x(t) \in \mathbb{R}^n$ est le vecteur d'état à l'instant t ,
- $u(t) \in \mathbb{R}^m$ est le vecteur d'entrée à l'instant t ,
- $y(t) \in \mathbb{R}^r$ est le vecteur de sortie à l'instant t ,
- \mathbb{X}_0 est l'ensemble connu a priori qui contient la condition initiale x_0 .

Les fonctions f et g sont des fonctions continues non-linéaires connues.

Pour estimer les états à partir des données expérimentales, il faut calculer l'ensemble \mathbb{X} de tous les états compatibles avec le modèle et les bornes d'erreurs prédéfinies. La sortie de notre modèle doit appartenir au domaine admissible \mathbb{Y} défini dans l'équation (I.3).

Les méthodes pour résoudre ce problème d'estimation d'état sont similaires à celles mises en oeuvre pour l'identification de paramètres. Les deux techniques utilisées seront donc

l'intégration numérique garantie d'équations différentielles ordinaires (EDO) par modèle de Taylor et l'inversion ensembliste.

L'estimation d'état dans le cas non-linéaire a déjà été traitée [KIE 02], par exemple pour une application à la localisation d'un robot mobile [KIE 00].

Dans le chapitre suivant, nous rappelons les notions de base de l'analyse par intervalles qui seront utilisées par la suite. Le troisième chapitre présente la méthode d'intégration numérique garantie choisie, basée sur les séries de Taylor, pour obtenir une solution garantie de l'équation différentielle ordinaire décrivant le comportement du système étudié. Des algorithmes pour estimer les états et les paramètres du modèle (à l'aide de l'algorithme SIVIA) sont proposés dans le Chapitre IV. Enfin, ces algorithmes sont appliqués à l'exemple du modèle d'un comportement en vol d'un avion dans le dernier chapitre.

Chapitre II

Analyse par intervalles

L'analyse par intervalles a été tout d'abord introduite par R. E. Moore [MOO 66]. Elle a été utilisée notamment pour la commande robuste de systèmes [JAU 02b] ainsi que pour l'estimation de paramètres [JAU 93a] [RAï 03]. Dans ce chapitre, les bases du calcul par intervalles sont présentées. Ces notions de l'analyse par intervalles peuvent être retrouvées dans l'ouvrage [JAU 01].

II.1 Définitions

L'analyse par intervalle repose sur la représentation des nombres réels ou entiers par des intervalles qui les contiennent. Les intervalles peuvent être utilisés dans la prise en compte d'une incertitude sur une valeur mesurée. Cette valeur est alors encadrée entre une borne inférieure et une borne supérieure connues. L'objectif de l'analyse par intervalles est de calculer des résultats sous la forme d'intervalles contenant à coup sûr la solution. Ces résultats sont alors dits garantis, ou encore validés.

II.1.1 Intervalle

Définition II.1 (Intervalle) *Un intervalle $[x]$ de \mathbb{R} , est un ensemble fermé et connexe de valeurs réelles se notant :*

$$[x] = [\underline{x}, \bar{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \bar{x}\}. \quad (\text{II.1})$$

où $\underline{x} \in \mathbb{R}$ est la borne supérieure de l'intervalle $[x]$ et $\bar{x} \in \mathbb{R}$ est la borne inférieure.

Par la suite, nous l'appellerons variable intervalle. L'ensemble de tous les intervalles de \mathbb{R} est noté \mathbb{IR} .

Un intervalle est caractérisé par sa longueur $w([x]) = \bar{x} - \underline{x}$ et par son centre $m([x]) = \frac{(\underline{x} + \bar{x})}{2}$.

II.1.2 Pavé et sous-pavage

Définition II.2 (Pavé) *Un pavé $[x]$ (ou boîte) de \mathbb{R}^n est le produit cartésien de n intervalles pouvant s'écrire sous la forme suivante :*

$$[x] = [\underline{x}_1, \bar{x}_1] \times \dots \times [\underline{x}_n, \bar{x}_n] = [x_1] \times \dots \times [x_n]. \quad (\text{II.2})$$

Un pavé $[x]$ est aussi appelé un vecteur intervalle car ses n composantes sont des intervalles de \mathbb{IR} . L'ensemble de tous les pavés de \mathbb{R}^n est noté \mathbb{IR}^n .

La longueur $w([x])$ d'un pavé $[x]$ correspond à la longueur de son plus grand côté :

$$w([x]) = \max\{\bar{x}_i - \underline{x}_i, i = 1, \dots, n\}. \quad (\text{II.3})$$

Par exemple,

$$w([-2, 2] \times [1, 3]) = 4.$$

La **bissection** d'un pavé $[x]$ est l'opération qui consiste à couper le pavé en deux suivant l'axe de symétrie perpendiculaire à son plus grand côté. Par exemple, la bissection du pavé $[x] = [-2, 2] \times [1, 3]$ génère deux nouvelles boîtes :

$$[x](1) = [-2, 0] \times [1, 3] \text{ et } [x](2) = [0, 2] \times [1, 3].$$

Un **sous-pavage** $\widehat{\mathcal{P}}$ de \mathbb{R}^n est une union de pavés de \mathbb{IR}^n qui ne se chevauchent pas, ce qui signifie que leur intersection deux à deux est vide sauf éventuellement s'ils ont une frontière commune.

Un sous-pavage de $[x]$ est **régulier** si chacun de ses pavés peut être obtenu par bisections successives de $[x]$.

Lorsqu'un sous-pavage $\widehat{\mathcal{P}}$ recouvre tout le pavé $[x]$, il est appelé **pavage** de $[x]$.

II.2 Calcul par intervalles

Après avoir introduit les variables sur lesquelles sont basées l'analyse par intervalles, nous présentons des formulations mathématiques permettant de les manipuler. De nombreux exemples mettant en oeuvre le calcul par intervalles sont étudiés dans [JAU 01] et dans [JAU 04].

II.2.1 Arithmétique par intervalles

Les opérations sur les nombres réels peuvent être étendues aux intervalles par la formule suivante :

$$[x] \diamond [y] = [\{x \diamond y \mid x \in [x], y \in [y]\}] \quad (\text{II.4})$$

avec $\diamond \in \{+, -, *, /\}$.

Soient $[x]$ et $[y]$, deux intervalles de \mathbb{R} , la caractérisation de $[x] \diamond [y]$ peut se faire à l'aide des formules suivantes :

$$\begin{aligned} [x] + [y] &= [\underline{x} + \underline{y}, \bar{x} + \bar{y}] \\ [x] - [y] &= [\underline{x} - \bar{y}, \bar{x} - \underline{y}] \\ [x] * [y] &= [\min(\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}), \max(\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y})] \\ [x] / [y] &= [x] * 1/[y] \\ 1 / [y] &= \begin{cases} [-\infty, \infty] & \text{si } 0 \in [y], \\ [1/\bar{y}, 1/\underline{y}] & \text{sinon.} \end{cases} \end{aligned} \quad (\text{II.5})$$

Exemple II.1 On considère deux intervalles de \mathbb{R} , $[x] = [1, 3]$ et $[y] = [-1, 2]$:

$$\begin{aligned} [x] - [y] &= [1, 3] - [-1, 2] = [-1, 4], \\ [x] * [y] &= [1, 3] * [-1, 2] = [-3, 6], \\ [x] / [y] &= [1, 3] * 1/[-1, 2] = [-\infty, +\infty] \text{ car } 0 \in [y]. \end{aligned}$$

II.2.2 Fonctions élémentaires

De façon plus générale, il est possible d'étendre les fonctions élémentaires réelles aux intervalles en utilisant une fonction intervalle donnée par la formule suivante :

$$f([x]) = [\inf_{x \in [x]} f(x), \sup_{x \in [x]} f(x)]. \quad (\text{II.6})$$

Il est donc aisé d'obtenir les expressions des différentes fonctions élémentaires monotones :

$$\begin{aligned}
 \exp([x]) &= [\exp(\underline{x}), \exp(\bar{x})]. \\
 \log([x]) &= \begin{cases} [-\infty, \log(\bar{x})] \text{ si } 0 \in [x], \\ \emptyset \text{ si } \bar{x} \leq 0, \\ [\log(\underline{x}), \log(\bar{x})] \text{ sinon.} \end{cases} \\
 [x^2] &= \begin{cases} [0, \max(\underline{x}^2, \bar{x}^2)] \text{ si } 0 \in [x], \\ [\min(\underline{x}^2, \bar{x}^2), \max(\underline{x}^2, \bar{x}^2)] \text{ sinon.} \end{cases} \\
 \sqrt{[x]} &= \begin{cases} [-\sqrt{\bar{x}}, \sqrt{\underline{x}}] \text{ si } \bar{x} \geq 0, \\ \emptyset \text{ sinon.} \end{cases}
 \end{aligned} \tag{II.7}$$

D'autres fonctions, comme les fonctions sinus et cosinus qui sont monotones par morceaux, ne peuvent pas s'écrire à l'aide d'expressions simples et nécessitent de développer des algorithmes spécifiques.

Exemple II.2

$$\begin{aligned}
 \exp([0,1]) &= [1,e] \\
 \log([-2, -1]) &= \emptyset \\
 [-2,2]^2 &= [0,4] \\
 \sqrt{[-2,4]} &= [0,2] \\
 \text{abs}([-2,1]) &= [0,2] \\
 \sin([0, \frac{2\pi}{3}]) &= [0,1] \\
 \cos([0, \frac{2\pi}{3}]) &= [-\frac{1}{2}, 1]
 \end{aligned}$$

II.3 Fonction d'inclusion

La fonction intervalle $[f]$ de \mathbb{IR} dans \mathbb{IR} est appelée fonction d'inclusion de f si et seulement si,

$$\forall [x] \in \mathbb{IR}, \quad f([x]) \subseteq [f]([x]). \tag{II.8}$$

Quand la relation d'inclusion (II.8) devient une égalité, la fonction d'inclusion est dite minimale.

Pour une même fonction f , il existe différentes fonctions d'inclusion $[f]$. Parmi ces fonctions d'inclusion, il est intéressant de rechercher des fonctions d'inclusion convergentes. Une fonction d'inclusion est convergente si et seulement si, pour un pavé $[x]$ de \mathbb{R}^n :

$$\lim_{w([x]) \rightarrow 0} w([f]([x])) = 0, \tag{II.9}$$

où $w([x])$ est la longueur du pavé $[x]$.

Différentes méthodes peuvent être mises en oeuvre pour obtenir une fonction d'inclusion convergente. La méthode la plus simple consiste à remplacer dans l'expression les nombres par les intervalles auxquels ils appartiennent et les fonctions élémentaires réelles par leur extension aux intervalles. La fonction d'inclusion obtenue est la **fonction d'inclusion naturelle**.

Exemple II.3 *Considérons la fonction*

$$f(x) = x^2 + 3x + 2. \tag{II.10}$$

Une fonction d'inclusion naturelle de f est donnée par

$$[f]([x]) = [x] * [x] + 3 * [x] + 2. \tag{II.11}$$

En évaluant f pour $x = [-2,1]$, on obtient

$$[f]([-2,1]) = [-2,1] * [-2,1] + 3 * [-2,1] + 2 = [-2,4] + [-4,5] = [-6,9]. \tag{II.12}$$

Remarquons que l'image de l'intervalle $[x]$ par f est $f([-2,1]) = [-4,4]$. La relation d'inclusion (II.8) est bien vérifiée.

Exemple II.4 *Soit la fonction*

$$\begin{aligned} f : \quad \mathbb{R}^3 &\rightarrow \mathbb{R}^2 \\ (x_1, x_2, x_3) &\rightarrow (x_1^2, x_2 x_3). \end{aligned}$$

sa fonction d'inclusion naturelle est

$$\begin{aligned} [f] : \quad \mathbb{I}\mathbb{R}^3 &\rightarrow \mathbb{I}\mathbb{R}^2 \\ ([x_1], [x_2], [x_3]) &\rightarrow ([x_1]^2, [x_2] * [x_3]). \end{aligned}$$

Si, par exemple, $[x] = [x_1] \times [x_2] \times [x_3] = [-1,1] \times [1,2] \times [0,1]$, alors $[f]([x_1], [x_2]) = [0,1] \times [0,6]$. Comme les variables intervalles x_i n'apparaissent qu'une seule fois dans chaque expression f_i , la fonction d'inclusion naturelle est minimale, pour $i = 1, 2, 3$.

La fonction d'inclusion naturelle est rarement minimale. Un pessimisme est souvent introduit par celle-ci provenant du fait que chaque occurrence d'une variable intervalle est considérée comme variant indépendamment des autres.

Reprenons l'exemple qu'avait traité M. Kieffer dans sa thèse [KIE 99] pour comparer les performances des fonctions d'inclusion à partir de quatre formulations d'une même fonction.

Exemple II.5

$$\begin{aligned} f_1(x) &= x(x+1), \\ f_2(x) &= x.x+x, \\ f_3(x) &= x^2+x, \\ f_4(x) &= (x+\frac{1}{2})^2 - \frac{1}{4}. \end{aligned}$$

Pour $[x] = [-1,1]$, on obtient

$$\begin{aligned} [f_1]([x]) &= [x]([x]+1) = [-2,2], \\ [f_2]([x]) &= [x] * [x] + [x] = [-2,2], \\ [f_3]([x]) &= [x]^2 + [x] = [-1,2], \\ [f_4]([x]) &= ([x] + \frac{1}{2})^2 - \frac{1}{4} = [-\frac{1}{4},2]. \end{aligned}$$

*Ainsi les expressions $[x] * [x]$ et $[x]^2$ ne donnent pas le même résultat, car la variable intervalle $[x]$ intervient deux fois dans la première expression. Chaque occurrence de $[x]$ peut donc varier indépendamment, ce qui introduit un pessimisme dans l'évaluation de la fonction f . La quatrième formulation fournit le plus petit intervalle solution, ainsi $[f_4]$ est minimale.*

La longueur des intervalles obtenus dépend donc assez fortement du nombre d'occurrences de chaque variable dans l'expression initiale de la fonction f .

Pour réduire ce pessimisme, il faut réécrire l'expression de manière à réduire le nombre d'occurrences de chaque variable. Si chacune des variables n'intervient qu'une seule fois dans l'expression de f alors la fonction d'inclusion naturelle sera minimale. Comme il n'existe pas de méthode systématique pour trouver la fonction d'inclusion minimale, nous utiliserons par la suite la fonction d'inclusion naturelle qui reste la plus simple à calculer.

Le pessimisme introduit dans l'évaluation des fonctions peut être réduit en utilisant des développements en série de Taylor. Ainsi, lorsque les intervalles $[x]$ sont de faible

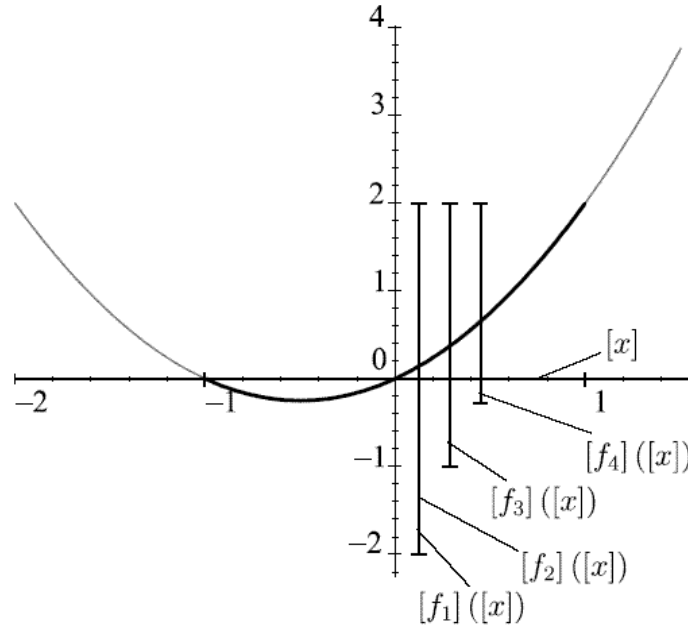


FIG. II.1 – Comparaison des différentes fonctions d'inclusion d'une même fonction

longueur, on peut utiliser la forme centrée de la fonction d'inclusion décrite.

Soit une fonction $[f] : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est différentiable et à dérivée continue sur le domaine $D \subset \mathbb{R}^n$ alors

$$\forall x \in D, \quad f(x) \in [f_m]([x]) = [f](m) + [f']([x])([x] - m) \quad (\text{II.13})$$

m est le centre du pavé $[x]$ et $[f']([x])$ est la fonction d'inclusion de la jacobienne de la fonction f .

Cette fonction d'inclusion centrée fournit des intervalles solutions généralement moins pessimistes que dans la fonction d'inclusion naturelle mais elle nécessite que $[x]$ soit de faible longueur.

II.4 Image réciproque d'un ensemble

Pour calculer l'image réciproque d'un ensemble [JAU 03], il est nécessaire de résoudre un problème d'inversion ensembliste en caractérisant l'ensemble \mathbb{X} défini par

$$\mathbb{X} = \{x \in \mathbb{R}^n | f(x) \in \mathbb{Y}\} = f^{-1}(\mathbb{Y}), \quad (\text{II.14})$$

où f est une fonction connue continue de \mathbb{R}^n dans \mathbb{R}^m ayant une fonction d'inclusion convergente $[f](\cdot)$, et où \mathbb{Y} est un ensemble connu inclus dans \mathbb{R}^m .

Ce problème d'inversion ensembliste peut être résolu de manière approximative mais garantie par l'analyse par intervalles et l'algorithme SIVIA (*Set Inversion Via Interval Analysis*) développé par Luc Jaulin [JAU 93b]. Cet algorithme permet la construction de deux sous-pavages $\underline{\mathbb{X}}$ et $\overline{\mathbb{X}}$ qui vérifient les inclusions suivantes :

$$\underline{\mathbb{X}} \subset \mathbb{X} \subset \overline{\mathbb{X}}. \quad (\text{II.15})$$

L'algorithme SIVIA , qui permet d'obtenir les deux sous-pavages précédents avec une précision requise ε , se compose essentiellement de trois étapes :

- le choix d'un pavé initial \mathbb{X}_0 supposé contenir toutes les solutions recherchées,
- les tests d'inclusion,
- la bisection.

II.4.1 Tests d'inclusion

Un test d'inclusion consiste à vérifier l'appartenance d'un intervalle obtenu par une fonction d'inclusion à un ensemble connu a priori.

Pour un pavé quelconque $[x]$, trois situations sont envisageables :

- $[f]([x]) \subset \mathbb{Y} \rightarrow [x] \subset \mathbb{X}$ et $[x]$ est dit acceptable.
- $[f]([x]) \cap \mathbb{Y} = \emptyset \rightarrow [x] \cap \mathbb{X}$ et $[x]$ est dit mauvais.
- $[f]([x]) \cap \mathbb{Y} \neq \emptyset \rightarrow [x]$ est dit incertain (ou indéterminé).

II.4.2 Bisection

Bisecter un pavé signifie couper le pavé en deux selon l'axe perpendiculaire à celui de son plus grand côté. Lorsque qu'un pavé est dit incertain (ou encore ambigu ou indéterminé), une bisection est réalisé sur celui-ci. Les deux nouveaux pavés générés sont à leur tour testés. Si ceux-ci sont à nouveau incertains, on les bissecte. On recommence cette opération jusqu'à ce qu'il ne reste plus de pavé incertain ou jusqu'à ce que les pavés restant à analyser aient une longueur inférieure à la précision requise ε que l'on aura fixé afin d'arrêter l'algorithme.

La solution de notre problème d'inversion ensembliste sera donnée par l'union des pavés acceptables et des pavés incertains jugés trop petits (ayant une longueur inférieure à la limite de précision).

II.4.3 Algorithme SIVIA

Nous utilisons l'algorithme SIVIA pour déterminer les sous-ensembles $\underline{\mathbb{X}}$ contenant les pavés solutions et $\overline{\mathbb{X}}$ contenant les pavés admissibles et les pavés indéterminés de petite

taille. Cet algorithme utilise des piles : une pile \mathcal{P}_{int} dans laquelle sont rangés les intervalles à analyser (les pavés intermédiaires) et une pile \mathcal{P}_{ad} servant à stocker les pavés acceptables ou incertains mais de petite taille. Le pavé initial $[x](0) \in \mathbb{X}_0$ supposé contenir toutes les solutions recherchées est placé dans la pile \mathcal{P} . Lors du déroulement de l'algorithme, le premier élément de la pile est récupéré dans un pavé $[x]$. Cette opération consiste donc à extraire l'élément au sommet de la pile. Son image par la fonction d'inclusion $[f]$ est ensuite testée dans l'ensemble connu a priori \mathbb{Y} . Si le pavé $[x]$ est indéterminé et que sa longueur est supérieure à ε , il est bissecté. Les deux pavés générés $[x_1]$ et $[x_2]$ sont ensuite rangés dans au sommet de la pile \mathcal{P}_{int} . Si le pavé $[x]$ est acceptable, la pile \mathcal{P}_{ad} est augmentée de $[x]$.

Algorithme II.1 (SIVIA)

Entrées : $f, [x](0), \mathbb{Y}, \varepsilon$;

Sorties : $\mathbb{X}, \overline{\mathbb{X}}$;

1. *initialisation* : $\mathcal{P}_{ad} := \emptyset, \mathcal{P}_{int} := [x](0)$;
2. $[x] := \text{ExtraireSommet}(\mathcal{P}_{int})$;
3. *si* $[f]([x]) \subset \mathbb{Y}$ *alors* $\mathcal{P}_{ad} := \mathcal{P}_{ad} \cup [x]$, *puis aller en 6*;
4. *si* $[f]([x]) \cap \mathbb{Y} \neq \emptyset$ *et* $w([x]) < \varepsilon$ *alors* \mathcal{P}_{ad} , *puis aller en 6*;
5. *si* $[f]([x]) \cap \mathbb{Y} \neq \emptyset$ *et* $w([x]) > \varepsilon$ *alors bissection de* $[x]$;
 $\mathcal{P}_{int} := \mathcal{P}_{int} \cup [x_1], \mathcal{P}_{int} := \mathcal{P}_{int} \cup [x_2]$, *puis aller en 6*;
6. *si* $\mathcal{P}_{int} \neq \emptyset$, *aller en 2*.

II.5 Image directe d'un ensemble

Le calcul de l'image directe d'un ensemble ou d'un sous-pavé \mathbb{X} par une fonction f est plus difficile que celui de l'image réciproque d'un ensemble.

Il consiste à caractériser l'ensemble

$$\mathbb{Y} = \{y \in \mathbb{R}^m \mid f(\mathbb{X})\}, \quad (\text{II.16})$$

où \mathbb{X} est un ensemble connu, f est une fonction connue continue et $[f]$ est une fonction d'inclusion convergente de f .

Un algorithme basé sur l'analyse par intervalles et l'inversion ensembliste permet la caractérisation de cet ensemble. C'est l'algorithme ImageSP développé par Luc Jaulin [JAU 04]. L'algorithme ImageSP construit un sous-pavage régulier $\overline{\mathbb{Y}}$ de \mathbb{Y} qui contient

l'image par la fonction d'inclusion $[f]$ d'un sous-pavage régulier \mathbb{X}_ε de \mathbb{X} , donc

$$\begin{aligned} \mathbb{Y} &= f(\mathbb{X}) \subset \overline{\mathbb{Y}} \\ \overline{\mathbb{Y}} &\subset [f](\mathbb{X}). \end{aligned} \tag{II.17}$$

Cet algorithme se décompose essentiellement en trois étapes :

1. Le hachage consiste à construire, à partir de bisections successives des pavés de \mathbb{X} , un pavage régulier non minimal \mathbb{X}_ε qui ne contient que des pavés dont la longueur est inférieure à une précision requise ε , avec $\varepsilon > 0$ et suffisamment petit.
2. L'évaluation calcule l'image par la fonction $[f]$ de tous les pavés $[x]$ contenus dans \mathbb{X}_ε . Toutes les boîtes résultantes sont alors stockées dans une pile \mathcal{U} .
3. La régularisation calcule un sous-pavage régulier $\overline{\mathbb{Y}}$ contenant l'ensemble \mathbb{U} , l'union de tous les pavés de la pile \mathcal{U} . La régularisation fait appel à l'algorithme SIVIA qui permet d'inverser l'ensemble \mathbb{U} par la fonction identité . En effet si $f(\mathbb{X}) \subset \mathbb{U}$, alors $f(\mathbb{X}) \subset \text{Id}^{-1}\mathbb{U}$, on se ramène donc à un problème d'inversion ensembliste.

L'algorithme ImageSP [JAU 04] permet donc de calculer un sous-pavage régulier $\overline{\mathbb{Y}}$ garanti de contenir l'ensemble \mathbb{Y} .

Algorithme II.2 (ImageSP)

Entrées : $f, \mathbb{X}, \varepsilon$;

Sorties : $\overline{\mathbb{Y}}$;

1. $\mathbb{X}_\varepsilon := \text{hachage}(\mathbb{X}, \varepsilon)$;
2. $\mathcal{U} := \emptyset$, \mathcal{U} est une pile et \mathbb{U} est l'ensemble des pavés contenus dans \mathcal{U}
3. Pour chaque $[x] \in \mathbb{X}_\varepsilon$, on ajoute $[f]([x])$ à la pile \mathcal{U}
4. SIVIA $(([y](0), \text{Id}, \mathbb{U})$.

L'analyse par intervalles permet la prise en compte des incertitudes (de mesures ou autres). Elle donne des résultats garantis en calculant un intervalle contenant à coup sûr la solution. Par contre, nous avons vu que la longueur des intervalles calculés par les méthodes ensemblistes énoncées dans ce chapitre dépend fortement de l'expression initiale de la fonction à évaluer. C'est un inconvénient de l'analyse par intervalles. Pour plus de détails sur ces méthodes ensemblistes, le lecteur intéressé pourra se référer aux ouvrages de Luc Jaulin [JAU 99][JAU 01] et au site internet du groupe de travail MEA du GDR MACS (<http://www.lag.ensieg.inpg.fr/gt-ensembliste>).

Chapitre III

Intégration numérique garantie des équations différentielles ordinaires

Nous allons présenter une méthode d'intégration garantie basée sur l'analyse par intervalles et les développements en série de Taylor [COR 94][NED 99][RAM 05]. Cette méthode est dite garantie dans le sens où les résultats sont fournis sous forme d'intervalles contenant à coup sûr la solution.

III.1 Intégration numérique garantie basée sur les séries de Taylor

Considérons un système dynamique autonome non linéaire dont la représentation d'état est donnée par les équations suivantes :

$$\begin{cases} \dot{x}(t) = f(x(t)) \\ y(t) = g(x(t)) \\ x(t_0) \in [x_0] \end{cases} \quad (\text{III.1})$$

où $t \in [t_0, T]$ pour $T > t_0$, $f, g \in C^{k-1}(\mathbb{D})$, $\mathbb{D} \subset \mathbb{R}^n$ est un domaine ouvert. Les fonctions $f : \mathbb{D} \rightarrow \mathbb{R}^n$ et $g : \mathbb{D} \rightarrow \mathbb{R}^r$ peuvent être linéaires ou non linéaires, mais continûment différentiables $k - 1$ fois sur \mathbb{D} .

La condition initiale $x(t_0)$ est inconnue mais elle est supposée appartenir à un domaine connu a priori \mathbb{X}_0 . Les sorties $y(t)$ sont calculées aux instants $t_i \in \{t_1, \dots, t_N\} \subseteq [t_0, T]$ et

sont notées $[y_j]$ pour $j = 1, \dots, N$.

Remarque III.1 *Nous ne considérons que les systèmes autonomes. Mais ce choix n'est pas restrictif. En effet, un système non autonome:*

$$\dot{x}(t) = f(x(t), t), x(t_0) \in [x_0], \quad (\text{III.2})$$

peut être transformé en un système autonome en construisant un vecteur d'état étendu :

$$x^e = \begin{bmatrix} x \\ t \end{bmatrix} \text{ avec } \dot{x}_{(n+1)}^e = 1, \text{ et } x_{(n+1)}^e(t_0) = t_0, \quad (\text{III.3})$$

où n est la dimension du vecteur d'état du système non autonome.

Remarque III.2 *Dans le cas d'un système qui dépend d'un vecteur de paramètres inconnu p que nous cherchons à estimer:*

$$\dot{x}(t, p) = f(x(t), p), x(t_0) \in [x_0], p \in \mathbb{P}, \quad (\text{III.4})$$

nous pouvons aussi considérer un vecteur d'état étendu

$$x^e = \begin{bmatrix} x \\ p \end{bmatrix} \text{ avec } \dot{x}_{(n+i)}^e = 0, \text{ et } x_{(n+i)}^e(t_0) \in \mathbb{P}_i, \text{ avec } i = 1, \dots, n_p \quad (\text{III.5})$$

où n_p est la dimension du vecteur de paramètres à estimer.

Nous souhaitons calculer des vecteurs intervalles $[y_j]$ pour $j = 1, \dots, N$ qui garantissent contenir la solution des équations d'état (III.1) à chaque instant d'intégration $t_i \in \{t_1, \dots, t_N\} \subseteq [t_0, T]$.

Une méthode d'intégration garantie calcule des bornes pour la solution exacte $y(t)$ et vérifie l'existence d'une unique solution au problème. Nous utilisons une méthode basée sur les séries de Taylor pour intégrer de manière garantie l'équation différentielle ordinaire :

$$\dot{x}(t) = f(x(t)), x(t_0) \in [x_0] \quad (\text{III.6})$$

où la solution $x(t_j)$ à l'instant t_j sera notée x_j .

Le développement en série de Taylor à l'ordre k pour un point solution x_{j+1} à l'instant t_{j+1} connaissant la solution x_j à l'instant t_j s'écrit:

$$x_{j+1} = x_j + \sum_{i=1}^{k-1} h^i f^{[i]}(x_j) + h^k f^{[k]}(x(t_\xi)), \quad (\text{III.7})$$

où $x(t_\xi)$ est la solution à $t_\xi \in [t_j, t_{j+1}]$ et $h = t_{j+1} - t_j$ est le pas de discrétisation choisi pour le modèle. Nous l'avons choisi constant pour intégrer notre modèle mais il n'est pas nécessaire qu'il le soit.

Les coefficients de Taylor $f^{[i]}$ sont calculés à partir de la formule récursive suivante :

$$f^{[i]} = \frac{1}{i!} \frac{\partial^{(i-1)} f}{\partial x} f = \frac{1}{i!} x^{[i]}, i \geq 1. \quad (\text{III.8})$$

Nous pouvons modifier le pas d'intégration h sans recalculer les coefficients de Taylor. La formule du développement de Taylor peut être étendue aux intervalles de la manière suivante :

$$[x_{j+1}] = [x_j] + \sum_{i=1}^{k-1} h^i f^{[i]}([x_j]) + h^k f^{[k]}(\xi). \quad (\text{III.9})$$

Pour simplifier les notations, les fonctions d'inclusion $[f]$ sont seulement notées f . Pour avoir une solution garantie, il nous faut trouver un ensemble contenant le terme de reste dans (III.7), c'est à dire $x(t_\xi)$. Nous prendrons l'ensemble $[\xi]$ contenant toutes les trajectoires entre un instant t_j et un instant t_{j+1} . Nous verrons par la suite comment cet ensemble que l'on peut considérer comme une solution a priori du problème III.1 peut être calculé.

La méthode d'intégration garantie basée sur les séries de Taylor se décompose essentiellement en deux étapes :

1. la recherche d'une solution a priori qui nous garantit l'existence et l'unicité de la solution de l'équation différentielle ordinaire,
2. la réduction de la solution par des formulations spécifiques des développements en série de Taylor.

III.2 Existence et unicité de la solution

Pour prouver l'existence et l'unicité d'une solution, nous utilisons une méthode basée sur le théorème du point fixe (dit aussi théorème de Brouwer) et sur l'opérateur de Picard-Lindelöf que nous définissons. Cette méthode est appelée *the constant enclosure method*.

III.2.1 Théorème du point fixe

Théorème III.1 (du point fixe) *Soit \mathbb{X} , un espace métrique complet non vide muni d'une distance $d(\dots)$. Si une fonction $\phi : \mathbb{X} \rightarrow \mathbb{X}$ satisfait la condition de Lipschitz :*

$$d(\phi(x), \phi(y)) \leq \gamma d(x, y) \quad (\text{III.10})$$

$\forall x, y \in \mathbb{X}$ et $\gamma \in [0,1[$, alors la fonction ϕ possède un et un seul point fixe $y^* \in \mathbb{X}$ défini par

$$\phi(y^*) = y^*. \quad (\text{III.11})$$

III.2.2 Opérateur de Lindelöf

Définition III.1 Soit $u(\cdot)$, une fonction appartenant à \mathbb{U} l'ensemble des fonctions continues définies sur $[t_j, t_{j+1}]$, l'opérateur ϕ défini par

$$\phi(u(t)) = u(t_j) + \int_{t_j}^t f(u(\tau)) d\tau \quad (\text{III.12})$$

est appelé opérateur de Picard-Lindelöf. f est la fonction non linéaire du système d'équations (III.1) et $t \in [t_j, t_{j+1}]$.

Proposition III.1 Soit u^* , le point fixe de l'opérateur de Picard-Lindelöf ϕ :

$$\phi(u^*(t)) = u^*(t) \Leftrightarrow u^*(t) = u^*(t_j) + \int_{t_j}^t f(u^*(\tau)) d\tau. \quad (\text{III.13})$$

En dérivant $u^*(t)$ par rapport au temps, nous obtenons

$$\frac{d}{dt}(\phi(u^*(t))) = \dot{u}^*(t) \Leftrightarrow \dot{u}^*(t) = f(u^*(t)). \quad (\text{III.14})$$

Il vient d'être démontré que le point fixe de la fonction ϕ est la solution de l'équation différentielle ordinaire (III.6).

III.2.3 Recherche d'une solution a priori

Pour avoir une solution garantie, il faut trouver l'ensemble $[\xi]$ contenant toutes les trajectoires entre un instant t_j et un instant t_{j+1} . Cet ensemble $[\xi]$ contient l'ensemble des points fixes de ϕ entre t_j et t_{j+1} si et seulement si,

$$\phi([\xi]) \subseteq [\xi] \quad (\text{III.15})$$

Cet ensemble sera considéré comme une solution a priori de notre équation différentielle à l'instant t_{j+1} .

Soient $h_j = t_{j+1} - t_j$ le pas de discrétisation de valeur positive et $[\tilde{x}_j] \subseteq \mathbb{D}$ tels que :

$$[x_j] + h_j f([\tilde{x}_j]) \subseteq [\tilde{x}_j]. \quad (\text{III.16})$$

Alors, le pavé $[\tilde{x}_j]$ contient toutes les trajectoires $x(t)$ solutions de (III.1) entre t_j et t_{j+1} .

Démonstration Soit \mathbb{U} l'ensemble de toutes les fonctions continues sur $[t_j, t_{j+1}]$ et dont les valeurs sont comprises dans le pavé $[\tilde{x}_j]$:

$$\mathbb{U} = \{u \mid u \in C^0([t_j, t_{j+1}]) \text{ et } u(t) \in [\tilde{x}_j] \text{ pour } t \in [t_j, t_{j+1}]\}. \quad (\text{III.17})$$

En appliquant l'opérateur de Picard-Lindelöf et en utilisant la relation d'inclusion (III.16), nous obtenons :

$$\begin{aligned} \phi(u(t)) &= x_j + \int_{t_j}^t f(u(\tau)) \, d\tau \\ &\in x_j + \int_{t_j}^t f([\tilde{x}_j]) \, d\tau \\ &\subseteq x_j + hf([\tilde{x}_j]) \\ &\subseteq [\tilde{x}_j] \end{aligned} \quad (\text{III.18})$$

Nous rappelons que par abus de notation, nous écrivons la fonction d'inclusion $[\phi]$ et la fonction ϕ de façon similaire. Nous remarquons que $\forall u \in \mathbb{U}$, $\phi(u(t)) \in [\tilde{x}_j]$ et comme $\phi(u(t)) \in C^0([t_j, t_{j+1}])$, alors :

$$\phi(\mathbb{U}) \subseteq \mathbb{U}. \quad (\text{III.19})$$

L'opérateur de Picard-Lindelöf possède un seul point fixe dans \mathbb{U} . Nous avons vu que le point fixe de l'opérateur de Picard-Lindelöf est solution de l'équation différentielle (III.1) pour $t \in [t_j, t_{j+1}]$ avec x_j comme condition initiale. Nous pouvons donc en déduire que le pavé $[\tilde{x}_j]$ contient toutes les trajectoires de la solution $x(t)$ pour $t \in [t_j, t_{j+1}]$.

Si l'état initial x_j n'est pas connu mais appartient à un ensemble $[X_j]$ et si la relation d'inclusion suivante est vérifiée :

$$[x_j] + hf([\tilde{x}_j]) \subseteq [\tilde{x}_j], \quad (\text{III.20})$$

alors nous pouvons dire que le pavé $[\tilde{x}_j]$ contient de manière garantie la trajectoire de l'équation différentielle (III.6) quelque soit la condition initiale $x_j \in [x_j]$. Nous rappelons que le pas d'intégration h est ici considéré constant. S'il est possible de calculer un pavé $[\tilde{x}_j]$ vérifiant l'inclusion (III.16), alors l'équation différentielle (III.1) possède une solution unique à chaque instant de $t \in [t_j, t_{j+1}]$.

Pour calculer le pavé $[\tilde{x}_j]$, il faut commencer par calculer l'expression suivante :

$$[\check{x}_j] = [x_j] + hf([x_j]). \quad (\text{III.21})$$

En pratique, cette expression ne contiendra pas la trajectoire solution. L'intervalle $[\check{x}_j]$ est alors agrandi jusqu'à satisfaire la relation d'inclusion (III.16), le pavé $[\check{x}_j]$ sera alors noté $[\tilde{x}_j]$. Le pavé $[\tilde{x}_j]$ contiendra de manière garantie la trajectoire de $x(t)$ et sera considéré comme une solution a priori de notre équation différentielle ordinaire (III.1).

L'algorithme *Enclosure* calcule $[\tilde{x}_j]$ et agrandit l'intervalle si celui-ci ne vérifie pas l'inclusion (III.16).

Algorithme III.1 (Enclosure)

Entrées : $[x_j]$, h , α ;

Sorties : $[\tilde{x}_j]$;

1. *Initialisation* : $[\tilde{x}_j] := [x_j] + hf([x_j])$;
2. *tant que* $([x_j] + hf([\tilde{x}_j]) \not\subseteq [\tilde{x}_j])$
3. $[\tilde{x}_j] := [\tilde{x}_j] + [-\alpha, +\alpha][|\tilde{x}_j|]$.

où $|\tilde{x}_j| = \max(|\underline{\tilde{x}}_j|, |\overline{\tilde{x}}_j|)$ représente la norme du vecteur intervalle $[\tilde{x}_j]$.

Le coefficient de gonflement α doit être suffisamment faible pour que le pavé $[\tilde{x}_j]$ soit à peine plus gros que celui obtenu à droite de l'expression au pas 2 de l'algorithme, mais il ne doit pas être nul, car sinon la condition d'inclusion (III.16) de cette expression ne sera jamais vérifiée. Le temps de calcul dépend du choix de α et sera d'autant plus long que α sera petit.

Une version plus complexe de l'algorithme pourrait permettre le calcul d'un pas d'intégration optimal en même temps que celui de la solution a priori $[\tilde{x}_j]$.

La recherche d'une solution a priori consiste en fait à effectuer un développement de Taylor au premier ordre. Cette approximation est connue pour son pessimisme. La longueur de l'intervalle $[\tilde{x}_j]$ augmente avec j . Pour réduire ce pessimisme, des méthodes basées sur la forme centrée des coefficients de Taylor vont être mises en oeuvre.

III.3 Réduction de la solution

L'ensemble $[\xi]$ contenant toutes les trajectoires entre un instant t_j et un instant t_{j+1} peut être considéré comme une solution a priori à l'instant t_{j+1} de notre équation différentielle ordinaire pouvant être calculée avec l'algorithme III.1. La formule de Taylor étendue aux intervalles (III.9) devient donc :

$$[x_{j+1}] = [x_j] + \sum_{i=1}^{k-1} h^i f^{[i]}([x_j]) + h^k f^{[k]}([\tilde{x}_j]). \quad (\text{III.22})$$

III.3.1 Méthode directe

La formule de Taylor étendue aux intervalles divergera après quelques instants d'intégration. En effet, en considérant la longueur de l'intervalle $[x_{j+1}]$ et par linéarité de l'opérateur $w(\cdot)$, nous obtenons :

$$w([x_{j+1}]) = w([x_j]) + \sum_{i=1}^{k-1} h^i w(f^{[i]}([x_j])) + h^k w(f^{[k]}([\tilde{x}_j])) \geq w([x_j]). \quad (\text{III.23})$$

Comme le pas d'intégration h est positif et que la longueur d'un intervalle défini par l'opérateur $w(\cdot)$ l'est aussi, nous remarquons que la taille de l'intervalle solution augmente

à chaque pas d'intégration.

Pour éviter ce phénomène, nous utilisons la forme centrée des coefficients de Taylor donnée par la formule

$$f^{[i]}([x_j]) = f^{[i]}(\hat{x}_j) + J(f^{[i]}, [x_j])([x_j] - \hat{x}_j), \quad (\text{III.24})$$

où $J(f^{[i]}, [x_j])$ est une fonction d'inclusion de la jacobienne du i^{eme} coefficient de Taylor et \hat{x}_j est le point milieu du pavé $[x_j]$.

En remplaçant la nouvelle expression des coefficients dans la formule du développement de Taylor étendu aux intervalles (III.22), nous obtenons l'expression :

$$[x_{j+1}] = [x_j] + \sum_{i=1}^{k-1} \{h^i f^{[i]}(\hat{x}_j) + h^i J(f^{[i]}, [x_j])([x_j] - \hat{x}_j)\} + h^k f^{[k]}([\tilde{x}_j]). \quad (\text{III.25})$$

qui peut aussi s'écrire

$$[x_{j+1}] = \hat{x}_j + \sum_{i=1}^{k-1} h^i f^{[i]}(\hat{x}_j) + h^k f^{[k]}([\tilde{x}_j]) + \{I + \sum_{i=1}^{k-1} h^i J(f^{[i]}, [x_j])\}([x_j] - \hat{x}_j) \quad (\text{III.26})$$

Cette expression peut être calculée en plusieurs étapes à l'aide de la méthode de la valeur moyenne (appelée aussi la méthode directe) détaillée dans l'algorithme qui suit.

Algorithme III.2 (Mean Value)

Entrées : $[x_j]$, h , $[\tilde{x}_j]$;

Sorties : $[x_{j+1}]$, \hat{x}_{j+1} ;

1. $\hat{x}_j = m([x_j])$;
2. $[v_{j+1}] = \hat{x}_j + \sum_{i=1}^{k-1} h^i f^{[i]}(\hat{x}_j) + h^k f^{[k]}([\tilde{x}_j])$;
3. $[S_j] = \{I + \sum_{i=1}^{k-1} h^i J(f^{[i]}, [x_j])\}$;
4. $[x_{j+1}] = [v_{j+1}] + [S_j]([x_j] - \hat{x}_j)$;
5. $\hat{x}_{j+1} = m([v_{j+1}]) = m([x_{j+1}])$.

L'égalité $m([v_{j+1}]) = m([x_{j+1}])$ est vérifiée car le vecteur $[S_j]([x_j] - \hat{x}_j)$ est symétrique. Le produit de la matrice intervalle $[S_j]$ par le vecteur intervalle $([x_j] - \hat{x}_j)$ au pas 4 de l'algorithme provoque un effet d'enveloppement du pavé solution $[x_{j+1}]$. Lorsque que nous utilisons l'algorithme avec des vecteurs intervalles, l'effet d'enveloppement (aussi appelé le *wrapping effect*) provoque une augmentation de la taille du pavé solution à chaque instant d'intégration.

III.3.2 Effet d'enveloppement

Avant d'expliquer pourquoi cet effet d'enveloppement apparaît dans la méthode de la valeur moyenne, nous présentons, à l'aide d'un exemple détaillé, ce problème souvent rencontré dans le calcul des vecteurs intervalles.

Exemple III.1 *L'exemple de R.E. Moore [MOO 66] illustre le wrapping effect à chaque pas d'intégration. Considérons l'équation différentielle suivante :*

$$\dot{u} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} u, u(0) \in \begin{bmatrix} [-0.1, 0.1] \\ [0.9, 1.1] \end{bmatrix} \quad (\text{III.27})$$

La solution de l'équation différentielle est un cercle de rayon unité dans le plan de phase $(u_{(1)}, u_{(2)})$ défini par

$$u(t) = \begin{bmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{bmatrix} u(0). \quad (\text{III.28})$$

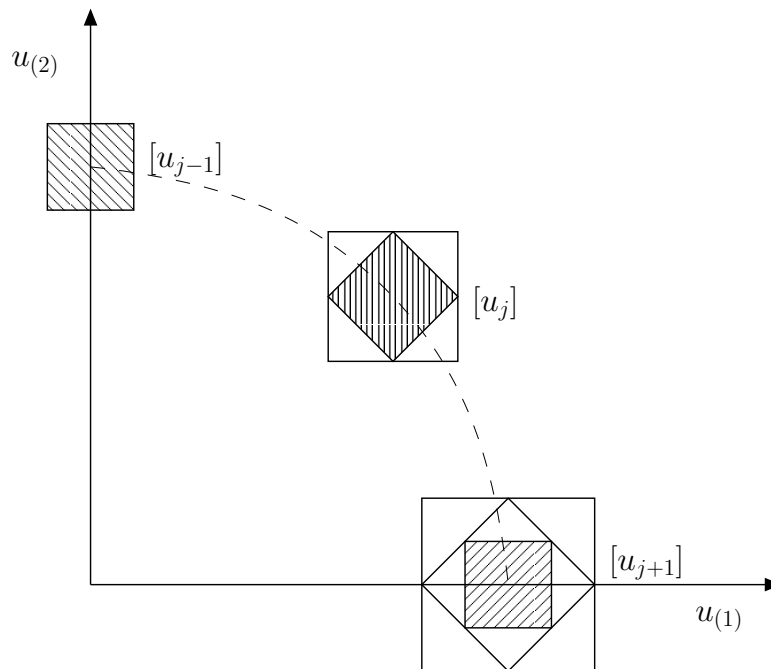


FIG. III.1 – L'effet d'enveloppement

A chaque itération, l'algorithme de la valeur moyenne (III.2) provoque une rotation du pavé solution qui est alors enveloppé dans un autre pavé dont les faces sont parallèles aux axes $u_{(1)}$ et $u_{(2)}$. La taille du pavé solution augmente à chaque instant d'intégration alors que la vraie solution ne change pas. La Figure III.1 illustre ce phénomène.

Dans la méthode de la valeur moyenne, l'effet d'enveloppement apparaît au niveau du produit de la matrice $[S_j]$ avec le vecteur intervalle $([x_j] - \hat{x}_j)$. Ceci est dû aux j produits de matrices intervalles avec des vecteurs intervalles. En effet, nous calculons

$$\begin{aligned}
 [x_1] &= [v_1] + [S_0]([x_0] - \hat{x}_0) \\
 [x_2] &= [v_2] + [S_1]([x_1] - \hat{x}_1) \\
 &= [v_2] + [S_1]([v_1] - \hat{x}_1) + [S_0]([x_0] - \hat{x}_0) \\
 &= [v_2] + [S_1]([v_1] - \hat{x}_1) + [S_1][S_0]([x_0] - \hat{x}_0) \\
 &\quad \vdots \\
 [x_{j+1}] &= [v_{j+1}] + [S_j]([x_j] - \hat{x}_j) \\
 &= [v_{j+1}] + [S_j]([S_{j-1}] \dots ([S_1]([S_0]([v_0] - \hat{x}_0)))) \dots
 \end{aligned} \tag{III.29}$$

où $[v_0] = [x_0]$. Le vecteur symétrique $([v_j] - \hat{x}_j)$ provoque donc bien un enveloppement du pavé solution à chaque pas d'intégration.

Un nouvel algorithme va être présenté de manière à réduire cet effet d'enveloppement en effectuant une transformation linéaire sur les matrices.

III.3.3 Méthode de la valeur moyenne étendue

Nous présentons la méthode de la valeur moyenne étendue développée par R. Rihm [RIH 94]. Cette méthode est basée sur un prétraitement des matrices à l'aide de transformations linéaires pour réduire le wrapping effect. L'algorithme utilisé est le suivant :

Algorithme III.3 (Extended Mean Value)

Entrées : $[x_j]$, \hat{x}_j , h , $[\tilde{x}_j]$, $[v_j]$, $[p_j]$, A_j ;
Sorties : $[x_{j+1}]$, \hat{x}_{j+1} , $[v_{j+1}]$, $[p_{j+1}]$, A_{j+1} ;

1. $[v_{j+1}] = \hat{x}_j + \sum_{i=1}^{k-1} h^i f^{[i]}(\hat{x}_j) + h^k f^{[k]}([\tilde{x}_j]);$
2. $[S_j] = \{I + \sum_{i=1}^{k-1} h^i J(f^{[i]}; [x_j])\};$
3. $[q_{j+1}] = ([S_j]A_j)[p_j] + [S_j]([v_j] - \hat{x}_j);$
4. $[x_{j+1}] = [v_{j+1}] + [q_{j+1}];$
5. $A_{j+1} = m([S_j]A_j);$
6. $[p_{j+1}] = A_{j+1}^{-1}([S_j]A_j)[p_j] + (A_{j+1}^{-1}[S_j])([v_j] - \hat{x}_j);$
7. $\hat{x}_{j+1} = m([x_{j+1}]).$

Valeurs des différentes variables de l'algorithme à l'instant $j = 0$:

$$- \hat{x}_0 = m([x_0]);$$

- $[p_0] = 0$;
- $A_0 = I$ où I est la matrice identité;
- $[v_0] = [x_0]$;

Il existe deux méthodes pour choisir la matrice A_{j+1} : la factorisation QR et la méthode parallélépipédique. Ces méthodes ont été développées par R. Löhner [LOH 87]. Nous utiliserons la méthode parallélépipédique pour sa simplicité. Dans ce cas, la matrice A_{j+1} peut être calculée par la formule suivante :

$$A_{j+1} = m([S_j]A_j). \quad (\text{III.30})$$

Pour contrôler l'effet d'enveloppement, la matrice A doit satisfaire la condition suivante :

$$A_{j+1}^{-1}([S_j]A_j) \approx I \quad (\text{III.31})$$

où I est la matrice identité.

Démonstration Soit $\hat{S}_j = m([S_j])$. Nous introduisons la matrice intervalle $[E_j]$ telle que $[E_j] = [S_j] - \hat{S}_j$, alors :

$$A_{j+1} = \hat{S}_j A_j,$$

A_j est une matrice non singulière dont les composantes sont des valeurs ponctuelles et non des intervalles donc $m(A_j) = A_j$.

$$\begin{aligned} A_{j+1}^{-1} \hat{S}_j A_j &= A_{j+1}^{-1} (\hat{S}_j A_j), \\ A_{j+1}^{-1} ([S_j] - [E_j]) A_j &= A_j^{-1} \hat{S}_j^{-1} (\hat{S}_j A_j), \\ A_{j+1}^{-1} ([S_j] A_j) &= A_j^{-1} \hat{S}_j^{-1} (\hat{S}_j A_j + [E_j] A_j), \\ &= I + A_j^{-1} \hat{S}_j^{-1} [E_j] A_j. \end{aligned}$$

Comme

$$\|A_j^{-1} \hat{S}_j^{-1} [E_j] A_j\| \leq \text{cond}(A_j) \|\hat{S}_j^{-1} [E_j]\|,$$

si $\|\hat{S}_j^{-1} [E_j]\|$ est assez petit et que $\text{cond}(A_j)$ n'est pas trop grand, alors $A_{j+1}^{-1}([S_j]A_j) \approx I$.

Les algorithmes d'estimation d'état et de paramètres que nous présentons dans le chapitre suivant combinent l'algorithme de la valeur moyenne étendue et l'inversion ensembliste.

Chapitre IV

Estimation d'état et de paramètres

Considérons le système décrit par les équations d'état suivantes :

$$\begin{cases} \dot{x}(t) = f(x(t), p) \\ y(t) = g(x(t), p) \\ x(t_0) \in \mathbb{X}_0 \\ p \in \mathbb{P}_0 \end{cases} \quad (\text{IV.1})$$

où les fonctions f et g sont non linéaires, \mathbb{X}_0 est le domaine a priori connu pour la condition initiale $x(t_0)$ et \mathbb{P}_0 est le domaine initial pour le vecteur de paramètres p .

Les sorties mesurées $y_{exp}(t_j)$ sont supposées disponibles à chaque instant $t_j \in \{t_1, \dots, t_N\}$. Dans un contexte à erreurs bornées, nous avons vu dans le Chapitre I qu'il existe un domaine admissible \mathbb{Y}_j , à chaque instant de mesure t_j , pour le vecteur des sorties y_j :

$$\forall t_j \in \{t_1, \dots, t_N\}, y_j = y(t_j) = g(x(t_j), p) \in \mathbb{Y}_j. \quad (\text{IV.2})$$

Le vecteur des sorties doit en fait être noté $[y_j]$ car il s'agit d'un vecteur intervalle. De même, il existe un domaine admissible \mathbb{E} défini dans (I.2) pour l'erreur entre les sorties mesurées $y_{exp}(t_j)$ et les sorties du modèle $[y_j]$.

IV.1 Estimation d'état

Nous utilisons maintenant la méthode d'intégration numérique garantie, présentée au chapitre précédent, pour évaluer la sortie du modèle et le vecteur d'état aux instants de mesure $t_j \in \{t_1, \dots, t_N\}$ de manière à ce que la relation (IV.2) soit vérifiée. Pour résoudre le problème d'estimation d'état [KIE 02][RAM 05], le vecteur de paramètres p est considéré connu.

IV.1.1 Estimateur causal

Lorsque l'état initial $[x_0]$ est relativement bien connu, c'est à dire que sa longueur est faible (jusqu'à 5% de la valeur exacte), un estimateur causal peut être mis en oeuvre. Pour réaliser une estimation en ligne de l'état, nous utilisons une méthode qui se compose de deux phases :

- une phase de prédiction qui calcule l'ensemble atteignable pour l'état $[x_{j+1}]$,
- une phase de correction qui élimine toutes les parties incohérentes de l'état prédit $[x_{j+1}]$, avec le domaine admissible \mathbb{Y}_i pour la sortie du modèle $[y_{j+1}]$.

Les phases de prédiction et de correction sont illustrées sur la Figure IV.1.

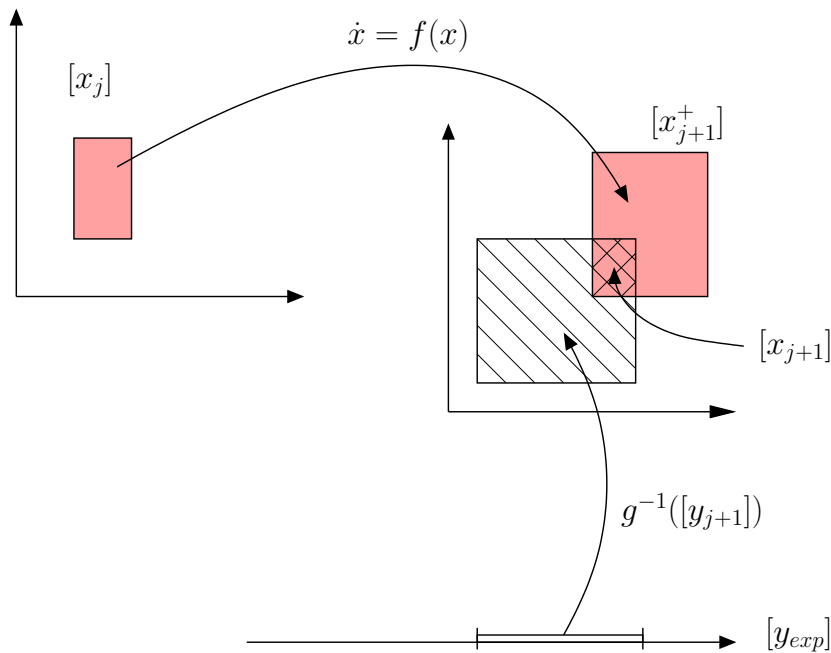


FIG. IV.1 – L'estimation d'état : la phase de prédiction et la phase de correction

Pour calculer l'ensemble atteignable pour l'état dans l'étape de prédiction, nous utilisons les résultats obtenus à l'aide de l'algorithme de la valeur moyenne étendue (E.M.V.)(III.3). L'ensemble admissible \mathbb{Y}_{j+1} pour les sorties du modèle $[y_{j+1}]$ est noté $[y_{exp,j+1}]$.

L'algorithme pour une estimation en ligne de l'état est le suivant :

Algorithme IV.1 (Estimateur Causal)

Entrées : $[x_0]$, $[y_{exp_j}]$, h , α ;

Sorties : $[x_1]$, \dots , $[x_N]$;

1. Pour j allant de 0 à $N - 1$,
2. $[\tilde{x}_j] := \text{enclosure}([x_j], h, \alpha)$,

3. *Prédiction : E.M.V.*

Entrées : $[x_j], \hat{x}_j, h, [\tilde{x}_j], [v_j], [p_j], A_j;$

Sorties : $[x_{j+1}]^+, \hat{x}_{j+1}, [v_{j+1}], [p_{j+1}], A_{j+1};$

4. *Correction :* $[x_{j+1}] = [x_{j+1}]^+ \cap g^{-1}([y_{j+1}]);$

5. *Fin.*

IV.1.2 Estimateur non causal

Si l'état initial $[x_0]$ est fortement incertain (par exemple, si sa longueur atteint 20% de sa valeur exacte), il est préférable de considérer une estimation hors-ligne, donc un estimateur non causal. Car l'algorithme mis en oeuvre pour ce type d'estimation permet de réduire l'intervalle correspondant à l'état initial $[x_0]$ incertain.

Cette estimation est basée sur une propagation de contraintes dans le sens direct et dans le sens rétrograde. Elle comporte donc deux phases de prédiction (effectuées à l'aide l'algorithme E.M.V.) et deux phases de correction (ne conservant que les parties cohérentes de l'état prédit avec le domaine d'admissibilité \mathbb{Y}_i pour les sorties du modèles). Nous obtenons alors l'algorithme suivant :

Algorithme IV.2 (Estimateur Non Causal)

Entrées : $[x_0], [y_{exp_j}], h, \alpha;$

Sorties : $[x_0], \dots, [x_N];$

1. **Propagation dans le sens direct :**

2. *Pour* j allant de 0 à $N - 1$,

3. $[\tilde{x}_j] := \text{enclosure}([x_j], h, \alpha),$

4. *Prédiction : E.M.V.*

Entrées : $[x_j], \hat{x}_j, h, [\tilde{x}_j], [v_j], [p_j], A_j;$

Sorties : $[x_{j+1}]^+, \hat{x}_{j+1}, [v_{j+1}], [p_{j+1}], A_{j+1};$

5. *Correction :* $[x_{j+1}] = [x_{j+1}]^+ \cap g^{-1}([y_{j+1}]);$

6. **Propagation dans le sens rétrograde :**

7. *Pour* j allant de $N - 1$ à 0,

8. $[\tilde{x}_{j+1}] := \text{enclosure}([x_{j+1}], h, \alpha),$

9. *Prédiction : E.M.V.*

Entrées : $[x_{j+1}], \hat{x}_{j+1}, -h, [\tilde{x}_{j+1}], [v_{j+1}], [p_{j+1}], A_{j+1};$

Sorties : $[x_j]^+, \hat{x}_j, [v_j], [p_j], A_j;$

10. *Correction :* $[x_j] = [x_j]^+ \cap g^{-1}([y_j]);$

11. *Fin.*

Cet algorithme peut être répété plusieurs fois jusqu'à ce qu'il n'y ait plus d'amélioration. Nous nous contenterons d'effectuer une seule propagation dans les deux sens.

IV.2 Estimation de paramètres

L'estimation de paramètres dans un contexte à erreurs bornées [RAï 03] [KIE 05] se pose comme un problème d'inversion ensembliste :

$$\begin{aligned}\mathbb{P} &= \{p \in \mathbb{P}_0 | [e(p)] \in \mathbb{E}\} \\ \mathbb{P} &= e^{-1}(\mathbb{E})\end{aligned}\tag{IV.3}$$

Nous cherchons à estimer le vecteur de paramètres p , initialement compris dans un domaine de recherche a priori \mathbb{P}_0 , tel que les sorties prédites du modèle $[y_j]$ soient cohérentes avec les mesures expérimentales y_{exp_j} et les bornes d'erreur acceptables.

Pour résoudre ce problème d'inversion ensembliste, il faut caractériser l'ensemble des paramètres \mathbb{P} , c'est à dire trouver l'ensemble image réciproque de l'ensemble d'erreur admissible \mathbb{E} par la fonction d'inclusion connue $[e]$. Ce problème d'inversion ensembliste peut être résolu de manière approximative par l'algorithme SIVIA présenté au Chapitre II [JAU 93b].

Cet algorithme nécessite trois étapes :

- le choix du pavé initial \mathbb{P}_0 qui est censé contenir toutes les valeurs du vecteur de paramètres recherché,
- les tests d'inclusion qui vérifient que l'erreur entre les sorties mesurées y_{exp_j} et les sorties du modèle $[y_j]$ calculées par l'algorithme de la valeur moyenne étendue (III.3) est un vecteur intervalle appartenant au domaine d'erreur admissible \mathbb{E} dont les bornes e_{min} et e_{max} sont connues,
- la bisection du vecteur de paramètres p lorsque celui-ci est dit incertain, c'est à dire lorsque l'intersection du vecteur intervalle d'erreur $[e]$ et du domaine admissible d'erreur \mathbb{E} n'est pas vide.

Le vecteur de paramètres sera obtenu sous forme d'intervalles dont la longueur $w([p])$ sera inférieure ou égale à la précision requise ε fixée par l'utilisateur. Cette précision ε stoppe les bisections du vecteur de paramètres.

Ces étapes sont détaillées dans l'algorithme suivant :

Algorithme IV.3 (Estimation Paramètres)

Entrées : $\mathbb{P}_0, \varepsilon, [y_{exp_j}], \mathbb{E}, [x_0], h, \alpha;$

Sorties : $[p], [x_1], \dots, [x_N];$

1. *Initialisation* : $\mathcal{P}_{ad} := \emptyset, \mathcal{P}_{int} := \text{DéposerSommet}([p_0]),$
2. *Pour* j allant de 0 à $N - 1,$

3. si $\mathcal{P}_{int} \neq \emptyset$ alors $[p] := \text{ExtraireSommet}(\mathcal{P}_{int})$;
4. sinon $[p] := \text{ExtraireSommet}(\mathcal{P}_{ad})$;
5. Construire un état étendu : $[x_{j+1}^e]$;
6. $[\tilde{x}_j^e] := \text{Enclosure}([x_j^e], h, \alpha)$;
7. $[x_{j+1}^e] := \text{E.M.V.}([x_j^e], h, \alpha)$;
8. Calcul des sorties du modèle : $[y_{j+1}] = g([x_{j+1}])$;
9. Calcul des erreurs : $[e_{j+1}] := y_{exp_{j+1}} - [y_{j+1}]$;
10. si $[e_{j+1}] \subset \mathbb{E}$ alors $\mathcal{P}_{ad} := \text{DéposerSommet}([p])$ puis aller au pas 2;
11. si $[e_{j+1}] \cap \mathbb{E} = \emptyset$ alors aller au pas 2;
12. si $[e_{j+1}] \cap \mathbb{E} \neq \emptyset$ et $w([p]) < \varepsilon$ alors $\mathcal{P}_{ad} := \text{DéposerSommet}([p])$ puis aller au pas 2.
13. sinon $([p](1), [p](2)) := \text{Bissecter}([p])$ et $\mathcal{P}_{int} := \text{DéposerSommet}([p](1), [p](2))$ puis aller au pas 2.

La fonction *ExtraireSommet* récupère l'intervalle placé au sommet de la pile considérée. La valeur de l'intervalle récupéré est alors donnée à l'intervalle de paramètres $[p]$ qui doit être testé.

La fonction *DéposerSommet* nous permet de rajouter des intervalles au sommet d'une pile.

Les algorithmes *Enclosure* et *E.M.V.* permettant d'intégrer l'équation d'état du problème (IV.1) ont été détaillés dans le chapitre III. Certains arguments de l'algorithme E.M.V ont été omis afin d'alléger les notations.

Les intervalles à tester sont stockés dans la pile \mathcal{P}_{int} . A chaque bisection, la pile \mathcal{P}_{int} est augmentée de deux nouveaux intervalles. Ainsi, au pas 13 de l'algorithme, les deux intervalles $[p](1)$ et $[p](2)$ générés par la bisection de $[p]$, dont la longueur est supérieure à la précision ε , sont placés au sommet de la pile \mathcal{P}_{int} .

La pile \mathcal{P}_{ad} contient les intervalles de paramètres qui ont été dits admissibles, c'est à dire qu'elle contient les intervalles solutions et les intervalles trop petits pour être bissectés.

La solution finale du problème d'estimation de paramètres est l'union des pavés disjoints contenus dans cette pile \mathcal{P}_{ad} .

Nous évaluons les algorithmes d'estimation d'état et de paramètres présentés dans cette partie sur un modèle de comportement en vol longitudinal d'un avion dans le chapitre suivant.

Chapitre V

Application

Nous cherchons à estimer les états d'un modèle de comportement en vol longitudinal d'un avion à partir de données simulées. Le cas de données aberrantes sera donc écarté. Notre simulation est effectuée sur $1s$ et les sorties sont mesurées toutes les $0.02s$. Nous prendrons donc le pas d'intégration $h = 0.02s$ pour pouvoir évaluer les sorties de notre modèle aux instants de mesure choisis.

V.1 Modèle de comportement en vol d'un avion

Le modèle étudié est celui du Dornier 28 établi par des essais en soufflerie [JAU 02a]. Il s'agit d'un modèle de comportement en vol longitudinal d'un avion dans une atmosphère en repos. Un choix possible pour le vecteur d'état est : $x = [V, \alpha, q, \theta]^T$, où

- V est la vitesse en $m.s^{-1}$,
- α est l'incidence en rad qui correspond à la direction du nez de l'avion par rapport au vecteur vitesse dans le plan vertical,
- q est la vitesse de tangage en $rad.s^{-1}$ qui est la vitesse de changement de direction de l'axe avion par rapport au repère terrestre dans le plan de symétrie de l'avion,
- θ est l'assiette longitudinale en rad qui correspond à la direction du nez de l'avion par rapport au repère terrestre dans le plan vertical.

La plupart des variables décrites précédemment peuvent être repérées sur le schéma d'avion de la Figure (V.1).

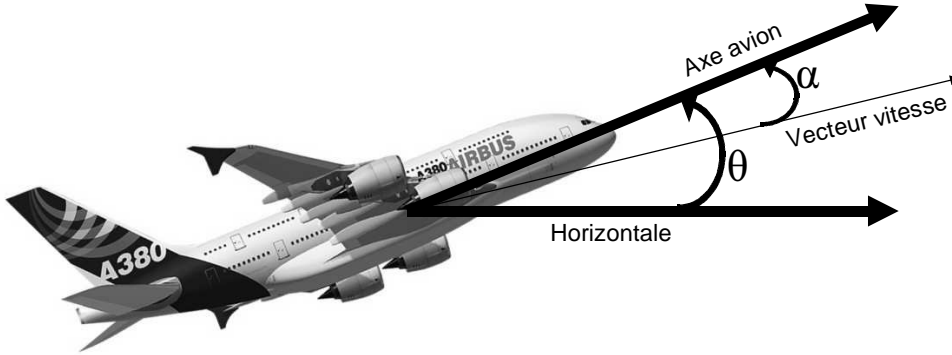


FIG. V.1 – Incidence et Assiette longitudinale d'un avion

Le modèle dynamique de comportement en vol du Dornier est non-linéaire contrôlé. Il peut être décrit par les équations suivantes :

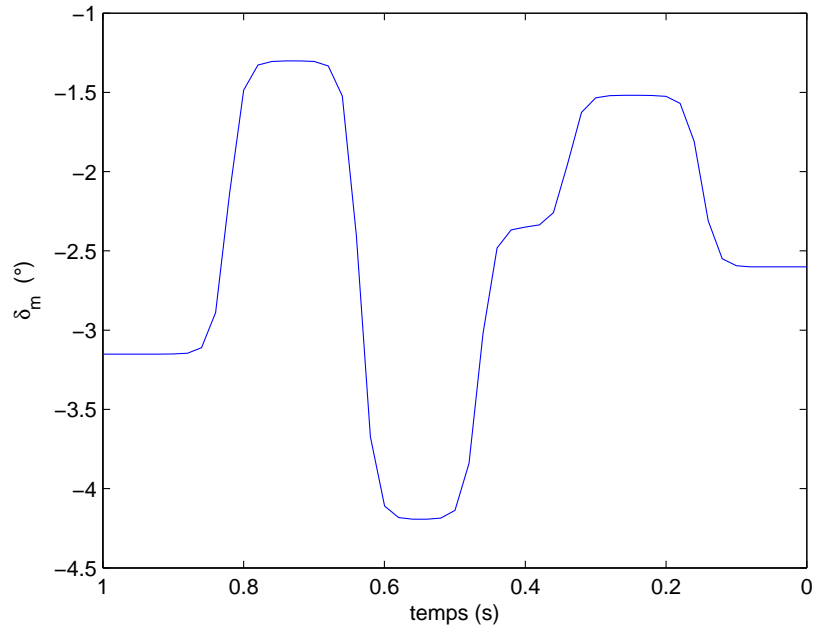
$$\left\{ \begin{array}{l} m\dot{V} = -mg \sin(\theta - \alpha) - \frac{\rho S V^2}{2} [C_x^0 + C_{x\alpha}(\alpha - \alpha_0) + C_{x\delta_m}(\delta_m - \delta_{m_0})], \\ mV(\dot{\alpha} - \dot{\theta}) = mg \cos(\theta - \alpha) - \frac{\rho S V^2}{2} \left[C_z^0 + C_{z\alpha}(\alpha - \alpha_0) + \frac{ql}{V} C_{zq} \right. \\ \quad \left. + C_{z\delta_m}(\delta_m - \delta_{m_0}) + C_{z\dot{\alpha}} \frac{\dot{\alpha} l}{V} \right], \\ I_{yy} \dot{q} = \frac{\rho S l V^2}{2} \left[C_{m\alpha}(\alpha - \alpha_0) + \frac{ql}{V} C_{mq} + C_{m\dot{\alpha}} \frac{\dot{\alpha} l}{V} + C_{m\delta_m}(\delta_m - \delta_{m_0}) \right], \\ \dot{\theta} = q, \end{array} \right. \quad (V.1)$$

où α_0 est la valeur d'équilibre de l'incidence α . Il ne s'agit pas d'une représentation d'état mais il est possible de s'y ramener facilement. L'entrée δ_m représente le braquage de gouverne de profondeur qui déplace un volet à l'arrière de l'avion créant ainsi un couple de tangage tendant à faire cabrer ou piquer l'avion. La valeur d'équilibre de l'entrée δ_m est notée δ_{m_0} .

Les paramètres que nous estimerons dans une future étude sont $C_{z\dot{\alpha}}$, C_{zq} , $C_{m\dot{\alpha}}$ et C_{mq} . Ce sont les dérivées de stabilité aérodynamique.

L'entrée utilisée pour notre modèle est une entrée optimale [JAU 02a]. Son comportement peut être visualisée sur la Figure V.2.

Nous ne considérerons que le problème d'estimation d'état pour cette application. Cependant les algorithmes présentés dans la partie théorique pour le problème d'estimation de paramètres ont été programmés. Ces résultats nécessitant d'être approfondis, ils ne

FIG. V.2 – *Entrée optimale*

seront pas présentés dans ce rapport.

V.2 Estimation d'état du modèle

Dans notre cas, le vecteur d'état est observé en totalité mais nous supposons que ces observations sont perturbées par des bruits de mesure inconnus $e(t_j, p)$ vérifiant l'équation suivante :

$$e(t_j, p) = y_{exp}(t_j) - y(t_j, p), \quad \text{pour } j = 1, \dots, N, \quad (\text{V.2})$$

où y_{exp} représente le vecteur des mesures et y celui des observations et où $e(t_j, p)$ correspond aux bruits de mesure inconnus mais dont les bornes inférieure et supérieure sont connues ou acceptables. Ces bornes sont déterminées à partir d'informations relevées sur les instruments de mesure [COT 01]. Dans notre cas, elles sont supposées constantes.

L'erreur sur la variable d'état V doit être comprise dans l'intervalle $[-0.15, 0.15]$ et l'erreur sur les variables d'état α , q , θ doit appartenir à l'intervalle $[-0.0105, 0.0105]$. Le domaine admissible \mathbb{E} est donc défini par le vecteur intervalle (ou pavé) suivant :

$$\mathbb{E} = \begin{bmatrix} -0.15 & 0.15 \\ -0.0105 & 0.0105 \\ -0.0105 & 0.0105 \\ -0.0105 & 0.0105 \end{bmatrix} \quad (\text{V.3})$$

Le domaine admissible pour les sorties du modèle peut donc s'écrire :

$$\mathbb{Y}_j = y_{exp_j} + \begin{bmatrix} -0.15 & 0.15 \\ -0.0105 & 0.0105 \\ -0.0105 & 0.0105 \\ -0.0105 & 0.0105 \end{bmatrix} \quad (\text{V.4})$$

où y_{exp_j} est la mesure bruitée obtenue par simulation.

Nous cherchons à estimer le vecteur des variables d'état du modèle de comportement en vol longitudinal de l'avion. Pour cela la plupart des paramètres du modèle sont considérés comme parfaitement connus. Le paramètre $C_{z\dot{\alpha}}$ est entaché d'une incertitude de plus ou moins 5%. Il appartient à l'intervalle $[1.7, 1.9]$.

La condition initiale pour l'état appartient à l'ensemble suivant :

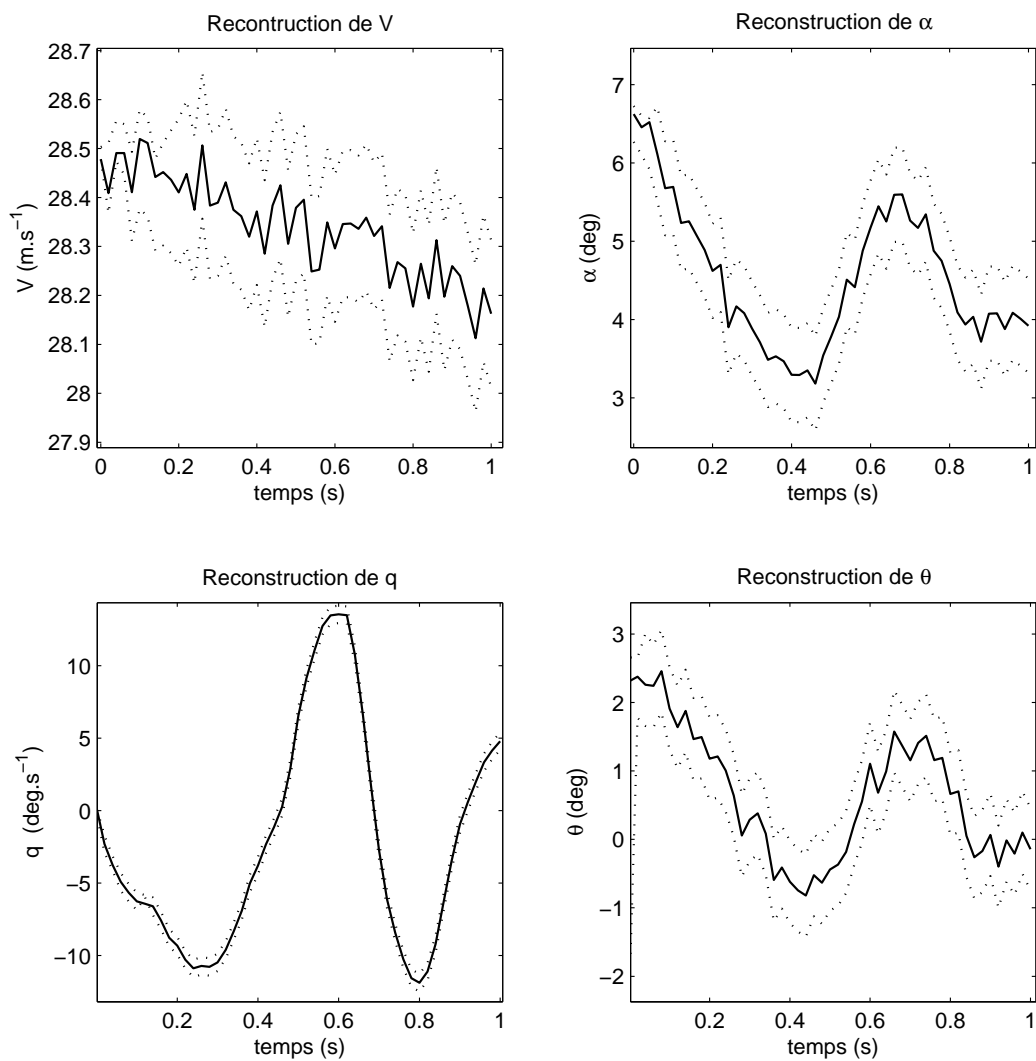
$$\mathbb{X}_0 = \begin{bmatrix} 28.4771 & 28.5029 \\ 0.1094 & 0.1174 \\ -0.0040 & 0.0040 \\ -0.0384 & 0.0464 \end{bmatrix} \quad (\text{V.5})$$

Les valeurs numériques sont ici exprimées en radians. La longueur des intervalles étant suffisamment petite, une estimation de type causale peut être mise en oeuvre.

En utilisant l'algorithme IV.1, nous obtenons les résultats au problème d'estimation sous la forme d'intervalles.

La Figure V.3 présente les trajectoires d'état reconstruites à l'aide de nos algorithmes.

La courbe correspondant au trait plein représente les données bruitées. Celles en pointillées sont les enveloppes calculées pour chaque variable d'état. Elles représentent donc la borne inférieure et la borne supérieure de l'intervalle solution du problème d'estimation ensembliste. Nous constatons que la valeur réelle appartient bien à chaque instant aux intervalles solutions.

FIG. V.3 – *Reconstruction des trajectoires*

Conclusion

Une nouvelle approche pour résoudre les problèmes d'estimation est étudiée dans ce rapport. Il s'agit de l'approche ensembliste permettant d'estimer la valeur de l'état et des paramètres sous la forme d'intervalles. Différents algorithmes ont été présentés. Leur efficacité a été illustrée sur le modèle de comportement en vol longitudinal d'un avion. Les premiers résultats obtenus dans le cas de l'estimation d'état sont encourageants. Concernant le problème d'estimation de paramètres, des améliorations sont en cours de développement.

Ce travail de Master Recherche s'ouvre sur un sujet de thèse où ces méthodes ensemblistes seront étendues à la détection de fautes et au diagnostic. En parallèle à ce travail théorique, l'accent sera également porté sur l'applicabilité des méthodes. Ceci se fera par de nouveaux algorithmes améliorant les résultats actuels, et par l'application des outils à des procédés industriels.

Références bibliographiques

- [ALA 05] T. ALAMO, J. BRAVO and E. CAMACHO, “Guaranteed state estimation by zonotopes”, *Automatica*, vol. 41, 2005, pages 1035-1043.
- [COR 94] G. CORLISS, “Guaranteed error bounds for ordinary differential equations”, Lectures notes presented at the VI-th SERC Numerical Analysis Summer School, 1994.
- [COT 01] P. COTON, C. BUCHARLES, C. JAUBERTHIE, T. LEMOING and L. PLANCKAERT, “Identification des dérivées de stabilités dynamiques”, Tech. report no. 1/05650, 2001, ONERA, Lille, France.
- [DUR 01] C. DURIEU and E. WALTER, *Identification des systèmes*, Chapitre Estimation ellipsoïdale à erreur bornée, Hermès, Paris, 2001.
- [JAU 93a] L. JAULIN and E. WALTER, “Guaranteed nonlinear parameter estimation from bounded-error data via interval analysis”, *Mathematics and Computers in Simulation*, vol. 35, 1993, pages 123-137.
- [JAU 93b] L. JAULIN and E. WALTER, “Set inversion via interval analysis for non-linear bounded-error estimation”, *Automatica*, vol. 29, 1993, pages 1053-1064.
- [JAU 99] L. JAULIN, “Le calcul ensembliste par analyse par intervalles et ses applications”, Habilitation à diriger des recherches, february 1999, LSS, Université Paris-Sud.
- [JAU 01] L. JAULIN, M. KIEFFER, O. DIDRIT and E. WALTER, *Applied Interval Analysis*, Springer-Verlag, London, 2001.
- [JAU 02a] C. JAUBERTHIE, “Méthodologies de planification d’expériences pour systèmes dynamiques”, PhD thesis, Université de Technologie de Compiègne, december 2002.
- [JAU 02b] L. JAULIN, B. I. and W. E., “Interval methods for nonlinear identification and robust control”, proceedings of *Conference on Decision and Control*, Las Vegas, december 2002.
- [JAU 03] L. JAULIN and P. ROUSTANT, “Calcul par intervalles et propagation des contraintes pour l’optimisation globale”, Cours, march 2003, Ecoles des mines, Saint Etienne.
- [JAU 04] L. JAULIN, “Interval contractors and their applications”, Cours, february 2004, Université Polytechnique de Catalogne.

- [KIE 99] M. KIEFFER, “Estimation ensembliste par analyse par intervalles, application à la localisation d’un véhicule”, PhD thesis, Université Paris-Sud, 1999.
- [KIE 00] M. KIEFFER, L. JAULIN, E. WALTER and D. MEIZEL, “Robust robot localization and tracking using interval analysis”, *Reliable Computing*, vol. 6, 2000, pages 337-362.
- [KIE 02] M. KIEFFER, , L. JAULIN and E. WALTER, “Guaranteed recursive nonlinear state bounding using interval analysis”, *International Journal of Adaptive Control and Signal Processing*, vol. 6, 2002, pages 191-218.
- [KIE 05] M. KIEFFER and E. WALTER, “Interval analysis for guaranteed nonlinear parameter estimation”, *Mathematical and Computer Modelling of Dynamic Systems*, vol. 11, no. 2, 2005, pages 171-181.
- [LES 03] S. LESECQ, A. BARRAUD and K. TRAN DINH, “Numerical accurate computations for ellipsoidal state bounding”, proceedings of *MED’03*, Rhodos, Grèce, june 2003.
- [LOH 87] R. LOHNER, *Enclosing the solutions of ordinary initial and boundary value problems*, In E. Kaucher, U. Kulisch, and C. Ullrich, editors, Stuttgart, Germany, 1987.
- [MIL 91] M. MILANESE and A. VICINO, “Estimation theory for nonlinear models and set membership uncertainty”, *Automatica*, vol. 27, no. 2, 1991, pages 403-408.
- [MOO 66] R. MOORE, *Interval Analysis*, Prentice-Hall, Englewood Cliffs, N.J., 1966.
- [NED 99] N. NEDIALKOV, “Computing rigorous bounds on the solution of an initial value problem for an ordinary differential equation”, PhD thesis, Université de Toronto, Canada, 1999.
- [RAï 03] T. RAÏSSI, N. RAMDANI and Y. CANDAU, “Parameter estimation for nonlinear continuous-time systems in a bounded error context”, proceedings of *Conference on Decision and Control*, Maui, Hawaii USA, december 2003.
- [RAM 05] N. RAMDANI, “Méthodes ensemblistes pour l’estimation”, Habilitation à diriger des recherches, october 2005, Université Paris XII.
- [RIH 94] R. RIHM, “Interval methods for IVP in ODEs, in topics in validated computations”, proceedings of *Topics in validated computations*, North-Holland, 1994, pages 173-208.
- [WAL 94] E. WALTER and L. PRONZATO, *Identification de modèles paramétriques à partir de données expérimentales*, Masson, 1994.

Index

Algorithme

- Enclosure, 27
- Estimateur Causal, 34
- Estimateur Non Causal, 35
- Extended Mean Value, 31
- ImageSP, 22
- Mean Value, 29
- SIVIA, 20

Bisection, 14

Domaine admissible

- pour l'erreur, 8
- pour les sorties, 8

Effet d'enveloppement, 30

Estimation d'état

- en ligne, 34
- hors ligne, 35

Fonction d'inclusion, 16

- centrée, 19
- naturelle, 17

Image ensembliste

- directe, 21
- réciproque, 19

Intervalle, 13

Opérateur de Lindelöf, 26

Pavés, 14

- pavage, 14
- sous-pavage, 14

Test d'inclusion, 20

Théorème du point fixe, 25