

الجمهورية الجزائرية الديمقراطية

الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد- تلمسان

Université Aboubakr Belkaïd- Tlemcen –

Faculté de TECHNOLOGIE



THESE

Présentée pour l'obtention du **grade de DOCTEUR EN SCIENCES**

En : Génie Electrique

Spécialité : Automatique

Par : BORSALI née KAHOUADJI Salima

**METHODES ENSEMBLISTES POUR L'IDENTIFICATION
DE PARAMETRES : APPLICATION AU BIOREACTEUR**

Soutenue publiquement, le 29 / 04 / 2017 , devant le jury composé de :

M^{me}Amel CHOUKCHOU BRAHAM

Maitre de conférence à Université de Tlemcen Président

M.Boumediène BENYAHIA

Maitre de conférence à université de Tlemcen Examineur

M.Hafid HAFFAF

Professeur à université Oran1 Examineur

M.Fayçal KHELFI

Professeur à université Oran1 Examineur

M.Brahim CHERKI

Professeur à université de TLEMCEEN Directeur de thèse

Laboratoire d'Automatique

UNIVERSITE de TLEMCEN
Faculté de Technologie
Département de Génie Electrique

A la mémoire de ma mère et de mon père

A la mémoire de mon époux

A ce que j'ai de plus cher au monde, mes enfants

Fatima Zahra,

Awicha Amina,

et Hamza

Mes petits enfants,

Mohammed Rida et shérazade

Remerciements

Je tiens à exprimer ma profonde gratitude à mon Directeur de thèse Pr. B.CHERKI pour son investissement durant toutes ces années, comme je le remercie aussi pour sa patience, sa disponibilité et ses précieux conseils. Cette thèse n'aurait jamais abouti sans son aide .

Je suis aussi profondément reconnaissante envers le Pr N.GHOUALI pour son aide et soutien.

Je tiens à remercier aussi tout le groupe spécialiste des méthodes ensemblistes en France, particulièrement Pr N.RAMDANI et Pr M.KIEFFER qui ont toujours accepté de m'accueillir, de m'aider et m'orienter dans mes recherches. Je suis très reconnaissante aussi envers le Pr E.WALTER et Pr Luc JAULIN pour l'organisation de l'école d'été de Grenoble en automatique qui a permis d'orienter mes recherches vers les méthodes ensemblistes et m'ouvrir ainsi une voie vers un domaine de recherche très promoteur.

Je tiens ensuite à remercier, Mme A.CHOUKCHOU BRAHAM Maître de conférences à l'Université de Tlemcen, pour avoir accepté de présider mon jury de thèse.

Mes vives remerciements vont aussi à M.KHELFI Fayçal professeur à université Oran1, M. HAFFAF Hafid professeur à université Oran1, et M.BENYAHIA Boumédiène maitre de conférence à université de Tlemcen, pour avoir accepté de constituer le Jury de cette thèse afin d'évaluer le travail, qu'ils soient tous profondément remerciés pour le temps et l'intérêt qu'ils ont porté à ce travail en acceptant de l'examiner.

Je tiens vivement à remercier tous les membres de mon laboratoire d'appartenance, le Laboratoire d'Automatique de Tlemcen pour leur aide, les conseils et encouragements qu'il ont toujours fait montré en mon égard durant toutes ces années.

Enfin, je remercie sincèrement toute ma famille pour leur continuel soutien.

Résumé

Le but de ce travail est d'explorer d'autres voies d'identification de paramètres, appliquées dans le domaine de la biotechnologie. Ce choix est justifié pour plusieurs raisons, comme les différents Bugs informatiques causés par les erreurs d'arrondis. L'approche ensembliste basée sur l'arithmétique des intervalles est bien adaptée dans ces cas, c'est à dire quand l'intervalle d'erreur introduit pendant l'acquisition des données est trop large comme c'est le cas en biologie ou sciences médicales... Dans ce contexte une variable est représentée uniquement par un ensemble \mathbb{X} , appelé ensemble de vraisemblance ou domaine, supposé contenir la valeur réelle x . Nous testons deux approches, l'identification se fait via une intégration numérique garantie utilisant la théorie de comparaison des inégalités différentielles. Cette approche est développée pour des systèmes d'équations différentielles monotones dans le contexte fixé par une règle déduite du théorème de Müller. Le but de la méthode consiste à transformer le problème avec les conditions initiales sont incertaines, donc appartenant à un intervalle, en deux problèmes déterministes dont la résolution fournit les trajectoires inférieures et supérieures qui englobent de manière garantie toute les solutions d'état générées. Par la suite, nous testons la méthode LSCR (Leave-out Sign-dominant Correlation Regions) développé par Campi qu'on a associé à une méthode d'identification utilisant l'inversion ensembliste. Nous procédons aussi à une étude comparative avec la méthode des moindres carrés.

Mots-clefs

identification, bioréacteur, inversion ensemblistes, arithmétique des intervalles.

Abstract

The purpose of this work is to explore other ways of identification parameters, applied in the field of biotechnology. This choice is justified for several reasons, such as different computer Bugs caused by rounding errors. The set-approach based on interval arithmetic is well suited in these cases, when the range of error introduced during the acquisition of data is too large, as is the case in biology or medical sciences ... The ensemblist method based on interval arithmetic is well suited in these cases. In this context a variable is represented only by a set \mathbb{X} , called set likelihood or field supposed to contain the real value x . We test two approaches, the identification is done via a guaranteed numerical integration using the theory of comparing differential inequalities. This approach is developed for monotonous differential equations in the context set by a deduced rule from the Müller theorem. The aim of the method is to transform the problem with the initial conditions are uncertain, thus belonging to an interval, into two deterministic problems whose resolution provides the lower and upper trajectories that include all generated solutions so guarantees . Subsequently, we test the method LRCS (Leave-out Sign-dominant Correlation Regions) developed by Campi then we have associated with a method of identification using the set inversion. We also perform a comparative study with the least squares method

Keywords

identification, bioreactor, set-inversion, interval arithmetic.

الملخص

الهدف من هذا البحث هو التعرف على طرق أخرى وتقنيات تطبيقية حديثة في مجال البيوتكنولوجي . هذا الاختيار يبرر عدة تساؤلات، مثل مختلف أنواع الخلل التي تحدثها الأخطاء المتكررة أو الدورية في الإعلام الآلي.

الطرق المجموعات التي تعتمد على الحساب الفروق , تعتبر الطريقة الملائمة في هذه الحالة يعني عندما يكون الخطأ الفاصل داخل خلال عملية اكتساب المعطيات طويل, مثل ما هو الحال في البيولوجي و العلوم الطبية . في هذا السياق يكون تمثيل المتغير بمجموعة س تدعى مجموعة احتمالية أو مجال يحتمل انه يحتوي على قيمة س الحقيقية. نختبر طريقتين , الأولى التحديد من خلال دمج الرقمي و الذي يضمن لنا استعمال نظرية المقارنة لعدم مساواة الفارقة , هذه الطريقة تطورت لنظام المعادلات الفارقة الروتيني في السياق محدد من خلال نظرية Müller الهدف من هذه الأخيرة يكمن في تغيير الإشكال مع العوامل الأصلية الذي يبقى غير محدد أي تنتمي إلى نفس المجال في إشكاليتين محددتين التي تكون المعالجة ذات مسار السفلي وعلوي والذي يجمع بكل الضمانات كل الحلول على العموم .

بعد ذلك نقوم بتقييم طريقة LSCR المعمول بها من طرف عندما ندمجها بطريقة التعرف و استخدام الانقلاب المجموعات دراسة مقارنة مع طريقة المربعات الناقصة.

الكلمات المفتاحية:

التعريفات - المفاعل الحيوي - الانقلاب المجموعات - حساب المفروقات

Table des matières

Table des figures	12
Liste des tableaux	15
Introduction	16
Notations	22
1 Modèle du Bioréacteur pour la Nitrification	24
1.1 Introduction	24
1.2 Traitement des eaux par boues activées	25
1.3 Cycle de l'azote	26
1.4 Traitement biologique de l'azote	26
1.5 La nitrification	27
1.5.1 Les bactéries nitrifiantes	28
1.5.2 Facteurs influents sur la nitrification	28
1.5.3 La dénitrification	29
1.6 Les réacteurs biologiques	29
1.7 Mode de fonctionnement d'un bioréacteur	30
1.7.1 Le fonctionnement discontinu	30
1.7.2 Le fonctionnement semi-discontinu	30
1.7.3 Le fonctionnement en continu	30
1.8 Dynamique des bactéries en chémostat	31
1.8.1 Mise en équation	31
1.8.2 Modèle de Chémostat avec cinétique de Monod	33
1.9 Dispositif expérimental	35
1.9.1 Modélisation de plusieurs populations en chémostat	35

1.9.2	Modèle mathématique	36
1.10	Conclusion	38
2	Analyse ensembliste	39
2.1	Introduction	39
2.2	Opérations sur les ensembles	39
2.2.1	Opérations ensemblistes pures	39
2.2.2	Opérations ensemblistes étendues	41
2.2.3	Représentation d'ensembles	41
2.3	Arithmétique des intervalles	43
2.3.1	Calcul par Intervalles	43
2.3.2	Définition d'un pavé	44
2.4	Fonction d'inclusion	46
2.4.1	Fonction d'inclusion naturelle	47
2.4.2	Fonction élémentaire	48
2.4.3	Fonction d'inclusion centrée	49
2.4.4	Fonction d'inclusion de Taylor	49
2.4.5	Définition d'un sous-pavage	50
2.4.6	Réduction d'un pavé	51
2.5	Pessimisme	52
2.5.1	Phénomène d'enveloppement	52
2.5.2	Phénomène de dépendance	53
2.6	Image directe et inverse d'un ensemble	54
2.7	Contraction	55
2.7.1	Problème de satisfaction de contraintes	55
2.7.2	Notion de consistance	57
2.7.3	Projection de contraintes	58
2.7.4	Sous-résolveurs	58
2.8	Inversion ensembliste	60
2.8.1	L'algorithme	61
2.8.2	L'algorithme SiviaP	62
2.9	Conclusion	62

3	Identification dans le contexte à erreurs bornées	63
3.1	Introduction	63
3.2	Méthodes classiques d'identification	63
3.2.1	Statistiques en identification	67
3.2.2	Les moindres carrés	69
3.3	Identification dans le contexte à erreurs bornées	72
3.4	Application en robotique	73
3.5	Identification des paramètres	75
3.5.1	Méthode des moindres carrés	75
3.6	Résultats	76
3.7	Discussion	78
3.8	Conclusion	84
4	Applications au bioréacteur	85
I	Théorie de comparaison des inégalités différentielles	86
4.1	Introduction	87
4.2	Définitions	87
4.2.1	Systèmes monotones	87
4.2.2	Systèmes préservateurs d'ordre	88
4.2.3	Représentation graphique	88
4.2.4	Systèmes coopératifs	92
4.3	Intégration numérique garantie des ODE	94
4.3.1	Cas des systèmes non monotones	94
4.3.2	Théorème de Müller	98
4.3.3	Règle de construction des solutions minimales et maximales	100
4.4	Identification dans le contexte à erreurs bornées	102
4.5	La bibliothèque VNODELP	103
4.6	Application au bioréacteur	104
4.6.1	Étude de la monotonie	105
4.6.2	Préservation d'ordre	106
4.6.3	Équation des trajectoires minimales et maximales	107

4.7	Résultats	107
4.8	Conclusion	118
II Le méthode LSCR(Leave-out Sign-dominant Correlation Regions)		120
4.9	Introduction	121
4.10	Principe de la méthode	121
4.11	Exemple de Campi	124
4.12	Caractérisation garantie	128
4.13	Application au Bioréacteur	128
4.14	Conclusion	132
A	Annexes	137
	Bibliographie	142

Table des figures

1.1	L'eutrophisation des eaux	24
1.2	Station d'épuration des eaux usées	26
1.3	Les deux étapes de la nitrification	27
1.4	Le Bioréacteur	30
2.1	Opération ensemblistes pures	40
2.2	Pavé	45
2.3	Composantes de vecteurs intervalles	45
2.4	Différentes images d'un pavé $[x]$ par une fonction f , sa fonction d'inclusion $[f]$ et sa fonction d'inclusion minimale $[f]^*$	48
2.5	Bissection d'un pavé	52
2.6	Effet d'enveloppement conséquent à la rotation d'un pavé	53
2.7	Image directe d'un sous ensemble	54
2.8	Contraction d'un ensemble	57
2.9	Sivia	61
3.1	Étude de système dynamique	63
3.2	Procédure d'identification	65
3.3	Trois estimateurs pour un paramètre θ	67
3.4	Nuage de points et droite de régression	70
3.5	Modèle d'identification	72
3.6	Robot scara	74
3.7	Temps de simulation en fonction de la précision ϵ	78
3.8	Trajectoires excitantes	78
3.9	Couples appliqués aux axes du robot et au modèle	79
3.10	Sous Pavage obtenu par Sivia pour une précision $\epsilon = 0.01$	79

3.11	Sous Pavage obtenu par SiviaP pour une précision $\epsilon = 0.01$	80
3.12	(a) Couples, (b) Zoom de la Fig 3.9	81
3.13	Sous Pavage obtenu par SiviaP avec une précision $\epsilon = 0.05$	82
3.14	Sous Pavage obtenu par Proj2d avec une précision $\epsilon = 0.01$ pour identifier les 2 paramètres : ZZR_1 et Fs_1	83
3.15	Sous Pavage obtenu par Proj2d, avec une précision $\epsilon = 0.01$ projection de MX_2 et Fs_2	83
4.1	graphe de représentation d'ordre de l'équation 4.6	90
4.2	graphe de représentation d'ordre de l'équation 4.8	91
4.3	Trajectoires englobantes de la solution x1	98
4.4	Trajectoires englobantes de la solution x2	98
4.5	Solution d'un problème de valeur initiale (IVP) par VNODELP	104
4.6	Bornes a priori et bornes strictes	104
4.7	Graphe de préservation d'ordre	106
4.8	Simulations des mesures enregistrées (<i>bleu</i>) et du modèle de nitrifica- tion(+)	108
4.9	Simulation du modèle avec VNODELP (a,b,c les substrats S1,S2,S3 resp, d la biomasse $X = X1 + X2$)	109
4.10	Identification des paramètres par moindres carrés Mesures prélevées : courbe bleue étoilée Modèle simulé : courbe verte Modèle obtenu pour les valeurs identifiées par MC : courbe rouge	110
4.11	Trajectoires englobantes pour une période $2j$ à $22j$ En bleue : courbes max En vert : courbe min En rouge : les mesures réelles	111
4.12	Trajectoires englobantes pour une période $18j$ à $27j$ En bleue : courbes max En vert : courbe min En rouge : les mesures réelles	111
4.13	Trajectoires englobantes(bleue), trajectoires englobantes optimales(rouge), mesures(*)	112
4.14	Pavés solutions relatifs aux paramètres μ_{max1} et μ_{max2} obtenu par Sivia	113
4.15	Pavés solutions relatifs aux paramètres μ_{max1} et μ_{max2} obtenu par SiviaP	113
4.16	Pavés solutions relatifs aux paramètres μ_{max1} et μ_{max2} obtenu par SiviaP	114

4.17 Pavés solutions relatifs aux paramètres μ_{max1} et μ_{max2} obtenu par SiviaP	114
4.18 Pavés solutions relatifs aux paramètres k_1 et k_2 obtenu par Sivia	115
4.19 Pavés solutions relatifs aux paramètres k_1 et k_2 obtenu par SiviaP	116
4.20 Pavés solutions relatifs aux paramètres k_1 et k_2 obtenu par SiviaP	116
4.21 Pavés solutions relatifs aux paramètres k_{s1} et k_{s2} obtenu par Sivia et SiviaP	117
4.22 Pavés solutions relatifs aux quatres paramètres μ_{max1} , μ_{max2} , k_1 et k_2 obtenu par SiviaP	117
4.23 Modèle linéaire	123
4.24 Courbe des mesures	125
4.25 Tableau des sous ensembles I_n	126
4.26 Fonctions de corrélations $g_i(\theta)$	126
4.27 (a) Maillage de l'espace de recherche obtenu par LSCR, (b) Pavage de domaine avec Sivia, et leur zoom autour du paramètre apriori (*)	129
4.28 (a) Maillage de l'espace de recherche obtenu par LSCR, (b) Pavage de domaine avec Sivia, et leur zoom autour du paramètre apriori (*)	130
4.29 (a) Maillage de l'espace de recherche obtenu par LSCR, (b) Pavage de domaine avec Sivia, et leur zoom autour du paramètre apriori (*)	131

Liste des tableaux

3.1	Paramètres articulaires	74
3.2	Paramètres estimés par moindres carrés	77
3.3	Identification des paramètres ZZR_1 et $Fs1$	82
3.4	Résultats comparatifs	83
4.1	Comparaison des résultats obtenus par trois méthodes	118
4.2	Résultats	132

Introduction

« Tout le monde croit que les erreurs sont normales, les mathématiciens parce qu'ils pensent que c'est un fait expérimentalement établi et les expérimentateurs parce qu'ils croient que c'est un théorème »

Gabriel Lippmann (prix Nobel de physique)

Comme l'a si bien dit E.Walter [95], « L'identification est plus un art qu'une science exacte ». Dans le but de simuler ou commander un processus, on doit rechercher le modèle qui décrit ce processus le plus parfaitement possible. La littérature abonde de techniques d'identification qui sont utilisées parfois intuitivement selon le domaine appliqué, en fonction du protocole d'acquisition de mesures ainsi que la nature du processus étudié : linéaire ou non linéaire, déterministe ou stochastique. L'identification des paramètres d'un modèle mathématique est donc une étape clé pour élaborer un modèle qui décrit le plus fidèlement possible un système dynamique puis réussir les commandes basées sur ce modèle. Les techniques les plus utilisées sont les moindres carrés ou leurs variantes, techniques très attrayantes pour leur facilité d'utilisation, rapidité et le faible coût de calcul. Cependant elles présentent des problèmes de convergence et un biais d'estimation dû à des perturbations causées par les erreurs de modélisation et aux bruits de mesures. Selon E.Walter[95], on ne peut plus compter que sur les méthodes classiques d'identification et ceci pour plusieurs raisons : la minimisation d'une fonction de coût se fait sur des hypothèses parfois trop simplifiées :

1. l'utilisation de techniques itératives locales pour l'estimation de paramètres de systèmes non linéaires peut nous piéger vers des minimum locaux à cause d'un mauvais choix de valeurs initiales ou du critère d'arrêt. Certaines méthodes d'identifications, comme des algorithmes génétique ou recuit simulé, peuvent pallier à ce problème mais sans pour autant garantir le résultat
2. En identification les algorithmes itératifs utilisent le calcul du gradient de la fonction de coût qui se fait parfois par différence finie ce qui peut compliquer les calculs et ne fournir à la fin qu'un résultat approximatif.
3. Quand on résout une équation différentielle non linéaire par des méthodes numériques, on ne peut rien conclure sur l'erreur globale commise sur la solution obtenue. Une erreur qui peut être considérable d'autant plus si l'on se place dans le cadre de paramètres et de conditions initiales incertaines.
4. L'erreur de faire trop souvent l'hypothèse d'une erreur gaussienne qui découle du théorème de la limite centrale. Grâce à ces hypothèses, les méthodes de maximum de vraisemblances sont appliquées pour minimiser la fonction de coût quadratique. Cependant, ces hypothèses restent non réalistes à cause du caractère approximatif du modèle utilisé, aux pannes des capteurs qui peuvent provoquer l'introduction de données aberrantes pouvant avoir des conséquences catastrophiques.
5. les formules mathématiques classiques sont parfois inadaptées pour la mise en œuvre informatique. En effet, utiliser la solution optimale obtenue par minimisation de la fonction de coût au sens des moindres carrés pour l'implémentation numérique est à éviter. Ceci est dû au mauvais conditionnement du régresseur. Il est plus correct de passer par une factorisation QR ou à une décomposition en valeurs singulières. D'autre part, Les calculs effectués en virgule flottante par les ordinateurs introduisent des erreurs d'arrondis qui se propagent jusqu'à fausser le résultat final. Beaucoup d'exemples de bugs informatiques liés à ce problème et ayant eu des répercussions catastrophiques sont cités dans la littérature [1]. Parmi eux, le fait marquant de la destruction de la fusée Ariane en 1996, le nouveau lanceur implanté a dévié de sa trajectoire puis explosé à cause du logiciel utilisé. En effet, l'ordinateur ne peut fonctionner que sur un nombre fini de bits, tous les autres ne sont qu'approchés, cependant quand il effectue trop d'opérations, les erreurs d'arrondi s'accumulent et le résultat final

devient trop loin de celui escompté. C'est cette même raison qui a provoqué la catastrophe pendant la guerre du golf lorsqu'un missile a dévié de sa trajectoire et a frappé les troupes américaines

Pour mieux comprendre les défaillances de l'arithmétique flottante S.M.Rump [91] a exposé l'exemple suivant :

Considérons la fonction suivante :

$$f(a, b) = 333.75b^6 + a^2(11a^2b^2 - b^6 - 121b^4 - 2) + 5.5b^8 + \frac{a}{2b}$$

le calcul de cette fonction pour $a = 77617$ et $b = 33096$ pour différentes précisions nous donne les résultats suivants :

Précision	Résultat
Simple précision	1.172603
Double précision	1.1726039400531
Précision étendu	1.172603940053178

En examinant ces trois précisions, on remarque que les sept premiers chiffres sont pareils. On peut éventuellement conclure que le résultat est exact. Ce n'est pourtant pas le cas car le résultat est complètement faux pour les trois cas. La valeur correcte de f est :

$$f = -0.8273960599468213$$

En effet, pour obtenir la valeur exacte, il faut factoriser correctement la fonction f et l'évaluer en utilisant des entiers, la méthode plus détaillé est donné par E.Loh[61] C'est toutes ces raisons qui ont poussé la communauté scientifique à rechercher d'autres techniques d'identification plus robustes. Par conséquent, la nécessité d'introduire l'arithmétique des intervalles s'est fait ressentir lorsque l'on a remarqué les défaillances concernant l'arithmétique flottante. Dans ce contexte une variable est représentée uniquement par un ensemble \mathbb{X} , appelé ensemble de vraisemblance ou domaine, supposé contenir la valeur réelle x

En vérité, l'arithmétique des intervalles remonte à beaucoup plus loin, quand Archimède réussit à encadrer le nombre pi dans un intervalle. Plus tard, en 1931 dans

sa thèse, C.YOUNG[98] a utilisé l'arithmétique des intervalles pour le calcul des intégrales de Riemann. En 1958, le japonais T.SANUGA[93] a appliqué ces techniques en analyse numérique. Cependant, Les bases de cette arithmétique ont été fondées par R. MOORE[68; 70; 69] puis NEUMAIER[74] et HANSEN[38]. C'est en Allemagne, vers 1980 que l'arithmétique des intervalles connu un réel progrès quand l'entreprise Nixdorf l'a implémenté sur calculateur grâce à UW Kulisch [58], puis il fut suivi par l'entreprise IBM qui a mit au point un compilateur. C'est en cette même période qu'est née la norme IEEE 754 qui a facilité l'implémentation de l'arithmétique des intervalles. Les méthodes ensemblistes trouvent actuellement des applications dans plusieurs domaines. On peut citer des applications en électrochimie, dans le but d'un étalonnage, on a utilisé ces techniques pour l'estimation de paramètres incertains d'un dispositif de mesures de grandeurs thermiques [18; 19; 16], beaucoup plus récemment des approches ensemblistes ont permis l'analyse et une estimation plus robuste de systèmes hybrides non linéaires [64; 85; 83]. Il existe aussi des applications dans le domaine de la robotique pour la résolution de problèmes d'étalonnage [10], d'estimation et d'identification [50; 78; 87; 86], ainsi que les planifications de trajectoires et la localisation [54; 47; 46]. Plus connue par méthode d'estimation dans le contexte à erreurs bornées [82; 88; 65; 66], l'approche ensembliste permet de résoudre en automatique des problèmes d'estimations non linéaire dont le traitement par une approche probabiliste reste difficile voir impossible. Dans cette thèse nous allons tester deux techniques d'estimation : L'intégration numérique d'équations différentielles ordinaires en présence d'incertitudes [64; 85; 73; 55]. Le but est de déterminer des trajectoires englobantes qui encadrent toutes les solutions de manière garantie. Pour l'identification des paramètres, on associe l'inversion ensembliste à l'intégration numérique garantie. La deuxième approche consiste à utiliser une méthode appelé LSCR(Leave-out Sign-dominant Correlation Regions), développée par Campi [20], permettant d'obtenir une région de confiance qui doit satisfaire deux conditions : avoir une probabilité garantie et être concentré autour du paramètre souhaité. La région de confiance obtenue est exacte et non asymptotique. Cette méthode est intéressante car elle ne nécessite pas beaucoup d'hypothèses sur le bruit pour la construction de régions de confiance et reste efficace même en présence des dynamiques non modélisées. La théorie de cette approche est aussi développée dans [21] et [22]. L'inconvénient de cette approche est qu'elle nous fournit des résultats

par maillage et ne nous procure pas un résultat numérique garanti. Pour y remédier, M.Kieffer [56] combine l'inversion ensembliste à cette approche afin de caractériser ces régions de manière garantie. On ne peut négliger dans cette étude, l'utilisation de la technique des moindres carrés afin de procéder à une comparaison avec les méthodes ensemblistes.

Notre étude s'applique au domaine de la biotechnologie, nous nous intéressons à l'identification des paramètres d'un modèle de bioprocédé aérobie de traitement des eaux usées. Parmi les premiers modèles décrivant ce processus, on peut citer ceux décrit par l'association internationale de la recherche sur la pollution des eaux IAW [24] comme le modèle à boues activées [41] utilisant des cinétiques de Monod. C'est un modèle complexe comprenant beaucoup de paramètres dont l'identification par les méthodes statistiques s'avère difficile voir impossible. C'est des modèles inadaptés pour un control en ligne. Un modèle plus réduit [51] suggère l'utilisation d'un modèle déterministe avec des entrées stochastiques pour un control optimal. Une commande adaptative non linéaire d'un processus de nitrification est considéré dans [77]. On peut citer d'autres modèles pour le contrôle en ligne de processus de boues activées utilisant des modèles ARMAX [76], la logique floue [27] et les réseaux de neurones [12; 31]. Les deux dernières méthodes permettent une manipulation plus aisée des non linéarités. Des auteurs [92] proposèrent d'identifier des paramètres d'un chémostat à l'aide d'algorithmes génétiques. La synthèse d'observateurs intervalles pour les modèles incertains a été étudié dans [35; 36; 37]. Les méthodes ensemblistes sont bien adaptées pour les applications en biotechnologie en raison de l'intervalle d'erreur important introduit pendant l'acquisition des mesures. Cette thèse est structurée comme suit :

Le **chapitre 1** sera consacré à la description du dispositif expérimental utilisé pour les différentes simulations testées. C'est un modèle de chémostat qui réalise le processus de nitrification et qui se trouve au laboratoire de biotechnologie de l'environnement (LBE), INRA, Narbonne. Dans un premier temps, on procède d'abord à la description du principe de traitement des eaux par boues activées, puis le traitement biologique de l'azote, pour aboutir aux équations décrivant les étapes de la nitrification et les bactéries mises en jeu.

Un modèle mathématique de chémostat modélisant deux populations de bactéries

est présenté par la suite.

Le **chapitre 2** est dédié à la définition des outils et aux principes de bases de l'arithmétique des intervalles. Nous présentons aussi les méthodes ensemblistes, particulièrement l'inversion ensembliste utilisée pour l'identification des paramètres du bioréacteur. Les contracteurs seront introduit, ils seront associés à l'inversion ensembliste afin de diminuer le pessimisme, notion qui sera aussi définie dans ce chapitre. Le principe de l'identification dans le contexte à erreurs bornées sera présenté dans le **chapitre 3**. Nous commençons par rappeler les méthodes classiques d'identification ainsi que les études statistiques sur lesquelles elles sont basées. Nous procédons par la suite à une comparaison des méthodes probabilistes avec les méthodes ensemblistes. Nous terminons ce chapitre par un exemple de l'identification dans le contexte à erreurs bornées appliquée à l'identification des paramètres d'un robot scara.

Le **chapitre 4** présente notre contribution dans cette thèse et qui consiste à l'identification des paramètres d'un chémostat en utilisant les techniques ensemblistes associées à l'arithmétique des intervalles. Nous avons testé deux méthodes, la première est l'intégration numérique garantie utilisant la théorie de comparaison des inégalités différentielles, pour cela nous avons besoin de définir les notions de systèmes monotones, les systèmes coopératifs et décrire le théorème de Müller. La deuxième partie sera consacrée à la description de la méthode LSCR (Leave-out Sign-dominant Correlation Regions) utilisant les fonctions de corrélation. Cette méthode sera ensuite associée à l'inversion ensembliste en vue de garantir le résultat. On terminera par une conclusion générale.

Notations

\mathbb{R} : L'ensemble des nombres réels. (Tous les ensembles sont désignés par des caractères doublés \mathbb{A} , \mathbb{B} ,...).

\mathbb{R}^n : L'ensemble des vecteurs ou les fonctions vectorielles

$\mathbb{I}\mathbb{R}$: Ensemble de tous les intervalles de \mathbb{R}

$\mathbb{I}\mathbb{R}^n$: Ensemble de toutes les boîtes de \mathbb{R}^n

NH_3 : Ammoniaque

NH_4^+ : Ammonium

N_2 : Azote gazeux

X : Concentration en biomasse $gDCO.L^{-1}$

S : Concentration en substrat $mg.L^{-1}$

V : volume dans la chambre du réacteur. (m^3)

D : Taux de dilution h^{-1}

Y : Coefficient de rendement

TSH : Le temps de séjour hydraulique.

Q_{in} : Débit d'entrée.

Q_{out} : Débit de sortie.

S_{in} : Concentration en substrat initiale $mg.L^{-1}$

K_S : Constante de Michaelis-Menten ou constante de demi saturation (constante d'affinité pour le substrat S) $mgS.L^{-1}$

μ : Taux de croissance spécifique j^{-1}

μ_{max} : Taux de croissance spécifique maximal j^{-1}

AOB, XB : Bactéries Ammonium Oxydantes (nitritante) $mgDCOL^{-1}$

NOB, XB : Bactéries Nitrite Oxydante (nitratante) $mgDCOL^{-1}$

k_1 : Rendement de dégradation de S_1 par X_1

k_2 : Rendement de production de S_2 par X_1 à partir de S_1

$\mu_1(S_1)$: Taux de croissance spécifique de X_1 sur S_1 (d^{-1})

$\mu_2(S_2)$: Taux de croissance spécifique de X_2 sur S_2 (d^{-1})

k_{s1} : La constante de demi-saturation pour la croissance correspondant à l'affinité des cellules pour le substrat S_1 .

k_{s2} : La constante de demi-saturation pour la croissance correspondant à l'affinité des cellules pour le substrat S_2 .

EDO : Equations différentielles ordinaires.

Chapitre 1

Modèle du Bioréacteur pour la Nitrification



« Et nous avons produit à partir de l'eau toute chose vivante »

Noble coran (*Sourate des Prophètes, v30*)

1.1 Introduction



FIGURE 1.1 – L'eutrophisation des eaux

L'eau c'est la vie, malheureusement des images telles que celle de la figure 1.1¹, deviennent de plus en plus fréquentes. La cause est due aux mauvaises habitudes de l'homme. En effet, régulièrement des eaux usées sont déversées dans les rivières et les

1. <http://www.servimg.com/view/11558293/1746>

fleuves. Elles proviennent de rejets domestiques contenant des matières organiques, minérales et des graisses, des matières difficilement biodégradables (le plastique met presque 1000 ans pour se désagréger naturellement). D'autres rejets proviennent des commerces, des hôpitaux et de l'industrie, . . . qui peuvent déverser des matières toxiques et des métaux lourds. Tous ces rejets provoquent un excès de phosphore, de nitrite, et un manque d'oxygène qui sont responsables de l'eutrophisation des eaux figure 1.1.

1.2 Traitement des eaux par boues activées

Des stations d'épuration des eaux usées figure 1.2² ont été construites pour traiter des effluents domestiques ou industriels, en se basant sur un traitement physico-chimique puis biologique.

Le principe étudié dans cette thèse est le traitement par boues activées, procédé dû à J.P CANLER [23] et qui consiste en une suspension boueuse contenant des populations bactériennes épuratrices des bassins d'aération.

Le traitement des eaux usées par boues activées dans une station d'épuration suit les étapes suivantes :

Relevage et dégrillage : élimination de gros déchets (branches, sachets, bouteille. . . etc).

Dessablage : extraction du sable et cailloux.

Déshuilage : récupération des huiles et des matières grasses.

Traitement biologique des eaux usées : c'est cette phase qu'on étudie. L'action des bactéries ou biomasse qui consiste à se nourrir petit à petit de certaines matières polluantes appelées substrats.

Traitement des boues : extraction d'une partie des bactéries sous forme de «boues » dans l'eau pour incinération en usine.

continuité du cycle de traitement : par réinjection de l'autre partie de bactéries dans les bassins de traitement.

Rejet de l'eau épurée : l'eau propre même si elle n'est pas potable, peut être rejetée dans les rivières et les fleuves.

2. <https://www.u-picardie.fr/beauchamp/conférences/assainissement-coll.html>

Tous les déchets, sable, cailloux, huiles, graisses et boues sont envoyés dans des centres de traitements ou d'incinération.

Afin de mieux comprendre le fonctionnement de dégradation biologique, les scientifiques ont élaboré des bioréacteurs qu'on va décrire en détail dans ce chapitre.

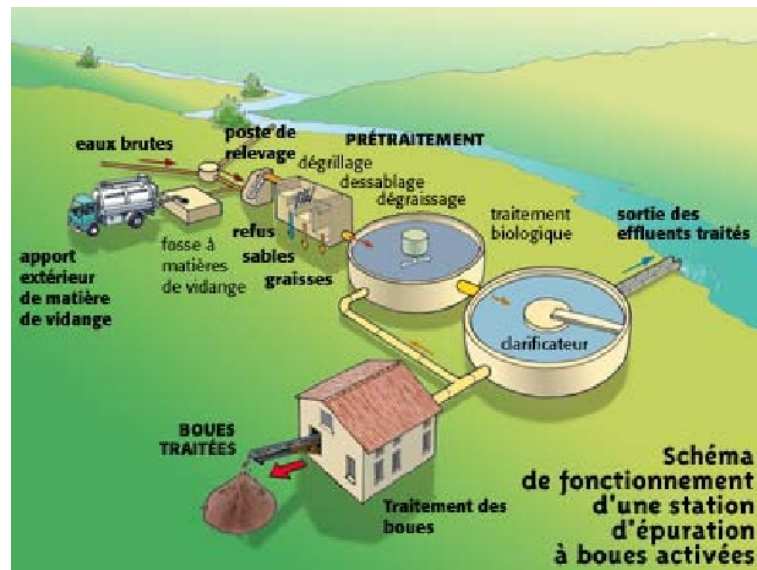


FIGURE 1.2 – Station d'épuration des eaux usées

1.3 Cycle de l'azote

L'azote est présent dans toutes les réactions biologiques de croissance, d'autant plus qu'il représente plus de 78 % de l'atmosphère gazeuse. Le cycle de l'azote fait intervenir un ensemble de réactions : de fixation, d'assimilation, d'ammonification, de nitrification, et de dénitrification. L'azote est considéré comme une importante source de pollution, avec le carbone et le phosphore. On le retrouve dissous dans l'eau sous forme d'ammoniac, de nitrite ou de nitrate, une trop forte concentration d'azote dans les eaux des rivières est responsable de beaucoup de problèmes qui peuvent se répercuter sur la santé de l'homme. Il devient donc impératif de traiter l'azote dans les procédés de dépollution des eaux usées.

1.4 Traitement biologique de l'azote

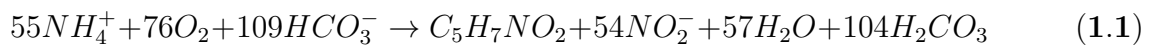
Dans les eaux usées, l'azote organique N_2 se transforme automatiquement en azote ammoniacal (NH_4^+), c'est l'ammonification. Le traitement de l'eau consiste

à éliminer cet azote ammoniacal. Pour se faire, on procède à des traitements biologiques qui sont effectués à l'aide de certaines populations de bactéries sous des contraintes environnementales comme la chaleur, l'oxygène et le pH de l'eau. Le processus s'effectue par étapes grâce à des réactions de nitrification-dénitrification. La transformation chimique de l'azote ammoniacal est décrite par la figure 1.3

1.5 La nitrification

La nitrification est un processus biologique responsable de la production de nitrates dans l'environnement [13]. Elle est réalisée en aérobie par des micro organismes autotrophes et s'effectue en deux étapes :

Nitrification : ou (Nitrosation) qui correspond à l'oxydation de l'ammonium en nitrite due à des bactéries nitrosantes qu'on nomme aussi AOB (*Ammonia Oxidizing Bacteria*) [80]. La réaction stœchiométrique donnée par la littérature [15].



Nitratation : correspondant à l'oxydation des nitrites en nitrates par des bactéries nitratantes appelées aussi NOB (*Nitrite Oxidizing Bacteria*) [80] et dont la réaction est :

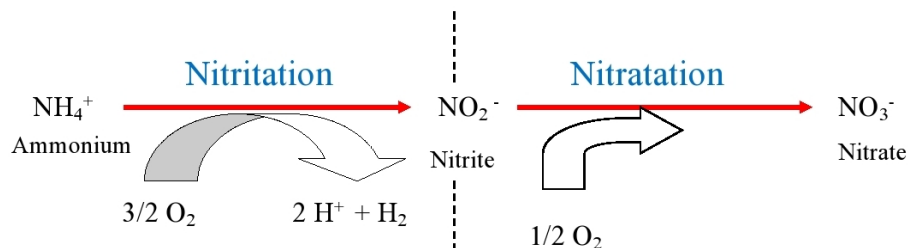
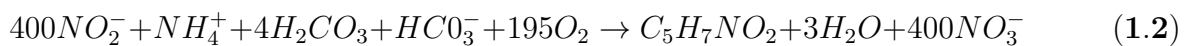


FIGURE 1.3 – Les deux étapes de la nitrification

1.5.1 Les bactéries nitrifiantes

Les micro-organismes responsables de la réaction de nitrification ont été identifiés par Winogradsky (1890) [97]. Il s'agit de deux types de bactéries : Nitrosomonas, capable d'oxyder l'ammonium en nitrite et Nitrobacter qui transforme les nitrites en nitrates pendant l'étape de la nitrification. C'est des bactéries lithotrophes, c'est-à-dire des bactéries autotrophes qui utilisent comme source d'énergie pour leur croissance, des éléments inorganiques comme l'azote minérale.

1.5.2 Facteurs influents sur la nitrification

L'accroissement ou l'inhibition des bactéries nitrifiantes dépend fortement de certains facteurs environnementaux tel que le pH, la température, ou la quantité d'oxygène.

Influence du pH : Au début de la réaction de nitrification, la libération des ions H^+ , provoque l'acidification du milieu donc une diminution du pH [8]. La valeur optimale du pH se situe entre 7 et 8. En dehors de ces valeurs, la réaction de nitrification risque d'être inefficace.

Influence de la température : Que se soit pour Nitrosomonas ou Nitrobacter, la vitesse maximale de croissance et l'affinité pour le substrat varient avec la température. La stratégie du procédé SHARON (Single reactor system for High activity Ammonia Removal Over Nitrite) [40], situe la température optimale de croissance de Nitrosomonas $30^{\circ}C$ et $36^{\circ}C$ alors que pour Nitrobacter, elle se situe entre $28^{\circ}C$ et $36^{\circ}C$.

Influence de l'oxygène : Une réaction d'oxydation nécessite forcément la présence d'oxygène. Il n'est pas indispensable d'avoir des concentrations élevées en oxygène dissous dans un milieu nitrifiant, $4mg/l$ sont suffisants sauf dans le cas de compétition entre différents types de bactéries [32]. Les constantes de demi-saturation K_S représente l'affinité plus au moins importante pour l'oxygène en fonction des espèces. Par exemple, Nitrobacter peut être sensible à des faibles concentrations d'oxygène dissous ce qui n'est pas le cas de Nitrosomonas.

Les bactéries nitrifiantes possèdent une forte capacité d'adaptation dans un milieu, cependant la nitrification peut être inhibée en présence de certains composés comme les métaux lourds (*zinc, cuivre, cobalt...*) des phénols ou des amines.

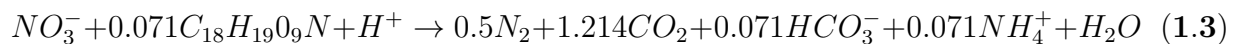
Remarque

Le taux de croissance spécifique des bactéries nitrifiantes est faible et le temps de génération est très long : de 7 à 24h pour les bactéries nitritantes, de 10 à 140h pour les nitratantes [14].

Les rendements et concentrations cellulaires résultants des réactions d'oxydation d'ammonium et de nitrites sont faibles même dans le cas de taux élevés.

1.5.3 La dénitrification

La dénitrification est un processus anoxique qui retourne l'azote sous sa forme moléculaire N_2 vers l'atmosphère après réduction des nitrates par des bactéries hétérotrophes aérobies. Les bactéries hétérotrophes sont des micro organismes incapables d'effectuer eux mêmes les synthèses de leurs constituants à partir d'éléments minéraux. L'équation stœchiométrique de la réaction biologique est :

**1.6 Les réacteurs biologiques**

Les bioréacteurs de laboratoires figure 1.4 sont des dispositifs qui permettent d'étudier la croissance de micro-organismes. Leurs principales composantes sont :

Un substrat S qui est un milieu de culture contenu dans la chambre du bioréacteur nécessaire à la croissance des micro-organismes et contrôlé par un flux de matière entrant S_{in}

Une biomasse X constituée de bactéries, ou phytoplancton ou bien encore de champignons. Dans le cadre d'un procédé de dépollution, elle peut contenir des centaines d'espèces.

Les dérivés synthétisés par les micro-organismes, résultats des réactions biochimiques sous l'influence de la température, le pH, l'oxygène, ... et de certains catalyseurs parfois.

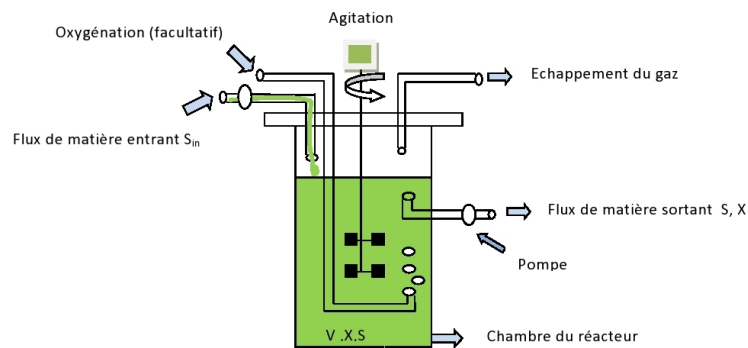


FIGURE 1.4 – Le Bioréacteur

1.7 Mode de fonctionnement d'un bioréacteur

Les débits d'entrée notés Q_{in} et de sortie notés Q_{out} d'un bioréacteur peuvent être choisis par l'expérimentateur ou peuvent résulter d'une loi de contrôle dépendant de l'état du système. On distingue trois grands modes d'alimentation.

1.7.1 Le fonctionnement discontinu

En fonctionnement discontinu ou (batch), Les espèces sont introduites à l'instant initial (inoculation) puis le bioréacteur est maintenu fermé et le volume constant ce qui produit une croissance exponentielle des organismes. Cette méthode permet d'éviter la contamination par d'autres micro-organismes. La durée d'une expérience dans un réacteur batch est limitée. Ce type de bioréacteur est plus utilisé dans l'industrie pour la production d'un métabolite ou de biomasse.

1.7.2 Le fonctionnement semi-discontinu

En fonctionnement semi-discontinu ou (fed batch), il y a le flux entrant de matière mais la sortie est nulle durant le temps de l'expérience qui est limité. Ce type de fonctionnement est utilisé pour le contrôle en boucle fermée du débit d'entrée afin d'étudier la croissance des micro-organismes.

1.7.3 Le fonctionnement en continu

Un bioréacteur en fonctionnement continu est aussi appelé chémostat. Son concepteur est Szilárd [75] qui fut le premier à introduire ce réacteur dans le cadre

de son étude sur la croissance de *Escherichia coli* en collaboration avec J.Monod [67]. Les chémostats sont les plus utilisés dans le domaine de l'épuration des eaux usées. Leurs caractéristiques sont : un volume constant dans la chambre du réacteur, c'est dire que le débit d'entrée est égal au débit de sortie.

La quantité de nutriments introduite par heure divisée par le volume de la chambre du réacteur est connue comme taux de dilution D . On peut donc ajuster la dilution en fonction de l'évolution des conditions dans le chémostat.

Dans les théories classiques des chémostats, quand le débit d'entrée est fixe, une espèce de bactérie va éliminer toutes celles qui cohabitent avec elle dans la chambre du réacteur : c'est le principe de l'exclusion compétitive. Cette théorie est contredite par les écosystèmes où plusieurs espèces peuvent cohabiter pendant un long terme. C'est le principe de compétition des espèces.

1.8 Dynamique des bactéries en chémostat

La modélisation d'un chémostat comporte deux parties, la partie physique qui tient compte de la circulation des matières due aux flux entrants et aux flux sortants puis aussi de la partie processus biologique à l'intérieur du chémostat. La modélisation de ce processus respecte la loi de conservation de LAVOISIER : « Rien ne se perd, rien ne se crée, tout se transforme ».

1.8.1 Mise en équation

Les équations qui régissent le fonctionnement du chémostat sont obtenues en faisant le bilan des matières d'entrées et des sorties.

Le taux de croissance μ est décrit par un modèle dit de Monod [67] qui s'exprime en fonction de la concentration du substrat S comme suit :

$$\mu = \mu_{max} \cdot \frac{S}{S+K_S} \quad (1.4)$$

La constante μ_{max} est la vitesse de croissance spécifique maximale et La constante K_S traduit l'affinité de la cellule pour le substrat S . Elle est appelée

constante de demi-saturation pour la croissance.

La concentration de l'un des constituants du milieu peut limiter la croissance des micro-organismes (facteur limitant). Dans un chémostat la concentration du substrat limitant est S (g/L). L'augmentation de la concentration du substrat est due à l'apport de milieu S_{in} .

$$\left(\frac{dS}{dt}\right)_{in} = D \cdot S_{in} \quad (1.5)$$

D représente le taux de dilution. La diminution de la concentration du substrat due à la sortie est :

$$\left(\frac{dS}{dt}\right)_{out} = -D \cdot S \quad (1.6)$$

Après consommation par les micro-organismes, il y'a diminution de la concentration du substrat :

$$\left(\frac{dS}{dt}\right)_{cons} = -\mu \cdot \frac{X}{Y} \quad (1.7)$$

Y représente le rendement de la transformation de substrat en biomasse. Il mesure la quantité de biomasse produite par unité de substrat consommé, On le définit aussi par :

$$\left(\frac{X}{S}\right)_{in} = Y \quad (1.8)$$

La concentration de la biomasse dans le chémostat est X (g/L). Son augmentation est proportionnelle à la biomasse. Elle est due à la croissance des micro-organismes

$$\left(\frac{dX}{dt}\right)_{croiss} = \mu \cdot X \quad (1.9)$$

D'autre part, La diminution de la biomasse due à la sortie est décrite par :

$$\left(\frac{dX}{dt}\right)_{out} = -D.X \quad (1.10)$$

avec $D = \frac{Q}{V}$, le taux de dilution appelé aussi taux de renouvellement. Q représente le volume de milieu qui s'écoule à l'intérieur et à l'extérieur du réacteur par unité de temps. V est le volume occupé par le milieu de culture.

1.8.2 Modèle de Chémostat avec cinétique de Monod

Il existe de nombreuses équations algébriques qui ont été établies pour décrire la fonction de croissance des microorganismes. Certaines tiennent compte de l'influence de la température ou du pH [43; 11]. Le modèle de Cantois [26] considère que la fonction de croissance dépend du ratio de substrat par organisme. Dans le cas de l'inhibition de la croissance causée par l'accumulation dans le milieu de produits métaboliques un modèle de Haldane est proposé puis repris par Andrews [7] est proposé.

Dans le cadre de cette thèse, on a utilisé le modèle de Monod qui fut le premier à modéliser une population microbienne. En se basant sur le bilan de matière et les équations décrites au paragraphe 1.8.1, le modèle du chémostat basé sur une cinétique de Monod est décrit par les équations suivantes :

$$\frac{dX}{dt} = X.(\mu - D) \quad (1.11)$$

$$\frac{dS}{dt} = D.(S_{in} - S) - \mu.\frac{X}{Y} \quad (1.12)$$

$$\mu = \mu_{max}.\frac{S}{S+K_S} \quad (1.13)$$

Parmi toutes les fonctions de croissance, celle de Monod reste la plus utilisée car mathématiquement, ses paramètres sont plus simples à exploiter.

Analyse de la fonction de croissance de MONOD

Les différents états stationnaires sont obtenus par les équations suivantes :

$$\frac{dX}{dt} = X.(\mu(s) - D) = 0 \quad (1.14)$$

Donc pour $X = 0$ ou $\mu(s) = D$

Un chémostat vide de tout micro organismes, représente le premier état stationnaire.

Ce qui correspond un état en substrat suivant :

$$D.(S_{in} - S) = 0 \text{ et donc, } S_{in} = S \quad (1.15)$$

Ce qui prouve qu'en absence de la biomasse, il n'y a pas de consommation de substrat qui reste égale à la valeur d'entrée. Un autre état stationnaire est atteint pour la valeur $\mu(s) = D$, correspondant à un état de substrat :

$$\mu_{max} \frac{S}{S+K_S} = D \text{ et par conséquent, } S = \frac{D.K_S}{\mu_{max}-D} \quad (1.16)$$

Et un état en biomasse défini par :

$$\frac{dS}{dt} = D.(S_{in} - S) - \mu \cdot \frac{X}{Y} = 0 \text{ d'où } X = Y(S_{in} - S) \quad (1.17)$$

Ce qui correspond à un état d'équilibre car l'accroissement de la biomasse diminue puis s'arrête avec le temps.

Remarque

Au début de l'expérimentation, il est peu probable d'obtenir un état d'équilibre. En effet :

1. Si $\mu(S) < D$, il y'a une diminution progressive de la concentration cellulaire dans le bioréacteur, ce qui produit une augmentation de la concentration en nutriments et par conséquent une augmentation du taux de croissance jusqu'à une valeur maximale proche de la valeur de D.
2. Si $\mu(S) > D$, il y a diminution de la concentration en substrat, ce qui provoquera une diminution de la croissance $\mu(S)$ qui tend par la valeur de D.

3. Si la valeur de D est très élevée, la biomasse va diminuer jusqu'à disparaître complètement du Bioréacteur

1.9 Dispositif expérimental

Le dispositif expérimental qui réalise le processus de nitrification sur lequel s'est portée notre étude est situé au laboratoire de biotechnologie de l'environnement (LBE), INRA, Narbonne [30]. Ce dispositif est constitué de cinq réacteurs biologiques fonctionnant en continu. On ne s'intéresse qu'aux deux premiers des réacteurs, identiques car soumis aux mêmes conditions et à la même entrée.

Le milieu de culture utilisé est constitué d'une solution minérale très concentrée en azote ammoniacal ($0,5$ à $2gN.L^{-1}$).

Les différents composés du milieu sont dissous avec de l'eau déminéralisée. La solution obtenue est tamponnée par des bicarbonates afin de minimiser les variations de pH qui est fixé à 7. Les températures sont maintenues basses et le débit d'entrée constant. On définit le temps de séjour hydraulique (TSH) comme la durée pendant laquelle une bactérie est en contact avec le milieu de culture dans la chambre du réacteur puis éliminée par une pompe de soutirage. C'est aussi le rapport du volume utile sur le débit exprimé en heure ou jour.

Un TSH de huit jours a été choisi pour les réacteurs afin d'éviter le lessivage (cas de purge cellulaire supérieure à l'augmentation de la biomasse) des bactéries nitrifiantes à cinétique trop rapide au profit des bactéries nitrifiantes dont la cinétique est plus lente, ce qui permet d'étudier chaque étape de la nitrification. Les espèces nitrifiantes présentes dans les chémostats ainsi que leur quantification relative ont été mises en évidence grâce à des analyses microbiologiques. D'autre part, la technique de mesure de la masse sèche [30] permet la quantification de la biomasse totale présente dans les chémostats. Pour plus de détails sur le dispositif de expérimental se référer à [30].

1.9.1 Modélisation de plusieurs populations en chémostat

Des analyses chimiques et physiques régulières sont réalisées sur les deux chémostats A et B, ce qui a permis d'obtenir de précieux renseignements sur la dynamique de la biomasse ainsi que sur toutes les composantes intervenants dans la réaction de

nitrification.

En se basant sur ces cinétiques et à partir d'un premier bilan de matière, un modèle dynamique dit « macroscopique » basé sur le modèle de Monod est obtenu. Il décrit le cas de compétition entre deux populations : l'oxydation de l'azote ammoniacal en nitrite par des bactéries ammonium-oxydantes (AOB) représentées par la variable X_1 et l'oxydation du nitrite en nitrate par des bactéries nitrite-oxydantes (NOB), représentées par la variable X_2 . Par la suite, la synthèse d'un observateur a permis d'estimer les concentrations de biomasse relatives aux deux types de bactéries.

1.9.2 Modèle mathématique

Les bioréacteurs sont considérés comme des systèmes dynamiques avec des entrées (débit, concentrations ...) et des sorties (concentrations de composants de la réaction) bien définies.

Pour représenter donc l'état de ce bioréacteur, six variables sont nécessaires : les concentrations de bactéries, appelées biomasses et désignés par X_1 et X_2 , ainsi que les concentrations de substrat S_1 , S_2 et S_3 , sachant que la nitritation est réalisée par des bactéries nitritantes X_1 qui transforme azote ammoniacal (S_1) en nitrites (S_2) et que la nitrification est réalisée par les bactéries X_2 qui transforme les nitrites (S_2) en nitrates (S_3). S_{in} est la concentration du substrat entrant . Le modèle ci-dessous représente l'évolution dynamique de ces variables d'état[30] .

$$\begin{aligned}
 \dot{S}_1 &= D(S_{in} - S_1) - k_1\mu_1(S_1)X_1 \\
 \dot{X}_1 &= (\mu_1(S_1) - D)X_1 \\
 \dot{S}_2 &= k_1\mu_1(S_1)X_1 - k_2\mu_2(S_2)X_2 - S_2D \\
 \dot{X}_2 &= (\mu_2(S_2) - D)X_2 \\
 \dot{S}_3 &= k_2\mu_2(S_2)X_2 - S_3D
 \end{aligned} \tag{1.18}$$

$$\text{avec} \quad \mu_1(S_1) = \mu_{max1} \frac{S_1}{(S_1 + k_{S1})}$$

$$\mu_2(S_2) = \mu_{max2} \frac{S_2}{(S_2 + k_{S2})}$$

$$\tag{1.19}$$

$\mu_1(S_1)$ et $\mu_2(S_2)$ sont les taux de croissance de la biomasse. Ils sont modélisés par la cinétique de Monod [67]. $\mu_{max1}(S_1)$ et $\mu_{max2}(S_2)$ sont le taux de croissance spécifique maximum, k_1 et k_2 sont les coefficients stœchiométriques qui représentent les rendements respectifs de biomasses X_1 et X_2 , k_{s1} et k_{s2} sont les constantes de demi-saturation pour la croissance correspondant à l'affinité des cellules pour le substrat de chaque population bactérienne. D est le taux de dilution, tel que $D = \frac{Q_{in}}{V}$, où Q_{in} est le débit d'entrée du bioréacteur et V son volume. Depuis le début de l'expérimentation jusqu'au jour 183, seule la production de nitrite a été réalisée efficacement. La fonction de nitratisation a été obtenue seulement après une diminution de la température de 30 ° C à 25 ° C le jour 183. Cependant l'augmentation de la concentration du substrat d'entrée provoque l'inhibition de la nitratisation.

Remarque

Le modèle mathématique peut être simplifié si l'on introduit un paramètre J_i relatif à chaque population i tel que :

$$J_i = K_{Si} \frac{D}{\mu_{max} - D}, \quad i = 1 \text{ ou } 2, \quad (1.20)$$

L'étude asymptotique du modèle mathématique met en évidence deux états d'équilibre : Le lessivage d'une population ou des deux populations.

Le cas où les deux populations sont lessivés se présente, lorsque la concentration du substrat limitant est inférieure aux deux valeurs de J .

Sinon c'est la population qui présente la plus faible valeur de J qui va persister dans le chémostat.

Nous pouvons conclure qu'il y a coexistence de deux populations à l'équilibre lorsque leur valeur J sont égales ou supérieures à la valeur de la concentration du substrat limitant. Ce qui se traduit mathématiquement par :

$$K_{S1} \frac{D}{\mu_{max1} - D} = K_{S2} \frac{D}{\mu_{max2} - D}, \quad (1.21)$$

Pour des valeurs μ_{max} et K_S considérées fixes, l'équation (1.21) ne va dépendre que du taux de dilution D . Plusieurs cas peuvent se présenter.

1. $\mu_{max1} = \mu_{max2}$ et $K_{S1} = K_{S2}$, l'équation admet une infinité de solutions, ce qui veut dire que les bactéries possèdent les mêmes paramètres de Monod.
2. $\mu_{max1} > \mu_{max2}$ et $K_{S1} > K_{S2}$ ou $\mu_{max1} < \mu_{max2}$ et $K_{S1} < K_{S2}$, l'équation n'admet pas de solutions, donc il n'y a pas de coexistence.
3. $\mu_{max1} < \mu_{max2}$ et $K_{S1} > K_{S2}$ ou $\mu_{max1} > \mu_{max2}$ et $K_{S1} < K_{S2}$, l'équation admet une solution unique de D, cependant il est pratiquement impossible de maintenir un taux de dilution à une valeur constante.

Les valeurs des concentrations des substrats, S_1 , S_2 , et S_{in} ainsi que celle de la biomasse totale $X = X1 + X2$ sont issues de la littérature [30]. Les conditions expérimentales sont décrites avec plus de détails dans [30]. Ces données représentent les connaissances a priori utilisées pour notre procédure de l'identification que nous allons présenter dans les chapitres suivants.

1.10 Conclusion

Dans ce chapitre, nous avons décrit les différentes composantes d'un bioréacteur ainsi que son fonctionnement afin de mieux comprendre le principe du traitement des eaux usées par boues activées, la description du cycle de l'azote et les deux étapes de la nitrification. On s'est particulièrement intéressé au bioréacteur en fonctionnement continu qu'on appelle chémostat. La biomasse est constituée de micro organismes où deux types de bactéries prédominent. Celles qui participent à l'oxydation de l'ammonium : la nitritation et celle qui est responsable de la nitratation. A partir des bilans de matière et l'étude des différentes cinétiques participant à la réaction de nitrification, un premier modèle mathématique est conçu.

Chapitre 2

Analyse ensembliste

2.1 Introduction

La nécessité d'introduire l'arithmétique des intervalles s'est fait ressentir lorsque l'on a remarqué des défaillances concernant l'arithmétique flottante. Les ordinateurs manipulent des nombres réels avec un nombre de bits limités (16 pour les entiers, 64 pour les réels en virgule flottante) donc seul un nombre limité de réels sont représentables exactement en machine, les autres sont approchés. Lorsque des millions d'opérations sont effectuées, les incertitudes s'accumulent et le résultat final peut être très éloigné du résultat exact. Cette préoccupation de déterminer les plages de variation des variables est à l'origine de la naissance de l'arithmétique des intervalles, qui vont nous permettre de manipuler des ensembles.

2.2 Opérations sur les ensembles

Les opérations effectuées avec les ensembles sont de deux types :

2.2.1 Opérations ensemblistes pures

Soit \mathbb{X} et \mathbb{Y} deux sous-ensembles de \mathbb{R}^n . \mathbb{X} est inclus dans \mathbb{Y} si et seulement si tout élément de \mathbb{X} appartient à \mathbb{Y} . Il vient de ce fait :

$$\mathbb{X} \subset \mathbb{Y} \Leftrightarrow (\forall x \in \mathbb{X}, x \in \mathbb{Y}) \tag{2.1}$$

Les opérations suivantes s'appliquent sur des sous-ensembles de \mathbb{R}^n en général. Elles concernent l'union, l'intersection, le produit scalaire et la projection. Étant donnés \mathbb{X} et \mathbb{Y} deux sous-ensembles de \mathbb{R}^n et \mathbb{R}^m , Nous avons :

$$\mathbb{X} \cap \mathbb{Y} \triangleq \{x \in \mathbb{R}^n | x \in \mathbb{X} \text{ et } x \in \mathbb{Y}\} \text{ avec } n=m$$

$$\mathbb{X} \cup \mathbb{Y} \triangleq \{x \in \mathbb{R}^n | x \in \mathbb{X} \text{ ou } x \in \mathbb{Y}\} \text{ avec } n=m \quad (2.2)$$

$$\mathbb{X} \times \mathbb{Y} \triangleq \{(x, y) \in \mathbb{R}^{n+m} | x \in \mathbb{X} \text{ et } y \in \mathbb{Y}\}$$

$$Proj_{\mathbb{X}}(\mathbb{Z}) \triangleq \{x \in \mathbb{X} | \exists y \in \mathbb{Y} \text{ tel que } (x, y) \in \mathbb{Z}\}$$

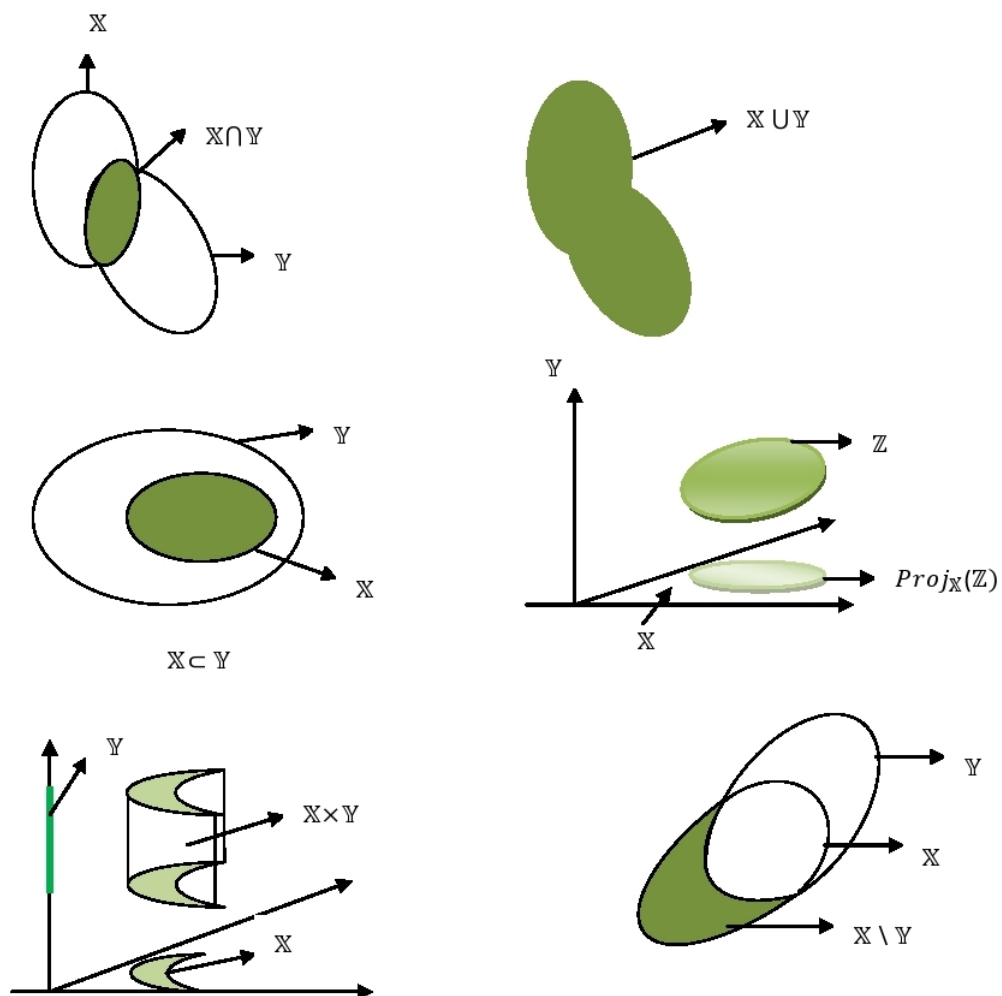


FIGURE 2.1 – Opération ensemblistes pures

2.2.2 Opérations ensemblistes étendues

Soit $\mathcal{P}(\mathbb{X})$ l'ensemble des parties de \mathbb{X} . On définit l'opération \circ étendue sur $\mathcal{P}(\mathbb{X} \times \mathbb{Y})$ tel que :

$$\mathbb{X} \circ \mathbb{Y} \triangleq \{x \circ y \mid x \in \mathbb{X}, y \in \mathbb{Y}\} \quad (2.3)$$

Comme par exemple l'addition :

$$\mathbb{X} + \mathbb{Y} \triangleq \{x + y \mid x \in \mathbb{X}, y \in \mathbb{Y}\} \quad (2.4)$$

Remarque : l'expression $\mathbb{Z} = \mathbb{X} + \mathbb{Y}$ n'est pas équivalente à : $\mathbb{X} = \mathbb{Z} - \mathbb{Y}$

Exemple 2-6

Soit $\mathbb{X} = [1, 0]$, $\mathbb{Y} = [-1, 0]$, $\mathbb{Z} = \{0\}$

$$\mathbb{Z} - \mathbb{Y} = [0] - [-1, 0] = [0, 1] = \mathbb{X}$$

alors que :

$$\mathbb{X} + \mathbb{Y} = [0, 1] + [-1, 0] = [-1, 1] \not\supseteq \mathbb{Z}$$

2.2.3 Représentation d'ensembles

La mise en œuvre informatique des techniques ensemblistes doit s'intéresser comment représenter un ensemble sur machine, puis comment réaliser des opérations en grand nombre sur ces ensembles, puis enfin il faut que tous les éléments de ces ensembles vérifient une même propriété \mathcal{P} . Les approches classiques de représentation sont **la représentation par nuage de point** qui consiste à représenter un ensemble \mathbb{X} par un nuage de points contenu dans \mathbb{X} , On utilise, par exemple la méthode de Monté Carlo ou un balayage déterministe, la méthode est très simple mais reste approximative et donc non garantie. Il y a aussi **la représentation par équations et inéquations**, c'est une méthode qui

permet une manipulation exacte des ensembles, mais utilise des algorithmes assez complexes en plus du coût en temps de calcul et de mémoire utilisée. Il y a enfin **la représentation par recouvrement**, ou l'on dispose d'une classe d'objets qui sont des sous ensembles de \mathbb{R}^n et qu'on nomme récipients. Le but est de pouvoir étudier des ensembles qui nous intéressent contenus dans ces récipients pouvant prendre différentes formes comme les ellipsoïdes, les polytopes, les zonotopes, les pavés... Cette méthode est simple et nous procure des résultats garantis. Dans le cadre de cette thèse, la représentation utilisée est celle par encadrement c'est-à-dire que les récipients utilisés sont des pavés. c'est une représentation aisée en machine et qui consiste à encadrer tout compact \mathbb{X} de \mathbb{R}^n , par deux récipients $\underline{\mathbb{X}}$ et $\overline{\mathbb{X}}$ tel que :

$$\underline{\mathbb{X}} \leq \mathbb{X} \leq \overline{\mathbb{X}} \quad (2.5)$$

On en déduit que :

$$Vol(\underline{\mathbb{X}}) \leq Vol(\mathbb{X}) \leq Vol(\overline{\mathbb{X}}) \quad (2.6)$$

l'encadrement par un pavé nous permet de vérifier une propriété \mathcal{P} de manière garantie sur \mathbb{X} , par exemple si $\underline{\mathbb{X}}$ n'est pas vide alors \mathbb{X} n'est pas vide.

Les pavés représentent la classe de récipients considérée dans la suite de notre étude. Soit \circ une opération de $\mathbb{X} \times \mathbb{Y}$ dans \mathbb{X} , et soient 2 sous-ensembles \mathcal{X} et \mathcal{Y} tels que :

$$\mathcal{X} \circ \mathcal{Y} = \{x \circ y | x \in \mathcal{X}, y \in \mathcal{Y}\} \quad (2.7)$$

l'extension de cette opération pour les récipients nous permet d'écrire :

$$\mathcal{X}[\circ]\mathcal{Y} = [\{x \circ y | x \in \mathcal{X}, y \in \mathcal{Y}\}] = [\mathcal{X} \circ \mathcal{Y}] \quad (2.8)$$

$[\mathbb{X}]$ est le plus petit récipient contenant \mathbb{X} , appelé enveloppe de \mathbb{X} . On définit l'opérateur union \sqcup pour les pavés, tel que :

$$[\mathbb{X}] \sqcup [\mathbb{Y}] = [[\mathbb{X}] \cup [\mathbb{Y}]] \quad (2.9)$$

2.3 Arithmétique des intervalles

Un intervalle, noté par $[x]$ est un ensemble borné et connexe de \mathbb{R} , il est défini par

$$[x] = [\underline{x}, \bar{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \bar{x}\} \quad (2.10)$$

Les nombres réels \underline{x} et \bar{x} sont respectivement les bornes inférieures et supérieures $[x]$, tous les intervalles de \mathbb{R} sont notés \mathbb{IR} , soit $[x] \in \mathbb{IR}$.

On définit :

Sa borne inférieure : $\inf([x]) = \underline{x}$

Sa borne supérieure : $\sup([x]) = \bar{x}$

Sa largeur : $w([x]) = \bar{x} - \underline{x} \geq 0$ (2.11)

Son milieu : $\text{mid}([x]) = \frac{\bar{x} + \underline{x}}{2}$

Son rayon : $\text{rad}([x]) = \frac{\bar{x} - \underline{x}}{2}$

2.3.1 Calcul par Intervalles

Les opérations mathématiques élémentaires sont étendues aux intervalles. Le résultat d'une opération entre deux intervalles de bornes finies est un intervalle obtenu en travaillant seulement sur leurs bornes.

Soit $[x]$ et $[y] \in \mathbb{IR}$, et un opérateur binaire $\circ \in \{+, -, *, /\}$

On définit :

$$[x] \circ [y] = \{x \circ y \mid x \in [x], y \in [y]\} \quad (2.12)$$

On en déduit :

$$[x] + [y] = [\underline{x} + \underline{y}, \bar{x} + \bar{y}]$$

$$[x] - [y] = [\underline{x} - \bar{y}, \bar{x} - \underline{y}]$$

$$[x] * [y] = \left[\min(\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}), \max(\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}) \right] \quad (2.13)$$

$$\frac{[x]}{[y]} = [x] * \frac{1}{[y]}$$

avec :

$$\frac{1}{[y]} \left\{ \begin{array}{l} = \emptyset \text{ si } [x] = [0, 0] \\ = \left[\frac{1}{\bar{y}}, \frac{1}{\underline{y}} \right] \text{ si } 0 \notin [x] \\ = \left[\frac{1}{\bar{y}}, +\infty \right] \text{ si } \underline{x} = 0 \text{ et } \bar{x} > 0 \\ = \left[-\infty, \frac{1}{\underline{y}} \right] \text{ si } \underline{x} < 0 \text{ et } \bar{x} = 0 \\ = \left[-\infty, +\infty \right] \text{ si } \underline{x} < 0 \text{ et } \bar{x} > 0 \end{array} \right.$$

Exemple 2-1

$$[-3, 18] + [-5, 2] = [-8, 20]$$

$$[-10, 17] - [-8, 12] = [-22, 25]$$

$$[-2, 3] * [-1, 5] = [-10, 15]$$

$$[-1, 3] / [2, 5] = [-1/2, 3/2]$$

2.3.2 Définition d'un pavé

On appelle pavé ou vecteur intervalle $[x]$ de \mathbb{R}^n , le produit cartésien de n intervalles :

$$[x] = [\underline{x}_1, \bar{x}_1] \times \dots \times [\underline{x}_n, \bar{x}_n] \quad (2.14)$$

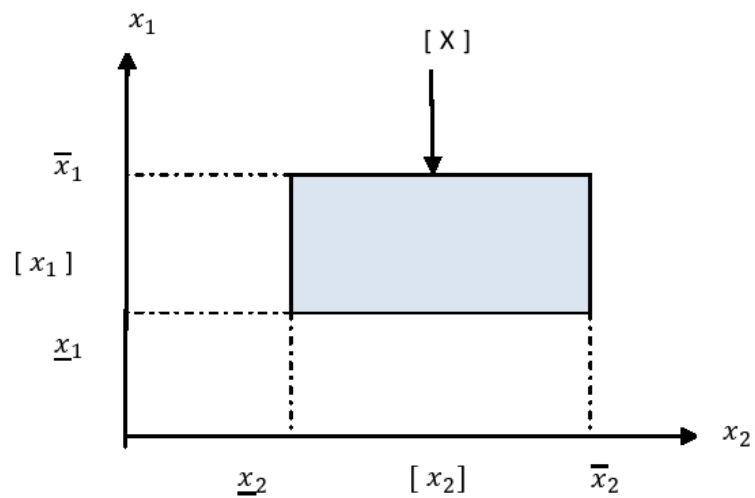


FIGURE 2.2 – Pavé

qui s'écrit aussi :

$$[x] = \begin{pmatrix} [x_1, \bar{x}_1] \\ \vdots \\ [x_n, \bar{x}_n] \end{pmatrix}$$

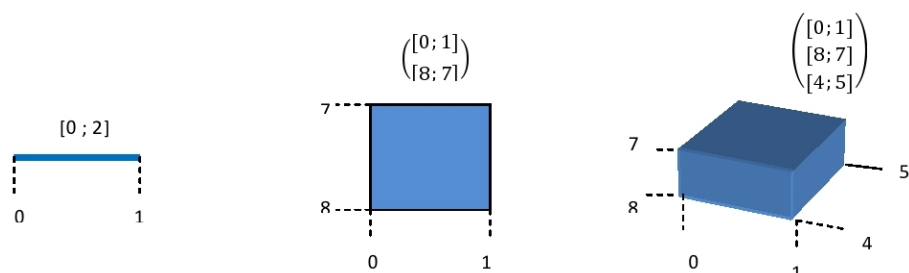


FIGURE 2.3 – Composantes de vecteurs intervalles

La longueur d'un pavé (en anglais : the width) correspond à la plus grande longueur de ses composantes.

$$\omega([x]) = \max_{i=1, \dots, m} \omega([x_i]) \quad (2.15)$$

son centre est défini aussi par :

$$mid([x]) = (mid([x_1]), mid([x_2]), \dots, mid([x_n]))^T \quad (2.16)$$

et son volume par :

$$\text{Vol}([x]) = (\bar{x}_1 - \underline{x}_1) \cdot (\bar{x}_2 - \underline{x}_2) \cdot \dots \cdot (\bar{x}_n - \underline{x}_n) \quad (2.17)$$

Remarque :

Un intervalle dont les deux bornes sont égales est dit dégénéré ie : $\underline{x} = \bar{x} = x$

2.4 Fonction d'inclusion

Considérons deux ensembles \mathbb{X} et \mathbb{Y} ainsi que deux récipients \mathcal{X} et \mathcal{Y} contenant les ensembles \mathbb{X} et \mathbb{Y} respectivement. Soit un sous ensemble \mathbb{X}_1 de \mathbb{X} .

Si $\mathbb{X}_1 \in \mathcal{X}$, alors on peut étendre une fonction $\mathbf{f} : \mathbb{X} \rightarrow \mathbb{Y}$ aux récipients tels que :

$$[\mathbf{f}](\mathbb{X}_1) = [\{\mathbf{f}(\mathbf{x}) | \mathbf{x} \in \mathbb{X}_1\}] \quad (2.18)$$

et encore :

$$[\mathbf{f}](\mathbb{X}_1) \supset \mathbf{f}(\mathbb{X}_1) \quad (2.19)$$

Ce qui introduit la notion de fonction d'inclusion.

Considérons \mathbf{f} une fonction de $\mathbb{R}^n \rightarrow \mathbb{R}^m$, la fonction intervalle $[\mathbf{f}]$ de $\mathbb{I}\mathbb{R}^n \rightarrow \mathbb{I}\mathbb{R}^m$ est une fonction d'inclusion de \mathbf{f} si

$$\forall [\mathbf{x}] \in \mathbb{R}^n, f[\mathbf{x}] \subseteq [\mathbf{f}]([\mathbf{x}])$$

$$\text{ou} \quad [\mathbf{f}]([\mathbf{x}]) \supset \{\mathbf{f}(\mathbf{x}) | \mathbf{x} \in [\mathbf{x}]\} \quad (2.20)$$

Propriétés

La convergence : Une fonction d'inclusion $[\mathbf{f}]$ d'une fonction \mathbf{f} est convergente si pour une suite de pavés $[x]_k$

$$\lim_{l \rightarrow \infty} w([x]_k) = 0 \Rightarrow \lim_{l \rightarrow \infty} w([f][x]_k) = 0 \quad (2.21)$$

La monotonie : Une fonction d'inclusion $[f]$ d'une fonction f est monotone si :

$$[x] \subset [y] \Rightarrow [f]([x]) \subset [f]([y]) \quad (2.22)$$

2.4.1 Fonction d'inclusion naturelle

Considérons $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, une fonction de plusieurs variables construite à partir de compositions d'opérateurs arithmétiques ($+$, $-$, $.$, $/$) et de fonctions élémentaires usuelles (\tan , \log , \exp , $\sqrt{(\)}$). Une fonction d'inclusion naturelle est obtenue en remplaçant chaque variable réelle x_i par son intervalle $[x_i]$ ($i = 1, 2, \dots, n$). Cependant, la fonction d'inclusion naturelle n'est pas minimale à cause du problème de pessimisme, notion qu'on définira par la suite. Une fonction d'inclusion continue est minimale si chaque variable n'apparaît qu'une seule fois comme le montre l'exemple académique suivant [81]

Exemple 2-2

Considérons la même fonction f écrite de 4 façons différentes :

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

Si on évalue les fonctions d'inclusions naturelles de ces 4 fonctions pour un intervalle $[x] = [-1, 1]$. On obtient les résultats suivants :

$$\begin{aligned} [f]_1([x]) &= [x]^2 + 2[x] = [-2, 3] \\ [f]_2([x]) &= [x]([x] + 2) = [-3, 3] \\ [f]_3([x]) &= [x].[x] + 2.[x] = [-3, 3] \\ [f]_4([x]) &= ([x] + 1)^2 - 1 = [-1, 3] \end{aligned}$$

On constate que la taille des intervalles obtenus diffère en fonction de l'écriture de la fonction. La fonction f_4 est minimale car le paramètre x n'apparaît qu'une

seule fois, elle nous fournit donc le plus petit intervalle contenant l'image de $[x]$ par f .

Dans la figure 2.4 suivante, l'image d'un pavé $[x]$ par une fonction f est un ensemble quelconque, la fonction d'inclusion $[f]([x])$ est un pavé qui encadre cet ensemble mais qui n'est pas minimal, c'est ce qui traduit cet effet d'enveloppement. Par contre, La fonction représentée par $[f]^*([x])$ est minimale.

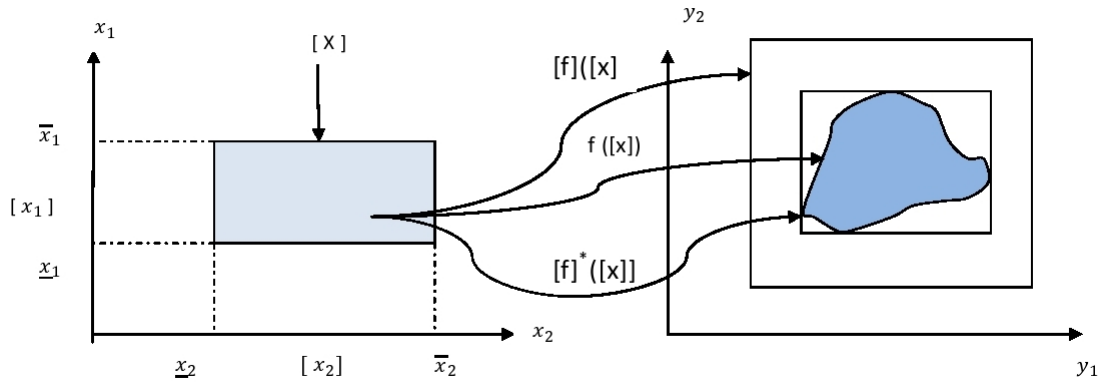


FIGURE 2.4 – Différentes images d'un pavé $[x]$ par une fonction f , sa fonction d'inclusion $[f]$ et sa fonction d'inclusion minimale $[f]^*$

2.4.2 Fonction élémentaire

Etant donnés \mathbb{X} et \mathbb{Y} , deux sous ensembles de \mathbb{R} . Une fonction f définie de \mathbb{X} dans \mathbb{Y} est appelé fonction élémentaire, s'il est possible de disposer d'une fonction d'inclusion minimale $[f]^* : \mathbb{I}\mathbb{R} \rightarrow \mathbb{I}\mathbb{R}$ tel que :

$$[f]^*([x]) = [\{f(x) | x \in [x]\}] \quad (2.23)$$

Remarque

Toute fonction élémentaire f , continue et monotone, possède une fonction d'inclusion minimale :

$$[f]^*([x]) = [\min(f(\underline{x}), f(\bar{x})), \max(f(\underline{x}), f(\bar{x}))] \quad (2.24)$$

Exemple 2-3

La fonction exponentielle étant continue et croissante sur \mathbb{R} , on a donc :

$$[exp(x)] = [exp(\underline{x}), exp(\bar{x})]$$

2.4.3 Fonction d'inclusion centrée

Afin de diminuer le problème de multioccurrences, c'est à dire le fait qu'une variable apparaît plusieurs fois dans une même fonction, on peut utiliser la forme centrée[39]. En utilisant le théorème de la valeur moyenne [74], on considère une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ différentiable sur un domaine D contenant un pavé $[x]$

On définit par \hat{x} le milieu du pavé $[x]$. On a

$$\forall x \in [x], \exists \xi \in [x] | f(x) = f(\hat{x}) + J(\xi)(x - \hat{x}) \quad (2.25)$$

où J représente le Jacobien de la fonction f . Soit $[J]$, une fonction d'inclusion de J , alors on a :

$$\forall x \in [x], f(x) \in f(\hat{x}) + [J]([x])(x - \hat{x}) \quad (2.26)$$

f sera notée f_c forme centrée. On a par la suite :

$$f([x]) \subseteq f_c([x]) = f(\hat{x}) + [J]([x])([x] - \hat{x}) \quad (2.27)$$

la fonction d'inclusion centrée n'est efficace en général que dans le cas d'intervalles de petite taille.

2.4.4 Fonction d'inclusion de Taylor

On peut réduire le pessimisme de manière significative, toujours pour des petits pavés, quand on utilise la forme de Taylor définie par :

$$[f]_T([x]) = f(\hat{x}) + J([x])([x] - \hat{x}) + \frac{1}{2}([x] - \hat{x})^T [H]([x])([x] - \hat{x}) \quad (2.28)$$

Où $[H]$ représente une fonction d'inclusion du Hessian de la fonction f .

2.4.5 Définition d'un sous-pavage

Un sous-pavage $\mathbb{X} \subset \mathbb{R}^n$ est une unions de pavés $([x])_{i=1,\dots,n}$ dont les intérieurs sont disjoints. Lorsque un sous-pavage $\hat{\mathcal{P}}$ recouvre tout le pavé $[x]$, il est appelé pavage de $[x]$. Pour déterminer les éléments d'un sous pavage, on doit vérifier s'ils possèdent des éléments vérifiant une même propriété donnée. Pour cela, on utilise un test appelé test d'inclusion.

Test d'inclusion

On définit l'ensemble des intervalles booléens $\mathbb{IB} \subset \mathbb{IR}$, tel que :

$$\mathbb{IB} = \{\emptyset, [0], [1], [0, 1]\} \quad (2.29)$$

où l'élément $[1]$ correspond à vrai, $[0]$ à faux, $[0, 1]$ à indéterminé et \emptyset pour impossible.

On définit une fonction test $t : \mathbb{R}^n \rightarrow \mathbb{B}$. Avec $\mathbb{B} = \{vrai, faux\}$

Comme pour les réels, la notion de fonction d'inclusion est étendue aux booléens.

La fonction $[t] : \mathbb{IR}^n \rightarrow \mathbb{IB}$ est appelée test d'inclusion pour t si $\forall [x] \in \mathbb{IR}^n$:

$$\begin{aligned} [t]([x]) = 1 &\Rightarrow \forall x \in [x], t(x) = 1 \\ [t]([x]) = 0 &\Rightarrow \forall x \in [x], t(x) = 0 \end{aligned} \quad (2.30)$$

Exemple 2.2

Soit le test $t : \mathbb{R}^2 \rightarrow \{0, 1\}$

$$(x_1, x_2)^T \rightarrow (x_1 + x_2 \leq 10)$$

ce qui veut dire :

$$t(x) = \begin{cases} 1 & \text{si } (x_1 + x_2) \leq 10 \\ 0 & \text{si } (x_1 + x_2) > 10 \end{cases}$$

La fonction d'inclusion $[t]$ est définie par :

$$[t](x) = \begin{cases} 1 & \text{si } (\bar{x}_1 + \bar{x}_2) \leq 10 \\ 0 & \text{si } (\underline{x}_1 + \underline{x}_2) > 10 \\ [0,1] & \text{ailleurs} \end{cases}$$

qu'on peut réécrire aussi :

$$[t](x) \Leftrightarrow ([x_1] + [x_2]) \leq 10$$

2.4.6 Réduction d'un pavé

Afin d'évaluer une fonction d'inclusion sur un sous-pavage, tout en essayant de réduire le phénomène d'enveloppement, on est amené à réduire un pavé par la technique de bisection.

La bisection

soit un pavé $[x] \subset \mathbb{R}^n$ tel que :

$$[x] = [x_1] \times [x_2], \dots, [x_n] \quad (2.31)$$

et soit l'index j sa première composante de largeur maximale, c'est à dire.

$$j = \min \{i \mid \omega([x_i]) = \omega([x])\} \quad (2.32)$$

On appelle bisection, une opération qui consiste à obtenir des pavés notés $L[x]$ et $R[x]$ à partir de $[x]$, tel que :

$$L[x] = [x_1] \times \dots \times [\underline{x}_i, x_{0i}] \times \dots \times [x_n] \quad (2.33)$$

$$R[x] = [x_1] \times \dots \times [x_{0i}, \bar{x}_i] \times \dots \times [x_n]$$

$$\text{avec : } x_{0i} = \frac{x_i + \bar{x}_i}{2}$$

i est l'indice correspondant au pavé de plus grande taille. Les pavés $L[x]$ et $R[x]$ sont dits frères et fils du pavé père $[x] = L[x] \cup R[x]$

Exemple 2.3

Soit $x = [1, 2] \times [2, 4] \times [1, 3]$, On a $\omega([x]) = 2$ donc $j = 2$, par conséquent :

$$L(x) = [1, 2] \times [2, 3] \times [1, 3] \quad \text{et} \quad R(x) = [1, 2] \times [3, 4] \times [1, 3]$$

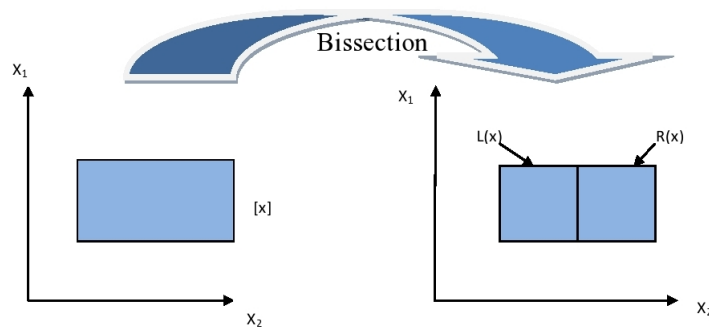


FIGURE 2.5 – Bissection d'un pavé

2.5 Pessimisme

Lorsque le résultat des opérations entre intervalles n'est pas minimal, on parle alors de pessimisme. Le pessimisme est une conséquence du phénomène d'enveloppement (wrapping effect) et du phénomène de dépendance (ou multioccurrence).

2.5.1 Phénomène d'enveloppement

Ce phénomène se produit quand on veut représenter un ensemble quelconque par un pavé. Il provoque une surestimation du pavé résultant. L'exemple suivant permet de mieux comprendre cet effet :

Exemple 2-4

Soit le pavé $[x] = [x_1] \times [x_2]$, si l'on fait subir à ce pavé des rotations successives à l'aide de la matrice A définie par :

$$A = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$$

avec $\theta = \frac{\pi}{4}$

La représentation de ce pavé image est un rectangle qui l'enveloppe. Après plusieurs rotations successives, ce rectangle augmente, ce qui provoque une sur-estimation du résultat final.

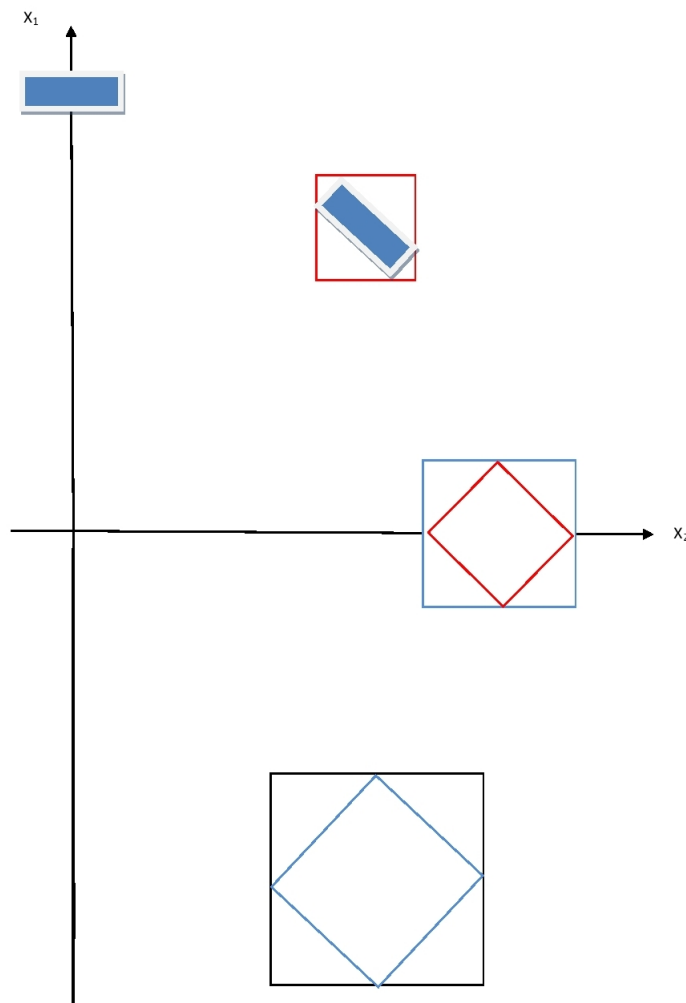


FIGURE 2.6 – Effet d'enveloppement consécutif à la rotation d'un pavé

2.5.2 Phénomène de dépendance

Encore appelé phénomène de multioccurrences, le phénomène de dépendance suggère que chaque variable est considérée différente. En d'autres termes, ce pessimisme est dû au fait que l'ensemble $X - X = \{x - y, x \in X, y \in X\}$ est différent

de l'ensemble $\{x - x, x \in X\} = 0$

Exemple 2-5

Soit $[x] = [-1, 1]$, alors $[x] - [x] = [-1, 1] - [-1, 1] = [-2, 2] \neq 0$

2.6 Image directe et inverse d'un ensemble

Soit \mathbb{X} un ensemble de \mathbb{R}^n , \mathbb{Y} un ensemble de \mathbb{R}^m et $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, une fonction vectorielle. On définit : l'image direct de \mathbb{X} par f :

$$f(\mathbb{X}) = \{y \in \mathbb{R}^m \mid \exists x \in \mathbb{X}, f(x) = y\} \quad (2.34)$$

Il est difficile d'obtenir l'image d'un pavé ou d'un sous ensemble par une méthode ponctuelle car on ne peut le faire que pour un nombre fini de points, or un ensemble contient une infinité de points. D'autre part l'image d'un pavé est un ensemble quelconque qui ne correspond pas à un pavé. Un algorithme (ImageSp) pour le calcul de l'image d'un sous-ensemble a été développé par Kieffer [54]. On peut décrire le principe, en considérant un ensemble \mathbb{X} dont on cherche à approximer l'image \mathbb{Y} par une fonction f , ce qui revient à caractériser :

$$f : \mathbb{R}^n \rightarrow \mathbb{R}^m \quad f(\mathbb{X}) = \{f(x) \mid x \in \mathbb{X}\} \quad (2.35)$$

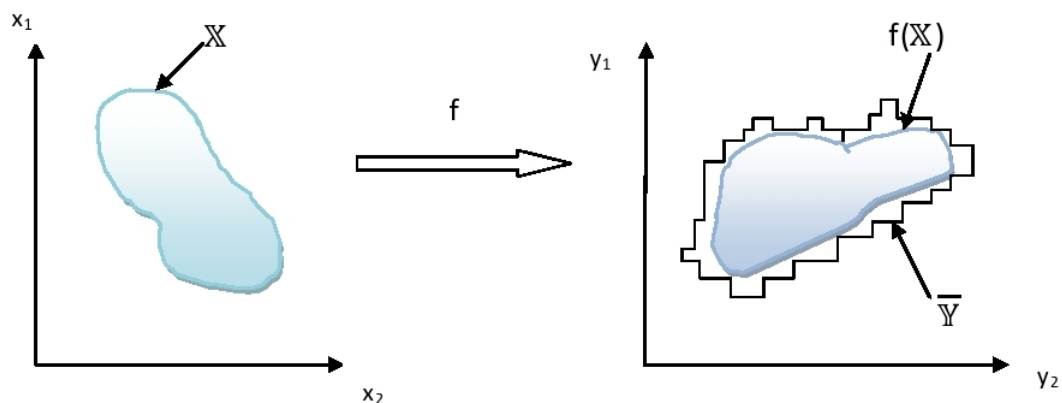


FIGURE 2.7 – Image directe d'un sous ensemble

La procédure se fait en 3 étapes, la première est le **hachage** qui consiste à partitionner \mathbb{X} en une liste L_x de pavés $[x]$. La seconde étape est **L'évaluation** dans laquelle on évalue les pavés $[x]$ issue de L_x , c'est la première approximation de $\bar{\mathbb{Y}}$, puis la dernière étape est **La régularisation** qui permet d'approximer plus finement l'ensemble $\bar{\mathbb{Y}}$ en utilisant l'algorithme Sivia qu'on décrira plus loin. La procédure est plus détaillé dans [49]

On définit aussi : L'image inverse de \mathbb{X} par f :

$$f^{-1}(\mathbb{Y}) = \{x \in \mathbb{R}^n \mid \exists y \in \mathbb{Y}, f(x) = y\} \quad (2.36)$$

C'est ce principe, connu aussi sous le nom d'inversion ensembliste qui a le plus d'application en automatique notamment pour la résolution des problèmes d'estimation.

On a les propriétés suivantes :

$$\begin{aligned} f(\mathbb{X}_1 \cap \mathbb{X}_2) &\subset f(\mathbb{X}_1) \cap f(\mathbb{X}_2) \\ f(\mathbb{X}_1 \cup \mathbb{X}_2) &= f(\mathbb{X}_1) \cup f(\mathbb{X}_2) \\ f^{-1}(\mathbb{Y}_1 \cup \mathbb{Y}_2) &= f^{-1}(\mathbb{Y}_1) \cup f^{-1}(\mathbb{Y}_2) \\ f^{-1}(\mathbb{Y}_1 \cap \mathbb{Y}_2) &= f^{-1}(\mathbb{Y}_1) \cap f^{-1}(\mathbb{Y}_2) \\ f(f^{-1}(\mathbb{Y})) &\subset \mathbb{Y} \\ f^{-1}(f(\mathbb{X})) &\supset \mathbb{X} \end{aligned} \quad (2.37)$$

2.7 Contraction

2.7.1 Problème de satisfaction de contraintes

Le problème de satisfaction de contraintes est plus connu par la communauté informaticienne, spécialisée en intelligence artificielle.

Soit une fonction $f \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$ avec \mathbb{X} un domaine de variables initiales et $x = (x_1, x_2, \dots, x_n)$ un vecteur variable. Nous considérons un système d'équations.

$$f(x) = 0 \Leftrightarrow \begin{cases} f_1(x_1, x_2, \dots, x_n) = 0 \\ \vdots \\ f_j(x_1, x_2, \dots, x_n) = 0 \\ \vdots \\ f_m(x_1, x_2, \dots, x_n) = 0 \end{cases} \quad n \leq m \quad (2.38)$$

Nous appelons problème de satisfaction de contraintes \mathcal{H} , la détermination de toutes les solutions du système d'équations à résoudre (2.38), contenues dans le domaine de recherche initial \mathbb{X} .

$$\text{On note :} \quad \mathcal{H} : (f(x) = 0, x \in \mathbb{X}) \quad (2.39)$$

Si \mathbb{S}_j est un ensemble de solutions de la contrainte $f_j(x_1, x_2, \dots, x_n) = 0$, alors la solution du problème de satisfaction de contraintes est obtenue par :

$$\mathbb{S} = \bigcap_1^n \mathbb{S}_j \quad (2.40)$$

Les techniques de contractions ont été élaboré [49; 82] afin de réduire des pavés sans avoir recours parfois aux bisections. Donc, un contracteur \mathcal{C} est un opérateur qui permet de contracter ou réduire un pavé de recherche initial sans bisection et en gardant l'intégralité des solutions, c'est à dire éliminer les valeurs inconsistantes. Un contracteur doit donc vérifier les deux propriétés suivantes :

$$\forall [x] \in \mathbb{I}\mathbb{R}^n \begin{cases} \forall [x] \subseteq \mathbb{X}, \mathcal{C}_s([x]) \subset [x] \text{ (contractance)} \\ \forall [x] \subseteq \mathbb{X}, [x] \cap \mathbb{S} \subset \mathcal{C}_s([x]) \text{ (Complétude)} \end{cases} \quad (2.41)$$

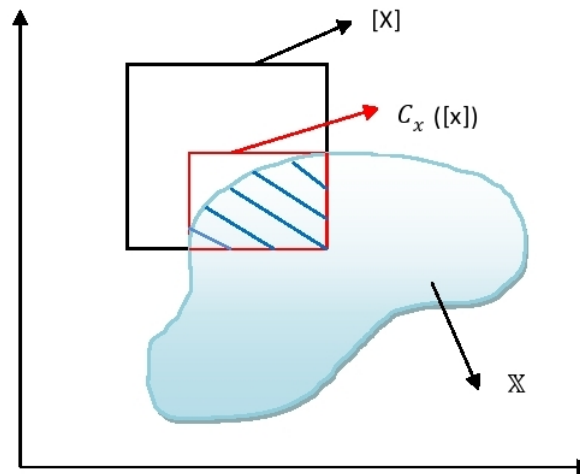


FIGURE 2.8 – Contraction d'un ensemble

Pour résoudre des problèmes de satisfaction de contraintes non linéaires, on a développé différents types de contracteur : Le contracteur de Newton [38], Le contracteur de Krawczyk [74], Le contracteur de propagation-rétropropagation. On dispose aussi d'un contracteur de Gauss-Seidel valable pour les cas linéaires. Dans le cadre de ce travail, on a utilisé un contracteur par projection.

2.7.2 Notion de consistence

Dans un premier temps, nous considérons une seule contrainte \mathcal{H}_j . Soit un pavé $[x] \subset \mathbb{X}$. La valeur $x_i \in [x_i]$ est dite consistante avec \mathcal{H}_j , s'il est possible d'instancier les autres composantes x_j de x , de manière que la relation $f_j(x) = 0$ soit satisfaite. En d'autres termes :

$$x_i \in \pi_i(\mathbb{S}_j \cap [x]) \quad (2.42)$$

π_i représente la projection de $\mathbb{S}_j \cap [x]$ sur $[x_i]$

Exemple 2-7

Soient les variables x , y et z suivantes telle que : $x \in [1, 5]$, $y \in [2, 4]$ et $z \in [6, 10]$ et soit la contrainte $z = x + y = [3, 9]$.

On constate que les valeurs 1 pour x et 10 pour z sont dites inconsistantes .

2.7.3 Projection de contraintes

Considérons un ensemble \mathbb{S} , nous notons $C_s([x])$ le plus petit pavé contenant $[x]$. l'opérateur de contraction C_s est en mesure de remplacer une boîte $[x]$ par un pavé $\mathbb{S} \cap [x]$ de plus petite taille tout en conservant l'ensemble solution au complet. Ce procédé que nous allons décrire dans le paragraphe suivant est basé sur le principe de projection de contraintes utilisant des sous solveurs [17].

2.7.4 Sous-résolveurs

Un sous solveur est un algorithme qui permet de calculer une composante x_i du vecteur x en fonction des autres composantes. Ce qui va nous permettre par la suite d'appliquer des contractions par projection inspiré du principe branch and bound et le calcul par intervalle

Soit $\vec{p} = (p_1, p_2, \dots, p_n)^T$ un vecteur de \mathbb{R}^n .

Un sous ensemble \mathcal{P}_1 de \mathbb{R}^n est une contrainte [17] telle que :

$$\mathcal{P}_1 = \{ \vec{p} \in \mathbb{R}^n | f(\vec{p}) \in [y] \} = f^{-1}([y]) \quad (2.43)$$

où $[y]$ est un intervalle et f une fonction continue. Considérons un pavé $[\vec{p}]$, Le problème de réduction consiste à obtenir un pavé $[x]$ le plus petit possible sans avoir recours à des bisections et qui contient l'ensemble :

$$\mathcal{S}_1 = f^{-1}([y]) \cap [\vec{p}] \quad (2.44)$$

On suppose qu'il est possible d'isoler p_i de l'expression $y = f(\vec{p})$, c.à.d. il existe une fonction g telle que :

$$f(\vec{p}) = y \Leftrightarrow p_i = g_i({}^i\vec{p}, y) \quad (2.45)$$

où ${}^i\vec{p} = (p_1, \dots, p_{i-1}, p_{i+1}, \dots, p_n)^T$ et g_i la fonction solution appelée aussi sous solveur, associée à p_i , soit π_i l'opérateur de projection sur le i_{eme} axe et $[g_i]$ la fonction d'inclusion associée à g_i

Theorem 1. *Supposons qu'il existe un sous résolveurs g_i d'entrée ${}^i p$ et de sortie p associé au problème de satisfaction de contraintes (2.39) On peut écrire :*

$$\pi_i (S_1) \subset [g_i] ({}^i[\vec{p}], [y]) \cap [p_i] \quad (2.46)$$

Si g_i est continu et $[g_i]$ est minimale, alors la relation (2.46) devient égalité.

Supposons maintenant que pour $j \in \{1, \dots, m\}$ et $i \in \{1, \dots, n\}$, il est possible d'isoler la variable p_i dans l'expression $f_j(\vec{p}) = y_i$ c.à.d. qu'il existe une fonction solution g_j^i qui satisfait :

$$f_j(\vec{p}) = y_j \Leftrightarrow p_i = g_j^i({}^i\vec{p}, y) \quad (2.47)$$

En tenant compte du théorème 2.1, On a :

$$\pi_i (S_j) \subset [g_j^i] [{}^i[\vec{p}], [y_j]] \cap [p_i] \quad (2.48)$$

Par la suite,

$\forall j \in \{1, \dots, m\}$ et $i \in \{1, \dots, n\}$, $\pi_i (S_j)$ est inclus dans l'intervalle

$$[h_j^i] ([{}^i\vec{p}], [y_j]) \begin{cases} [g_j^i] ({}^i[\vec{p}], [y_i]) \cap [p_i] \text{ si } g_j^i \text{ existe} \\ [p_i] \text{ autrement} \end{cases} \quad (2.49)$$

Exemple 2.8

Soit la contrainte $p_1 p_2 \in [8, 40]$ et un pavé $[p] = [1, 4] \times [1, 4]$ On peut obtenir un encadrement de S_1 en réduisant $[p]$ comme suit [44].

$p_1 p_2 = y \Leftrightarrow p_1 = \frac{y}{p_2} \Leftrightarrow p_2 = \frac{y}{p_1}$ pour $p_i \neq 0$ Les fonctions solutions sont :
 $g_1(p_2, y) = \frac{y}{p_2}$ et $g_2(p_1, y) = \frac{y}{p_1}$

En tenant compte du théorème 2.1, la projection de S_1 est :

$$\begin{aligned}
\pi_1(\mathcal{S}_1) &= [g_1]([p_2], y) \cap [p_1] = \frac{[y]}{[p_2]} \cap [p_1] \\
&= \frac{[8,40]}{[1,4]} \cap [1,4] = [2,4] \\
\pi_2(\mathcal{S}_1) &= [g_2]([p_1], y) \cap [p_2] = \frac{[y]}{[p_1]} \cap [p_2] \\
&= [2,4]
\end{aligned}$$

par conséquent l'intervalle enveloppe de \mathcal{S}_1 est

$$[\mathcal{S}_1] = [2,4] \times [2,4]$$

2.8 Inversion ensembliste

Considérons un ensemble dans \mathbb{R}^n et une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$. Soit \mathbb{Y} un sous ensemble de \mathbb{R}^m défini comme suit :

$$\mathbb{X} = f(\mathbb{Y}) \tag{2.50}$$

L'inversion d'ensembles nous permet de déterminer l'ensemble \mathbb{X} , comme image réciproque par la fonction f de l'ensemble \mathbb{Y} :

$$\mathbb{S} = \{x \in \mathbb{R}^n | f(x) \in \mathbb{Y}\} = f^{-1}(\mathbb{Y}) \tag{2.51}$$

On peut résoudre un problème d'inversion ensembliste par le biais de l'algorithme SIVIA (Set Inversion Via Interval Analysis)[49]. SIVIA est un algorithme récursif permettant d'obtenir, à partir d'un pavé initial suffisamment large, deux sous ensembles encadrant l'ensemble solution, tel que :

$$\underline{\mathbb{S}} \leq \mathbb{S} \leq \bar{\mathbb{S}} \tag{2.52}$$

L'approximation, ou encadrement, intérieure $\underline{\mathbb{S}}$ représente l'ensemble des pavés dits acceptables ou faisables. C'est l'ensemble contenant toutes les solutions recher-

chées. Cependant il peut y avoir des solutions admissibles qui ne sont pas contenues dans $\underline{\mathbb{S}}$. Si on démontre que $[x] \cap \overline{\mathbb{S}} = \emptyset$ alors le pavé $[x]$ est inacceptable et sera supprimé. Dans le cas où le pavé est ni acceptable, ni inacceptable, on dit alors que le pavé $[x]$ est indéterminé. L'algorithme va donc découper ce pavé le long du plus grand côté en deux sous-pavés qui vont être testés à leur tour afin de vérifier s'ils peuvent être gardés ou rejetés. Le processus est itéré à nouveau jusqu'à l'obtention d'un pavé dont la largeur atteint un certain seuil ϵ , fixé par l'expérimentateur. Dans tous les autres cas, le pavé sera dit indéterminé.

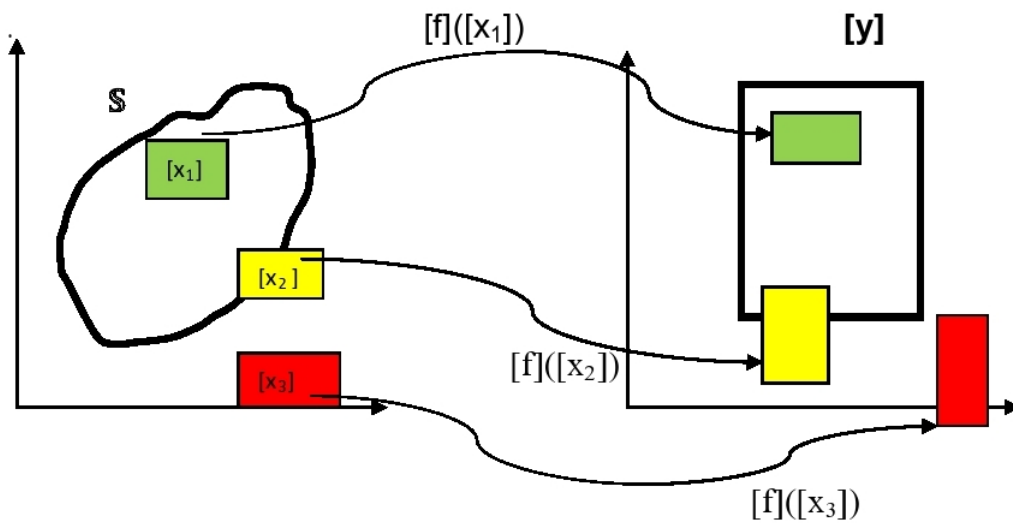


FIGURE 2.9 – Sivia

La description de l'algorithme Sivia est donc :

2.8.1 L'algorithme

SIVIA (*entrée* : $[t], [x], \epsilon$; *sorties* : $\underline{\mathbb{S}}, \overline{\mathbb{S}}$),

1. si $[t]([x]) = [0]$, rejeter $[x]$;
2. si $[t]([x]) = [1]$, $\underline{\mathbb{S}} := \underline{\mathbb{S}} \cup [x]$, $\overline{\mathbb{S}} := \overline{\mathbb{S}} \cup [x]$
3. si $w([x]) \leq \epsilon$, $\overline{\mathbb{S}} := \overline{\mathbb{S}} \cup [x]$
4. bissecter $[x]$ en $([x_1], [x_2])$
 $SIVIA(e : [t], x_1, \epsilon; s : \underline{\mathbb{S}}, \overline{\mathbb{S}})$;
 $SIVIA(e : [t], x_2, \epsilon; s : \underline{\mathbb{S}}, \overline{\mathbb{S}})$

Résoudre un problème d'inversion ensembliste en ayant recours au principe des bisections est un problème NP-dur dont la résolution se limite au cas où le nombre de variables est réduit. C'est pour cette raison qu'on a développé des techniques de contractions [48; 25] permettant de réduire les domaines solutions. Si l'on intègre un contracteur par projection à l'algorithme Sivia afin de diminuer le nombre de bisection, on obtient l'algorithme SiviaP

2.8.2 L'algorithme SiviaP

SiviaP (*entrées* : $C, [p], [y], \epsilon$; *sorties* : $\underline{\mathbb{S}}, \overline{\mathbb{S}}$)

1. $[x] = C([p])$
2. si $[p] = \emptyset$, *rejeter* $[p]$
3. si $[f]([p]) \subseteq [y]$ alors
 $\underline{\mathbb{S}} := \underline{\mathbb{S}} \cup [x], \overline{\mathbb{S}} := \overline{\mathbb{S}} \cup [x]$
4. si $w([p]) < \epsilon$, alors $\overline{\mathbb{S}} := \overline{\mathbb{S}} \cup [x]$
5. bissecter $[p]$ en $[p1]$ et $[p2]$
6. SiviaP ($[p1]$), SiviaP ($[p2]$)

2.9 Conclusion

Dans ce chapitre, nous avons présenté les différentes définitions de l'arithmétique des intervalles telles qu'elles furent posées initialement par R.Moore. Puis nous avons vu comment l'on peut manipuler les ensembles à l'aide de cette arithmétique, en les enveloppant dans des pavés, puis comment les réduire par bisection. On a vu aussi comment déterminer l'image directe et inverse d'un ensemble par l'application de la fonction d'inclusion. On a défini ensuite une notion importante qui est l'inversion ensembliste, qui associée à un contracteur va nous servir pour notre application dans les chapitres suivants.

Chapitre 3

Identification dans le contexte à erreurs bornées

3.1 Introduction

Le principe de l'identification est de développer un modèle mathématique issu d'une classe de modèles bien connue. Puis on soumet ces modèles à des signaux d'entrées tests, afin de produire une réponse qui soit la plus proche possible que celle obtenue par le système réel.[94]. L'identification du système se concentre donc sur la modélisation des systèmes dynamiques à partir de données expérimentales. Les applications concernent de nombreux domaines des sciences et de l'ingénierie, même si c'est une tâche assez coûteuse, l'identification est nécessaire pour construire des systèmes de contrôle en boucle fermée.

3.2 Méthodes classiques d'identification

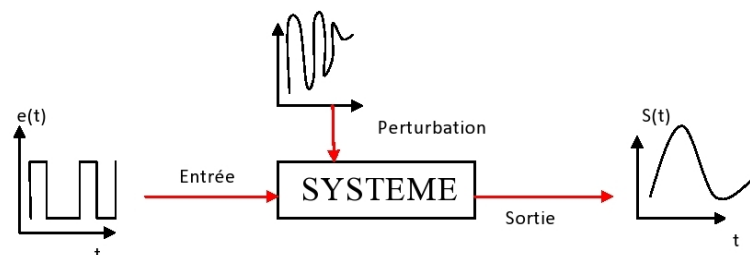


FIGURE 3.1 – Étude de système dynamique

Selon les informations disponibles concernant le système dynamique étudié, deux structures sont proposées :

Le modèle comportementale : modèle purement mathématique pouvant décrire un phénomène trop complexe. Le vecteur paramètre à identifier n'a pas vraiment de signification physique.

Le modèle de connaissance : C'est un modèle obtenu à partir des lois physiques, dans ce cas les paramètres ont un sens physique

Lors de la modélisation, on peut se trouver dans un cas où l'on cherche à évaluer des paramètres pour lesquels on ne dispose pas d'appareils de mesure, on fournit donc une estimée \hat{x} de x , c'est une mesure indirecte de x , on parle aussi de capteur logiciel (ou observateur), très utilisé pour l'étude de systèmes chimiques ou biologiques .

Le choix d'une structure est une tâche difficile, pour trancher entre deux structures, il est conseillé de procéder à une analyse de discernabilité ainsi qu'à l'étude d'identifiabilité. Ce qui va permettre à l'expérimentateur de choisir les bons capteurs déterminés par un cahier de charges, pour recueillir les données expérimentales. C'est l'étape a priori où l'on planifie un protocole expérimental. Dans l'étape a posteriori, les données recueillies subissent un pré-traitement, filtrage débruitage, puis le choix d'un bon estimateur va permettre d'ajuster la valeur du paramètre x de manière à minimiser une distance entre le système étudié et le modèle mathématique choisi, au sens d'un critère à définir appelé fonction de coût.

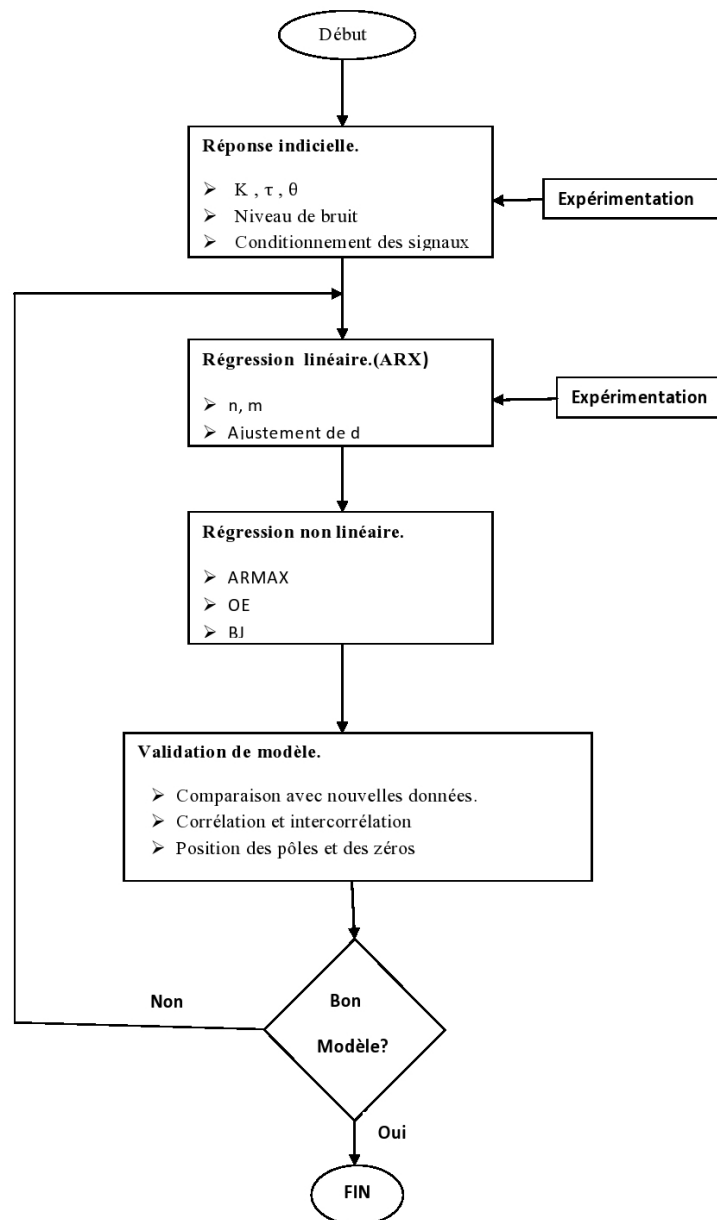


FIGURE 3.2 – Procédure d'identification

Une procédure d'identification va nous permettre de faire un choix parmi les différentes structure de systèmes et leurs variables de conception en termes de leur influence sur le biais et la variance. Elle va permettre aussi une utilisation efficace de la connaissance a priori concernant le système à identifier et les exigences de performance de l'application envisagée (par exemple, la simulation, la prédiction, le contrôle en boucle fermée. Les exigences des procédures d'identification concernent :

1. les problèmes de niveau de compétence : l'existence de nombreuses méthodes

d'identification du système suppose que l'utilisateur possède une vaste expérience dans les statistiques, traitement du signal, systèmes à temps discret, et l'optimisation.

2. Le Choix du signal d'entrée : binaire aléatoire, binaires pseudo-aléatoires (PRBS), choix du temps , amplitude, (gain K , constante de temps τ , retard θ) ...
3. Phase de prétraitement des données : préfiltrage , l'élimination des valeurs aberrantes, etc.
4. La sélection d'un modèle de structure pour l'estimation des paramètres : ARX, ARMAX, Erreur de sortie, Box-Jenkins. Fixer l'ordre du modèle.
5. La validation du modèle : simulation, validation croisée, analyse de corrélation, représentation pôle-zéro, et les réponses impulsionnelles.

L'estimation des paramètres est obligatoirement associée à une étude statistique afin d'évaluer la qualité des résultats

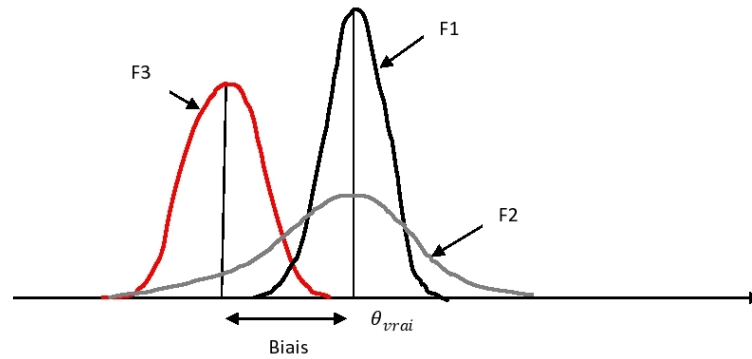
- Variance : Les erreurs aléatoires introduites par la présence de bruit dans les données ne permettent pas au modèle de reproduire exactement la sortie. La variance mesure la dispersion des données autour de la moyenne ou son espérance.

Elle est définie par :

$$Var(X) = E[(X - E[X])^2] = \frac{1}{n} \sum_{i=1}^n (x_i - m)^2 = \sigma^2 \quad (3.1)$$

E représente l'espérance mathématique, c'est la moyenne des valeurs attendues, après répétition de n expériences. σ est appelé écart type.

- Biais : c'est un écart entre la vraie valeur et la valeur estimée. Les erreurs systématiques sont causées par une entrée caractéristique du signal par exemple, le degré d'excitation, le choix de la structure du modèle, le mode de fonctionnement (boucle fermée ou boucle ouverte).

FIGURE 3.3 – Trois estimateurs pour un paramètre θ

Sur la figure (3.3), sont représentés trois estimateurs F1, F2 et F3 pour la variable aléatoire θ . F1 et F2 sont deux estimateurs sans biais avec :

$Var(F1) < Var(F2)$. L'estimateur F3 est biaisé.

Il est donc indispensable de faire appel, pour identifier les paramètres, à des techniques statistiques dont l'efficacité dépend en partie de la modélisation du problème, de la procédure d'acquisition de données ainsi que de l'algorithme d'estimation choisi.

3.2.1 Statistiques en identification

L'estimation d'une grandeur incertaine passe obligatoirement par un calcul statistique, c'est à dire déterminer la loi de probabilité qui décrit la variable estimée. Une estimation peut être ponctuelle : c'est à dire, on calcule une estimé \hat{x} de x ou bien elle peut être intervalle : on cherche à déterminer un intervalle dans lequel l'estimé \hat{x} se trouve avec une grande probabilité. Dans [45] Jaulin procède à une étude comparative entre les deux approches statistiques : l'approche probabiliste et l'approche ensembliste. En effet, dans l'approche probabiliste une variable réelle ou vectorielle est décrite par sa densité de probabilité qu'on appelle loi, tel que :

$$\text{soit} \quad \pi : \mathbb{R}^n \rightarrow \mathbb{R}^+ \quad \int_{\mathbb{R}^n} \pi(x) dx = 1 \quad (3.2)$$

ce calcul nous fournit un ensemble ou support :

$$\mathbb{X} = \{x \in \mathbb{R}^n \mid \pi(x) \neq 0\} \quad (3.3)$$

qui contient avec certitude la variable incertaine et il nous renseigne aussi sur la

manière dont elle est distribuée sur ce support.

Dans l'approche ensembliste, la variable incertaine est représentée juste par un ensemble \mathbb{X} censé la contenir et qu'on appelle *ensemble de vraisemblance ou domaine*. Par conséquent, cette approche permet de manipuler les variables plus facilement sans pour cela nécessiter des connaissances statistiques sur ces variables, ce qui va permettre donc, de résoudre beaucoup plus de problèmes.

Exemple 3-1

Considérons trois variables aléatoires x , y , et z reliées par la contrainte $z = x + y$. Il s'agit de déterminer les propriétés statistiques de ces variables par les deux approches, l'approche probabiliste puis ensembliste.[45]

Approche probabiliste :

On suppose que x et y ont des densités de probabilité uniforme $\pi(x) = \frac{1}{5}$ sur un intervalle $[0, 5]$ et $\pi(y) = \frac{1}{2}$ sur un intervalle $[1, 3]$. On suppose aussi que les variables x et y sont indépendantes. Ce que nous permet de considérer une densité uniforme du couple $\pi(x, y)$ tel que : $\pi(x, y) = \frac{1}{10}$ sur le pavé $[0, 5] \times [1, 3]$. Le calcul de la densité de probabilité $\pi(z)$ se fait à partir du calcul intégrale suivant :

$$\pi_Z(z) = \int_{-\infty}^{+\infty} \pi_x(t)\pi_y(z-t)dt \quad (3.4)$$

Les variables aléatoires x et y étant indépendantes, leurs densités respectives sont :

$$\pi_x(t) = \begin{cases} \frac{1}{5} & \text{si } t \in [0, 5] \\ 0 & \text{ailleurs} \end{cases}$$

et

$$\pi_y(t) = \begin{cases} \frac{1}{2} & \text{si } t \in [1, 3] \\ 0 & \text{ailleurs} \end{cases}$$

On a aussi :

$$\pi_y(z-t) = \begin{cases} \frac{1}{2} \text{ si } z-t \in [1, 3] \text{ ie : } z-3 \leq t \leq z-1 \\ 0 \text{ ailleurs} \end{cases}$$

Pour chaque intervalle, le calcul intégrale nous donne le résultat suivant :

$$\begin{aligned} \pi(z) &= 0 \text{ si } z \leq 1 \\ \pi(z) &= \frac{1}{10}(z-1) \text{ si } z \in [1, 3] \\ \pi(z) &= \frac{1}{5} \text{ si } z \in [3, 6] \\ \pi(z) &= \frac{1}{10}(-z+8) \text{ si } z \in [6, 8] \\ \pi(z) &= 0 \text{ si } z \geq 8 \end{aligned}$$

Approche ensembliste :

Dans ce cas, c'est un ensemble qui décrit l'incertitude associée à x et à y .

Si nous considérons dans ce cas aussi $x = [0, 5]$ et $y = [1, 3]$ et en raisonnement ensembliste, il est logique que $z = [0 + 1, 5 + 3] = [1, 8]$. On remarque que le calcul par l'approche ensembliste est plus simple et nécessite aucune hypothèse concernant la loi de probabilité $\pi(x)$ relative à la variable aléatoire x . Tout ce que l'on sait, c'est qu'un ensemble \mathbb{X} contient la variable x ainsi que le support de sa loi de probabilité, qu'elle soit uniforme ou non.

3.2.2 Les moindres carrés

Dans tous les domaines de l'ingénierie, lorsqu'on dispose de séries de mesures expérimentales de taille n , $\{x_1, x_2, \dots, x_n\}$ et $\{y_1, y_2, \dots, y_n\}$ pouvant être reliées entre eux par une relation affine du type $y = ax + b$, on parle de régression linéaire. Tenant compte des erreurs de mesures, on considère les mesures x_i et y_i qu'on peut représenter par un nuage de points, comme des réalisations de variables aléatoires X et Y .

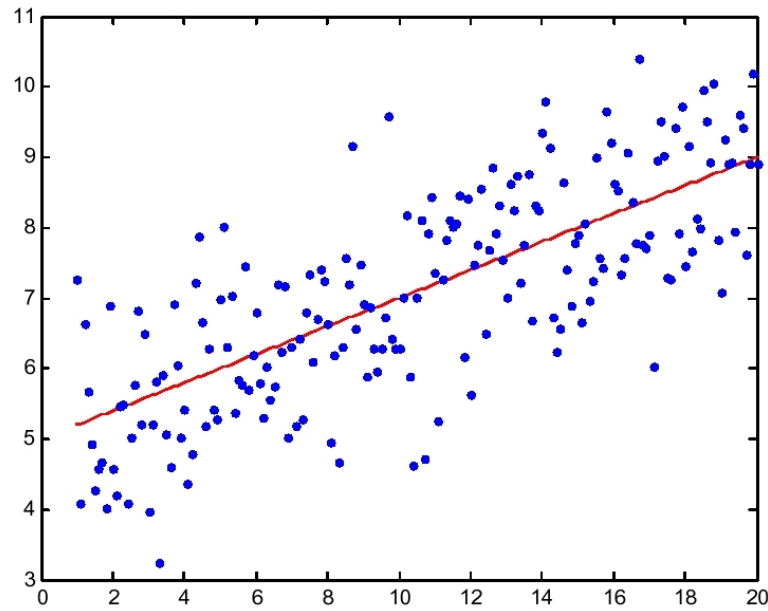


FIGURE 3.4 – Nuage de points et droite de régression

Lorsqu'on recherche une relation affine entre X et Y , on cherche à ajuster la meilleure droite appelée régression linéaire, au nuage de point. La meilleure de ces droites c'est celle qui minimise la somme des carrés des écarts des valeurs mesurées à la droite estimée $\hat{y}_i = a\hat{x}_i + b$. La méthode des moindres carrés consiste à déterminer a et b qui minimisent le carré de la norme de l'erreur ϵ

$$\sum_{i=0}^n (\|\epsilon\|^2) = \sum_{i=0}^n (y_i - (a\hat{x}_i + b))^2 = \epsilon^T \epsilon \quad (3.5)$$

qui peut se réécrire sous forme matricielle en considérant aussi un bruit additif ϵ :

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_{n-1} & 1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix} \quad (3.6)$$

La fonction de coût à minimiser est :

$$J(a, b) = \sum_{i=0}^n \epsilon_i^2 = \sum_{i=0}^n (y_i - (a\hat{x}_i + b))^2 \quad (3.7)$$

La solution optimale obtenue est :

$$\begin{pmatrix} a \\ b \end{pmatrix} = (W^T W)^{-1} W y^T \quad (3.8)$$

avec :

$$w = \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_{n-1} & 1 \end{pmatrix} \quad (3.9)$$

Le théorème suivant permet d'obtenir des estimations \hat{a} et \hat{b}

Theorem 2. *Si la variance $\text{Var}(X)$ de la série statistique $X = (x_i)$ est non nulle, il existe une droite unique qui minimise le critère $J(a,b)$. Elle vérifie :*

$$\hat{a} = \frac{\text{cov}_{xy}}{\sigma_x^2} \quad \text{et} \quad \hat{b} = \bar{y} - \hat{a}\bar{x} \quad (3.10)$$

avec :

$$\sigma_x^2 = \frac{1}{n} \sum_{i=0}^n (x_i - \bar{x})^2 \quad \text{et} \quad \text{cov}_{xy} = \frac{1}{n} \sum_{i=0}^n (x_i - \bar{x})^2 (y_i - \bar{y})^2 \quad (3.11)$$

Afin d'évaluer la qualité de l'approximation d'un nuage de point par une droite déterminée par moindres carrés, on calcul le coefficient de corrélation linéaire défini par :

$$r_{xy} = \frac{\text{cov}_{xy}}{\sigma_x \sigma_y} \quad (3.12)$$

Sachant que si : $|r_{xy}|$ est proche de 1, l'approximation par moindres carrés est de bonne qualité.

Dans le cas où le système est non linéaire, on utilise des itérations successives en se basant sur une version linéarisée du modèle initial. Le principe est de calculer la matrice Jacobienne tenant compte d'un point de départ et d'un critère d'arrêt. Il existe d'autres variantes des moindres carrés, comme les moindres carrés généralisés ou pondérés.

Dans le cadre de notre thèse, nous avons utilisé la fonction 'Lsqnonlin' de Matlab qui représente des moindres carrés non linéaires.

3.3 Identification dans le contexte à erreurs bornées

Résoudre un problème d'inversion ensembliste en utilisant le principe de bisections est un problème NP-dur dont la résolution est limitée aux cas où le nombre de variables est réduit.

Les paramètres estimés par des méthodes classiques sont des valeurs optimales obtenues par optimisation d'un critère comme celui des moindres carrés décrit au chapitre précédent mais leur qualité et fiabilité dépend de l'espérance mathématique et la matrice de covariance de l'estimateur choisi, cela suppose donc qu'on travaille avec des variables aléatoires caractérisées par une densité de probabilité et une covariance. En pratique, ces données sur les erreurs introduites ne sont pas toujours disponibles ce qui conduit à une estimation parfois biaisée voir erronée. Dans le contexte ensembliste, on suppose connaître les bornes sur ces erreurs et donc on n'a besoin d'aucune hypothèse concernant leur distribution statistique. Le résultat de l'identification se présentera sous la forme d'un ensemble de tous les paramètres compatibles avec les bornes des erreurs introduites. On parle d'identification ensembliste dans le contexte à erreurs bornées

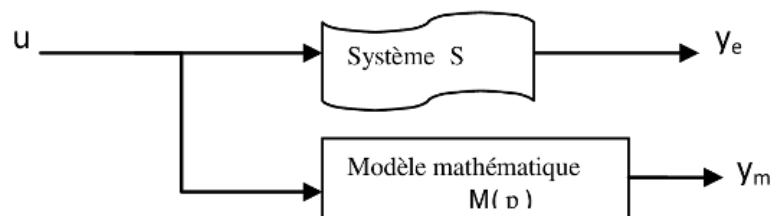


FIGURE 3.5 – Modèle d'identification

Dans ce chapitre, on s'intéresse à l'identification d'un vecteur p d'un modèle mathématique $M(p)$ de sortie $y_m(t_i)$, à partir de données expérimentales $y_e(t_i)$ disponibles à la sortie d'un système S à tout instant t_i .

D'autre part on définit un vecteur $[y_e(t_i)]$ tel que :

$$y_e(t_i) \in [y_e(t_i)] \quad (3.13)$$

Identifier les paramètres du modèle $M(p)$ revient à rechercher l'ensemble des valeurs de p tel que :

$$y_m(t_i) \in [y_e(t_i)] \quad \text{pour } i = 1, \dots, n \quad (3.14)$$

On définit l'erreur de sortie comme l'écart entre les deux vecteurs de sorties figure 3.5 :

$$e(p) = y_e(t_i) - y_m(t_i) \quad (3.15)$$

Si le vecteur $e(p)$ appartient à un vecteur intervalle $[e] = [-\delta_e, \delta_e]$ alors le vecteur $y_m(t_i)$ de $M(p)$ doit appartenir à l'intervalle $[y_m(t_i)] = y_m(t_i) + [-\delta_e, \delta_e]$. Une estimée \hat{p} du vecteur paramètre p est dite acceptable si et seulement si $y_m(t_i)(\hat{p}) \in [y_e]$. Il s'agit donc de déterminer un ensemble de tous les \hat{p} acceptables appartenant à un espace de recherche \mathbb{P} connu a priori.

$$\mathbb{S} = \{p \in \mathbb{P} / y_m(t_i, p) \in [y_e(t_i)]\} \quad (3.16)$$

La relation (3.16) peut se réécrire avec la notation ensembliste qu'on appelle inversion ensembliste.

$$\mathbb{S} = y_m(t_i, p)^{-1} \cap [y_e(t_i)] \quad (3.17)$$

3.4 Application en robotique

Le robot scara étudié est un robot plan à deux degrés de liberté figure 3.6 pour définir sa structure on a adopté les conventions de Denavit-Hartenberg modifiées [53], les paramètres articulaires sont regroupés dans le tableau 3.1.

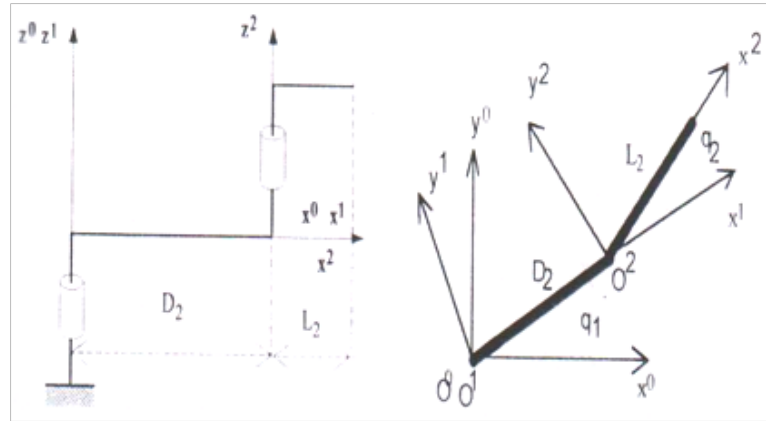


FIGURE 3.6 – Robot scara

Tableau 3.1 – Paramètres articulaires

J	σ_j	α_j	d_j	θ_j	r_j
1	0	0	0.5	q_1	0
2	0	0	0.5	q_2	0

Le modèle dynamique obtenu à partir des équations de Lagrange est donné par les équations suivantes :

$$\Gamma_1 = ZZ_1\ddot{q}_1 + ZZ_2(\ddot{q}_1 + \ddot{q}_2) + 2MX_2\cos(q_2)\ddot{q}_1 - \sin(q_2)\dot{q}_2^2 + \cos(q_2)\ddot{q}_2 - 2\sin(q_2)\dot{q}_1\dot{q}_2 + 2MY_2\sin(q_2)\ddot{q}_1 + \cos(q_2)\dot{q}_2^2 + \sin(q_2)\ddot{q}_2 + 2\cos(q_2)\dot{q}_1\dot{q}_2 + Fv_1\dot{q}_1 + Fs_1\text{sign}(\dot{q}_2)$$

(3.18)

$$\Gamma_2 = ZZ_2(\ddot{q}_1 + \ddot{q}_2) + MX_2\cos(q_2)\ddot{q}_1 + \sin(q_2)\dot{q}_1^2 + MY_2\sin(q_2)\ddot{q}_1 - \cos(q_2)\dot{q}_1^2 + Fv_2\dot{q}_2 + Fs_2\text{sign}(\dot{q}_2)$$

Ce modèle dépend de sept paramètres inertiels standards relatifs à chaque axe : Les éléments de la matrice d'inertie (ZZR_1, ZZR_2), les premiers moments d'inertie (MX_2, MY_2) ainsi que les frottements secs et visqueux (Fs_1, Fs_2, Fv_1, Fv_2)

Nb : en simulation, la valeur de MY_2 a été négligée en raison de la symétrie du second bras.

L'application de la convention de DENAVIT-HARTENBERG modifiée [52] ainsi que les méthodes de regroupements et de simplifications [33] ont permis d'obtenir un modèle de robot simplifié décrit par les paramètres de bases suivants :

$$X = [ZZR_1 \ ZZR_2 \ MX_1 \ MY_2]^T \quad (3.19)$$

Avec :

$$ZZR_1 = ZZ_1 + M_2 D_2^2 + I a_2 \quad ZZR_2 = ZZ_2 + I a_2$$

$I a_2$ et $I a_2$ représentent les moments d'inerties des moteurs relatifs à chaque articulation, M_2 la masse et D_2 la longueur du deuxième segment. En modélisant les frottements secs et visqueux, les paramètres dynamiques à identifier sont :

$$X = [ZZR1 \ Fv1 \ Fv2 \ ZZR2 \ MX2 \ MY2 \ Fv2 \ Fs2]^T \quad (3.20)$$

3.5 Identification des paramètres

3.5.1 Méthode des moindres carrés

Il existe plusieurs techniques d'identification des robots, mais la plupart présentent les trois points communs :

1/L'utilisation d'un modèle de connaissance (dynamique ou énergétique) linéaire part rapport aux paramètres à identifier :

$$Y = W.X \quad (3.21)$$

avec :

$$Y = \begin{cases} y(1) \\ \vdots \\ y(r) \end{cases} \quad et \quad W = \begin{cases} d(1) \\ \vdots \\ d(r) \end{cases}$$

2/La construction d'un système linéaire surdéterminé par échantillonnage du modèle le long de trajectoires dites « excitantes »

$$Y = W.X + \rho \quad (2.22)$$

avec ρ vecteur ($r \times 1$) des erreurs

3/ L'estimation des paramètres par minimisation d'une fonction coût, utilisant les techniques des moindres carrés ordinaires ou autres variantes. L'estimée \hat{x} de x solution du système (3.22) au sens des moindres carrés ordinaires est donnée par :

$$\hat{x} = \underset{x}{\operatorname{Argmin}} \|\rho\|^2 \quad (3.23)$$

Si la matrice d'observation W est de rang maximal, l'expression formelle de \hat{x} est donnée par :

$$\hat{x} = (W^T.W)^{-1}.W^T.Y \quad (2.24)$$

3.6 Résultats

Pour expérimenter la méthode ensembliste, nous avons élaboré un modèle simulink (cf. Annexe1) composé de deux parties, la première représente notre robot dont les consignes sont générées par une loi de commande du type proportionnel-dérivée (commande PD). Les connaissances à priori ont été prise de la thèse de C.Pressé [79] et la seconde partie représente notre modèle dynamique inverse du robot. Afin d'éviter la perte de rang numérique de la matrice d'observation W due aux perturbations, il faut garantir l'excitation des paramètres dynamiques par l'utilisation de trajectoires dites « excitantes ». Notre choix s'est porté sur un interpolateur d'ordre 5 obtenu par minimisation du conditionnement de la matrice d'observation figure 3.8.

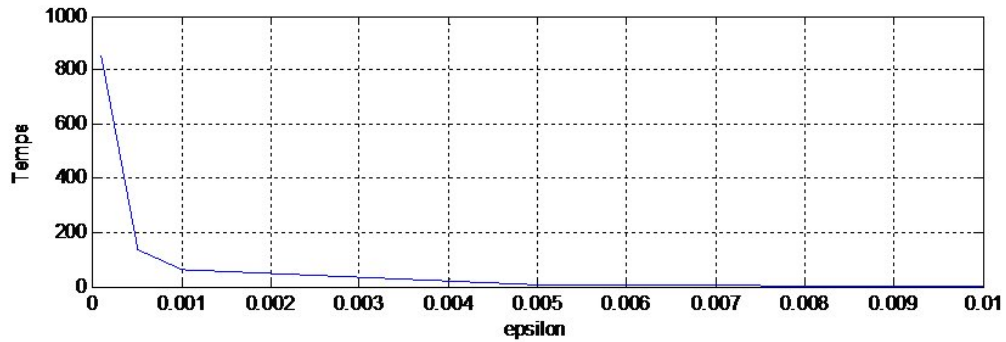
Les Couples appliqués aux axes du robot simulé ainsi que sur le modèle sont illustrés par les figure 3.9. L'application de la méthode des moindres carrés permet par la suite d'obtenir une estimation des valeurs des paramètres dynamiques du robot scara.

Ces valeurs sont regroupées dans le tableau 3.2 suivant :

Tableau 3.2 – Paramètres estimés par moindres carrés

Paramètres	p_0	\hat{p}
ZZR ₁	3.68	3.6834
Fv ₁	0.073	0.0803
Fs ₁	0.62	0.5786
ZZR ₂	0.078	0.0780
MX ₁	0.284	0.2837
MY ₂	0.000	0.000
Fv ₂	0.015	0.0118
Fs ₂	0.1560	0.1463

Pour utiliser les techniques ensemblistes, plusieurs bibliothèques et logiciels ont été élaborés pour l'implémentation du calcul par intervalle. La bibliothèque LINBOX [29] a été utilisée pour les manipulations de matrices. Sous Matlab, une toolbox INTLAB a été développée par Rump [89], il existe aussi une bibliothèque VNODELP [72] écrite en c^{++} , pour la résolution d'équations différentielles avec des intervalles, on peut aussi citer BOOST [2] et plus récemment IBEX [3] conçu pour manipuler les contracteurs. Dans notre application, nous avons utilisé Intlab sous Matlab sur un core i3. Deux solveurs, Interval peeler et Proj2d ont été élaborés par Xavier Baguenard [9] et Massa Dao [28],[4] respectivement. Ils sont basés sur le principe de propagation de contraintes et la projection de contraintes 2D. Nous les avons utilisés afin de comparer les résultats obtenus par l'inversion ensembliste sous Intlab. Pour une erreur surajoutée sur les couples, l'inversion ensembliste nous permet d'obtenir pour différentes précisions, un encadrement des solutions garanties. Un des problèmes auxquelles sont confrontées les méthodes ensemblistes est l'augmentation exponentielle du temps de simulation avec le nombre de paramètres à identifier figure 3.7

FIGURE 3.7 – Temps de simulation en fonction de la précision ϵ

3.7 Discussion

Dans un premier temps, nous avons procédé à l'identification de deux paramètres : le paramètre inertiel ZZR_1 et le frottement sec Fs_1 du premier axe, les autres ont été maintenus constants, pour les mêmes conditions de simulation, précision $\epsilon = 0.01$ et erreur sur le couple $e = 0.05$, la figure 3.10 est obtenu par l'algorithme Sivia et la figure 3.11 par SiviaP (Sivia associé à une réduction par contraintes). Le pavé initial de recherche est $:p_0 = [2, 5] \times [0.1, 0.7] \times [0.078, 0.078] \times [0.284, 0.284] \times [0.156, 0.156]$.

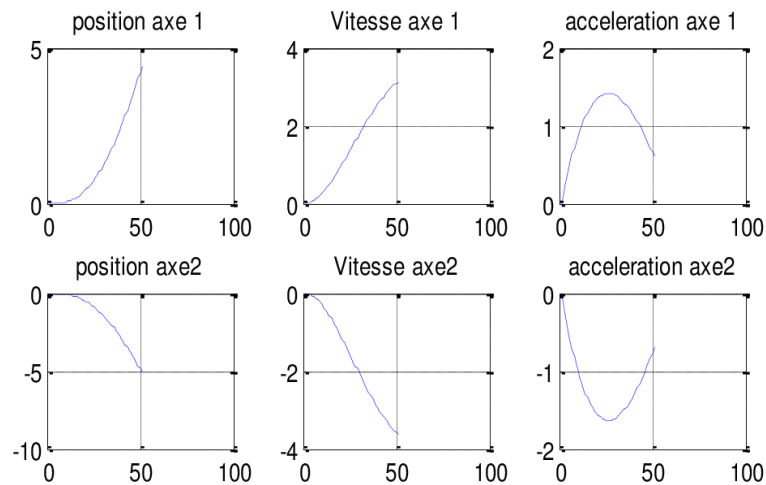


FIGURE 3.8 – Trajectoires excitantes

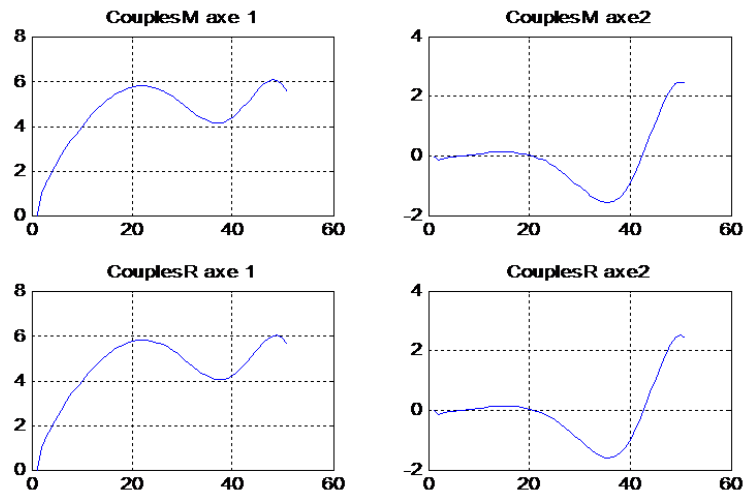
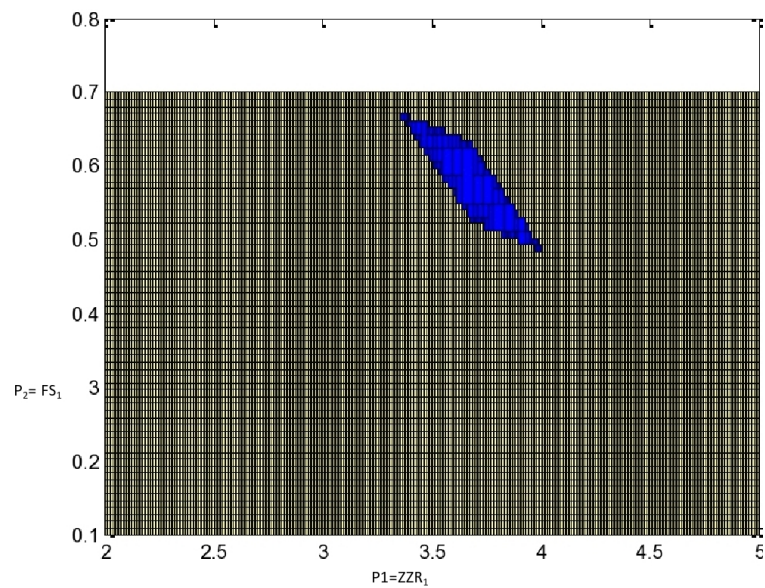


FIGURE 3.9 – Couples appliqués aux axes du robot et au modèle

FIGURE 3.10 – Sous Pavage obtenu par Sivia pour une précision $\epsilon = 0.01$

Avec $T = 787.97s$, $ZZR1 = [3.3437, 4.0157]$ et $Fs1 = [0.4812, 0.6750]$

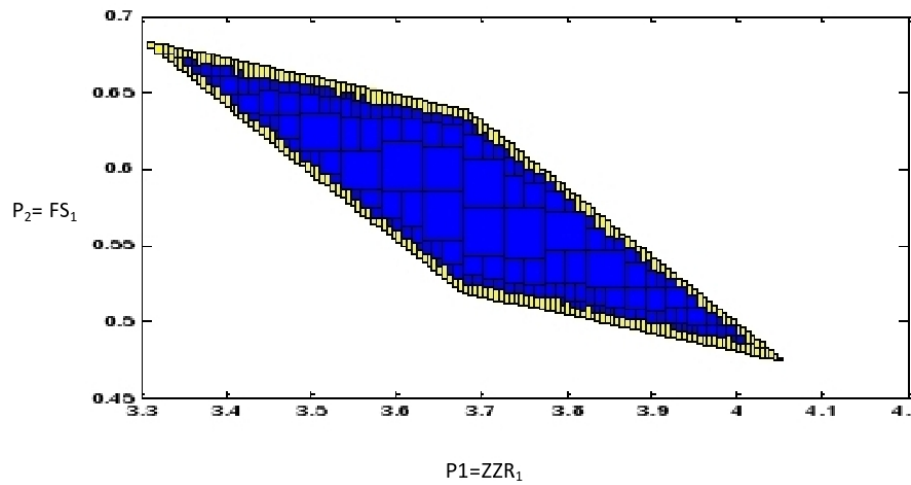
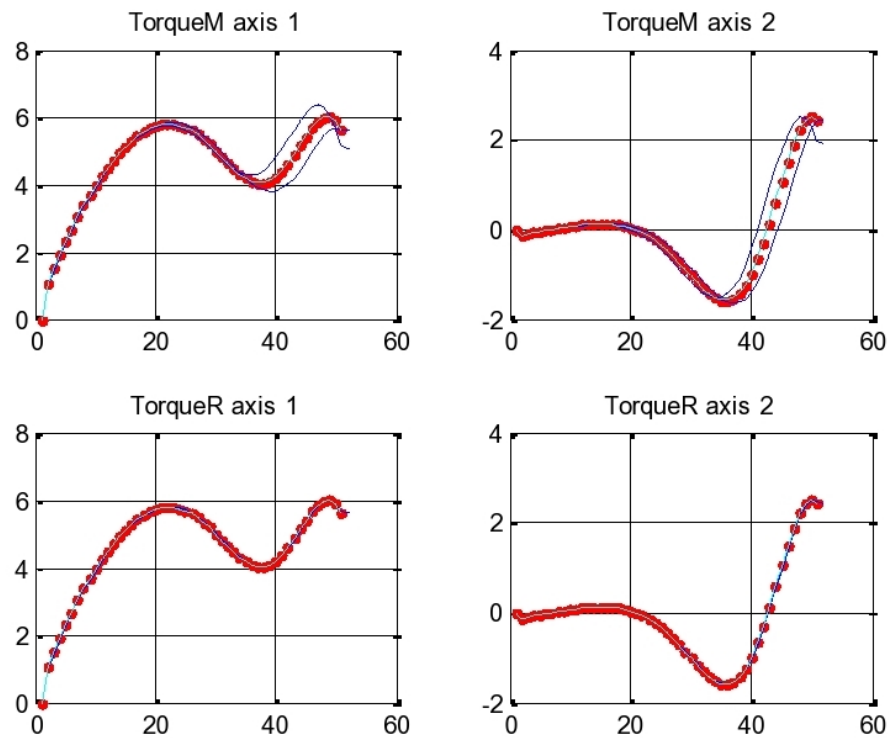


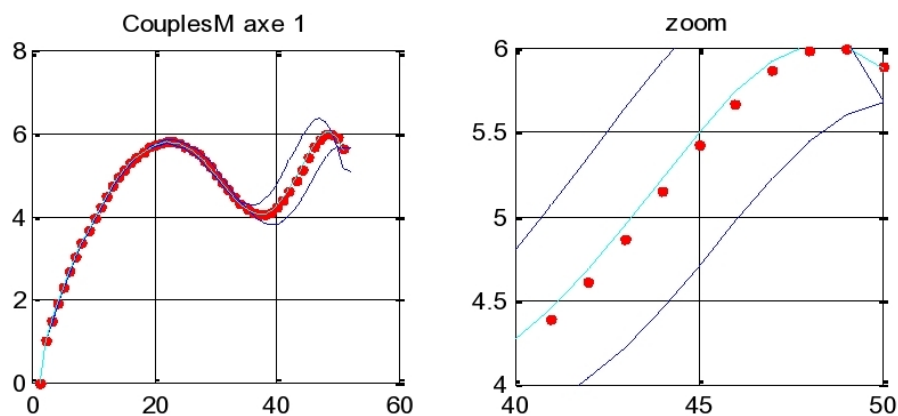
FIGURE 3.11 – Sous Pavage obtenu par SiviaP pour une précision $\epsilon = 0.01$

Avec $T = 25.1316s$, $ZZR1 = [3.3491, 4.0051]$ et $Fs1 = [0.4853, 0.6735]$

Afin de mieux expliciter le caractère garanti des résultats, on a représenté les couples obtenus à partir des valeurs aux bornes des intervalles obtenus pour chaque paramètre identifié. Le couple obtenu avec la valeur du paramètre a priori est représenté par les cercles figure.3.12-a. On trace également la trajectoire par rapport à la valeur du milieu de l'intervalle, on constate qu'il coïncide sensiblement avec le trajet initial figure 3.12-b



(a) Les couples aux bornes des intervalles



(b)

FIGURE 3.12 – (a) Couples, (b) Zoom de la Fig 3.9

On constate bien que les résultats obtenus Tableau 3.3 par SiviaP sont nettement meilleurs, l'intervalle obtenu est plus précis et le temps de calcul plus court, de plus SiviaP nous permet d'améliorer les résultats en augmentant la précision même si c'est toujours au dépend du temps qui augmente proportionnellement avec la précision. Nous avons ainsi pu identifier par la suite, en utilisant l'algorithme SiviaP, trois, puis quatre paramètres, les paramètres MY_2 , Fv_1 et Fv_2 ont été négligé pour leur faible contribution ou leur valeur pratiquement nulle. La figure 3.13 est le résultat

obtenu par Intlab pour identifier les quatre paramètres et ceci pour une erreur sur le couples $e = 0.06$ et une précision $\epsilon = 0.01$. Le pavé initial de recherche est $:p_0 = [2, 4] \times [0.2, 0.7] \times [0.1, 0.3] \times [0.1, 0.3]$

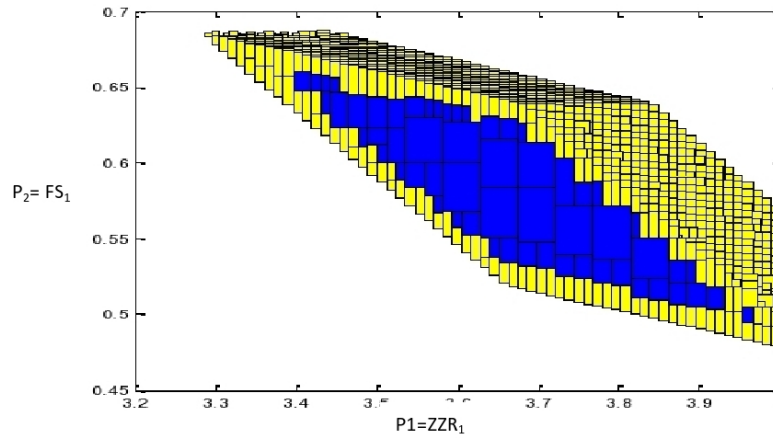


FIGURE 3.13 – Sous Pavage obtenu par SiviaP avec une précision $\epsilon = 0.05$

Avec $T = 28422.26s$, $ZZR1 = [3.4231, 4.000]$ et $Fs1 = [0.5051, 0.6683]$

On a regroupé les résultats concernant l'identification des deux paramètres dans le tableau suivant :

Tableau 3.3 – Identification des paramètres ZZR_1 et $Fs1$

Paramètres	\hat{p}	Sivia	SiviaP	Proj2d	Intervalpeeler
ZZR_1	3.6834	[3.3437,4.0157]	[3.3491,4.0051]	[3.4439,3.9163]	[3.4215,3.9386]
Fs_1	0.5786	[0.4812,0.6750]	[0.4853,0.6735]	[0.55040,0.6077]	[0.54808,0.6100]
T (s)	1.7	787.977	25.1316	10.54	0.016
Précision ϵ	–	0.01	0.01	0.001	

La Figure 3.14 et la Figure 3.15 représentent la projection des paramètres ZZR_1 et Fs_2 obtenus par Proj2d dans les mêmes conditions de simulation que ceux réalisées sous Intlab. Par contre dans ce cas là, on a pu obtenir des résultats avec une erreur $e = 0.01$ et une précision allant jusqu'à $\epsilon = 0.001$. ZZR_2 est le seul paramètre qui n'a pas pu être identifié, probablement à cause de sa valeur très faible. Tous les résultats pour l'identification de quatre paramètres sont regroupés dans le tableau 3.4.

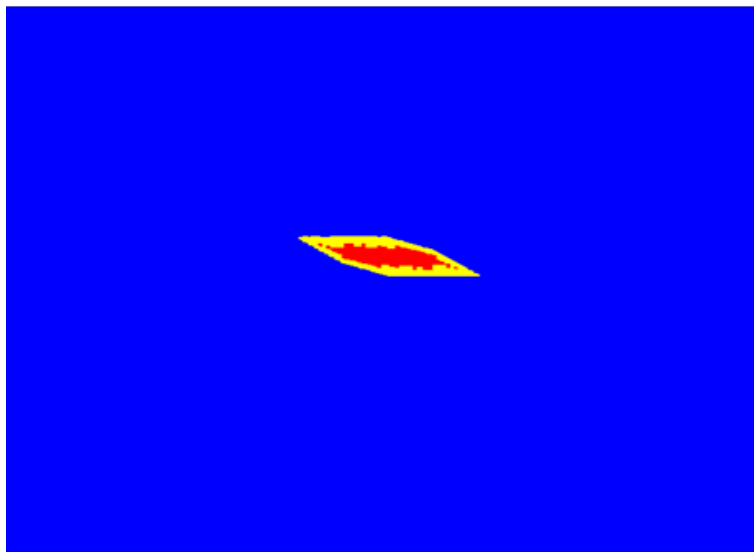


FIGURE 3.14 – Sous Pavage obtenu par Proj2d avec une précision $\epsilon = 0.01$ pour identifier les 2 paramètres : ZZR_1 et F_{s_1}

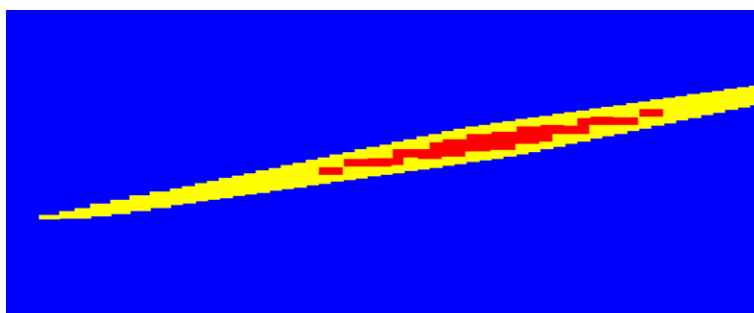


FIGURE 3.15 – Sous Pavage obtenu par Proj2d, avec une précision $\epsilon = 0.01$ projection de MX_2 et F_{s_2}

Tableau 3.4 – Résultats comparatifs

Paramètres	\hat{p}	SiviaP	Proj2d	Intervalpeeler
ZZR_1	3.6834	[3.4231,4.000]	[3.4695,3.9440]	[3.0670,4.3880]
F_{s_1}	0.5786	[0.5051,0.6683]	[0.5543,0.6019]	[0.4714,0.685]
MX_1	0.2837	[0.1750,0.3]	[0.2043,0.3565]	[0.0519,0.4]
F_{s_2}	0.1463	[0.1,0.1624]	[0.1325,0.1797]	[0.0912,0.1957]
T (s)	–	28422.267	148.778	0.03

3.8 Conclusion

Dans ce chapitre, nous avons fait un rappel des méthodes classiques d'identification, afin de procéder à une comparaison avec l'approche ensembliste, particulièrement le contexte à erreurs bornées. On est arrivé à la conclusion que l'approche ensembliste permet de manipuler le non linéaire plus aisément, puisqu'elle ne nécessite pas de calculs compliqués de lois de probabilité, ni aucune connaissance statistique sur les variables aléatoires. Cependant, dans le cas de problèmes linéaire, gaussiens l'approche probabiliste est si simple et efficace que l'approche ensembliste n'apporte aucun intérêt supplémentaire. Nous avons par la suite décrit l'identification dans le contexte à erreurs bornées en utilisant l'inversion ensembliste. L'exemple de l'identification de paramètres dynamique d'un robot Scara par inversion ensembliste a illustré cette approche.

Chapitre 4

Applications au bioréacteur

Dans ce chapitre, nous présentons deux méthodes d'identification ensemblistes pour estimer les paramètres d'un chimostat. La première concerne l'intégration numérique garantie utilisant la théorie de comparaison des inégalités différentielles que nous présentons en première partie, puis dans la seconde partie, la méthode LSCR (Leave-out Sign-dominant Correlation Regions), une méthode basée sur le calcul des fonctions de corrélation associées à l'inversion ensembliste qui va permettre de caractériser les solutions de manière garantie.

Première partie

Théorie de comparaison des inégalités différentielles

4.1 Introduction

Dans ce chapitre, on s'intéresse à la résolution d'équations différentielles en associant les méthodes ensemblistes. En effet, les techniques ensemblistes basées sur l'arithmétique des intervalles permettent de déterminer un encadrement garanti des solutions des équations différentielles incertaines dont le principe est de transformer un problème incertain en deux problèmes déterministes. La résolution de ce problème déterministe permet l'obtention de deux solutions minimales et maximales qu'on nomme trajectoires englobantes et qui encadrent toutes les trajectoires solutions de manière garantie. Pour se faire, on utilise la théorie des inégalités différentielles développée par [42; 90] pour les systèmes autonomes et celle utilisant le théorème de Müller [71; 62; 96] dans le cas général.

4.2 Définitions

4.2.1 Systèmes monotones

Les équations différentielles sont dit monotones si les solutions d'état obtenues respectent une relation d'ordre " \geq " définie sur un domaine précis et tenant compte des conditions initiales.

soit l'équation différentielle :

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

soit $z(t, z_0, u_1)$ et $y(t, y_0, u_2)$ deux trajectoires de solutions de (4.1)

A partir des conditions initiales :

$x(t_0) = z_0, u(t) = u_1$ et $x(t_0) = y_0, u(t) = u_2$, respectivement, le système est dit monotone si :

$$y_0 \geq z_0 \text{ et } u_2 \geq u_1 \Rightarrow y(t, y_0, u_2) \geq z(t, z_0, u_1) \quad \forall t \geq t_0 \quad (4.2)$$

Remarque

Si n et m représentent les dimension de x et u , on a :

$$y_{0,i} \geq z_{0,i} \quad \forall i \in \{1, \dots, n\} \quad \text{et} \quad u_{2,j} \geq u_{1,j} \quad \forall j \in \{1, \dots, m\} \quad (4.3)$$

Ce qui veut dire que la relation d'ordre " \geq " doit être respectée composante par composante. La vérification de la monotonie d'un système peut se faire à travers de la notion de préservation d'ordre.

4.2.2 Systèmes préservateurs d'ordre

Soit une relation d'ordre quelconque, un système d'équations différentielles est dit préservateur d'ordre par rapport à cette relation d'ordre s'il est possible de déterminer deux matrices diagonales de dimensions $n \times n$ et $m \times m$ telle que :

$$D = \text{diag}[(-1)^{\epsilon_1}, \dots, (-1)^{\epsilon_n}] \quad \epsilon_i \in \{0, 1\}$$

et

$$(4.4)$$

$$P = \text{diag}[(-1)^{\epsilon_1}, \dots, (-1)^{\epsilon_m}] \quad \epsilon_j \in \{0, 1\}$$

tel que :

$$Dy_0 \geq Dz_0 \quad \text{et} \quad Pu_2 \geq Pu_1 \Rightarrow Dy(t, y_0, u_2) \geq Dy(t, z_0, u_1) \quad \forall t \geq t_0 \quad (4.5)$$

le principe de préservation d'ordre est basé sur une méthode graphique qu'on va décrire dans le paragraphe suivant.

4.2.3 Représentation graphique

Afin de vérifier la préservation d'ordre, on construit un graphe d'incidence associé à l'équation (4.1). Ce graphe est constitué d'un ensemble de nœuds représentant les variables x_1, \dots, x_n et u_1, \dots, u_m et d'un ensemble d'arcs signés $a_{i,j}^x$ reliant les nœuds x_i et x_j puis des arcs $a_{i,j}^u$ reliant x_i et u_k . Pour construire ce graphe, on doit appliquer la règle suivante :

■ Arcs entre les nœuds x_i et u_k

1. Si $\frac{\partial f_i}{\partial u_k} > 0 \forall (x, u) \in \mathbb{X} \times \mathbb{U}$ un arc $a_{i,j}^u$ positif et orienté, relie le nœud u_k au nœud x_i
2. Si $\frac{\partial f_i}{\partial u_k} < 0 \forall (x, u) \in \mathbb{X} \times \mathbb{U}$ un arc $a_{i,j}^u$ négatif et orienté, relie le nœud u_k au nœud x_i
3. Si $\frac{\partial f_i}{\partial u_k} = 0 \forall (x, u) \in \mathbb{X} \times \mathbb{U}$ il n'y a aucun arc $a_{i,j}^u$ entre x_i et u_k

■ Arcs entre les noeuds x_i

1. Si $\frac{\partial f_i}{\partial x_j} > 0 \forall (x, u) \in \mathbb{X} \times \mathbb{U}$ un arc $a_{i,j}^x$ positif et orienté, relie le nœud x_j au nœud x_i
2. Si $\frac{\partial f_i}{\partial x_j} < 0 \forall (x, u) \in \mathbb{X} \times \mathbb{U}$ un arc $a_{i,j}^x$ négatif et orienté, relie le nœud x_j au nœud x_i
3. Si $\frac{\partial f_i}{\partial x_j} = 0 \forall (x, u) \in \mathbb{X} \times \mathbb{U}$ il n'y a aucun arc $a_{i,j}^x$ entre x_j et x_i

Par ailleurs, on définit un cycle comme étant une séquence finie de noeuds z_1, z_2, \dots, z_n tels que $z_1 = z_n$ et $a_{i,j}$ un arc, pas particulièrement orienté, qui relie deux nœuds successifs. Le signe d'un cycle est le produit entre les signes des arcs de chaque cycle.

Propriété 4.1

Un système d'équations différentielles défini par (4.1) et admettant un graphe d'incidence réalisé en appliquant 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 [59].

Exemple 4.1

soit le système d'équations suivant[63] défini dans $\mathbb{R}^2 \geq 0$:

$$\begin{cases} \dot{x}_1 = x_1(-x_1 + x_2) \\ \dot{x}_2 = 3x_2(-x_1 + u) \end{cases} \quad (4.6)$$

Si la première équation est nommée f_1 et la deuxième f_2 , on calcule les dérivées partielles suivantes en déduisant leur signe :

$$\frac{\partial f_1}{\partial x_2} = x_1 > 0$$

$$\frac{\partial f_1}{\partial u} = 0$$

$$\frac{\partial f_2}{\partial x_1} = -3x_2 < 0$$

$$\frac{\partial f_2}{\partial u} = 3x_2 > 0$$

Ce qui se traduit par un arc : positif de x_2 vers x_1 , négatif x_1 vers x_2 , positif u vers x_1 et aucun arc entre x_1 et u puisque la dérivée est nulle.

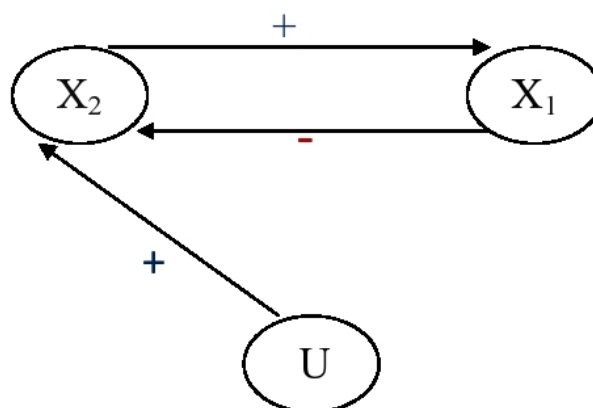


FIGURE 4.1 – graphe de représentation d'ordre de l'équation 4.6

Ce graphe contient un seul cycle dont le signe est :

$$\text{sign}(a_{1,2})\text{sign}(a_{2,1}) < 0 \quad (4.7)$$

Dans ce cas le système n'est pas préservateur d'ordre donc il n'est pas monotone.

Exemple 4.2

Dans l'exemple suivant [63] :

$$\begin{cases} \dot{x}_1 = -u\alpha_1(x_1) + \alpha_2(1 - x_1 - x_2) \\ \dot{x}_2 = u\alpha_3(1 - x_1 - x_2) - \alpha_4(x_2) \end{cases} \quad (4.8)$$

tel que $\alpha_i(0) = 0$ et toutes les $\alpha_i(\cdot)$ ont des dérivées strictement positives, de plus le système (4.8) est défini sur le domaine :

$$\mathbb{S} = \{x_1 > 0, x_2 > 0, x_1 + x_2 \leq 1\}$$

Le graphe représentant ce système est le suivant :

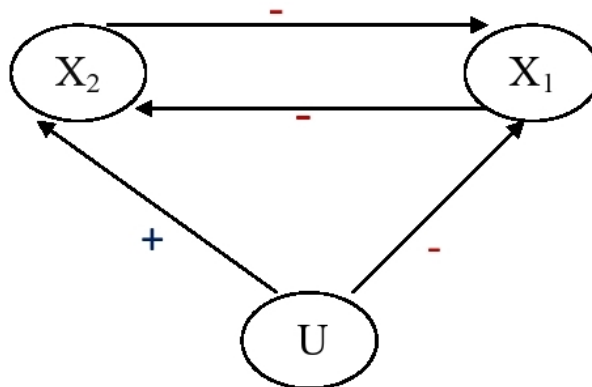


FIGURE 4.2 – graphe de représentation d'ordre de l'équation 4.8

Ce graphe contient trois cycles :

1. cycle 1 (x_1, x_2, x_1) avec $\text{sign}(a_{1,2}^x) \cdot \text{sign}(a_{2,1}^x) > 0$

2. cycle 2 (x_1, x_2, u, x_1) avec $\text{sign}(a_{1,2}^x) \cdot \text{sign}(a_{1,2}^u) \cdot \text{sign}(a_{1,1}^u) > 0$
3. cycle 3 (x_1, x_2, u, x_1) avec $\text{sign}(a_{2,1}^x) \cdot \text{sign}(a_{1,2}^u) \cdot \text{sign}(a_{1,1}^u) > 0$

Ce système est préservateur d'ordre, donc il est monotone vis à vis d'une certaine relation d'ordre définie sous certaines conditions initiales.

4.2.4 Systèmes coopératifs

Les systèmes dynamiques dans lesquels les variables d'état ont un comportement positif les uns vis-à-vis des autres sont appelés systèmes coopératifs. Cette propriété est vérifiée par un certains nombres de modèles biologiques, chimiques ou économiques [90].

Propriété 4.2

Soit f une fonction continûment différentiable, le système (4.1) est coopératif si est seulement si, il satisfait les conditions suivantes :

$$\begin{aligned} \frac{\partial f_i}{\partial x_j}(x, u) &\geq 0 \quad \forall (x) \in \mathbb{X}, \quad \forall (u) \in \mathbb{U}, \quad \forall i \neq j \\ \frac{\partial f_i}{\partial u_j}(x, u) &\geq 0 \quad \forall (x) \in \mathbb{X}, \quad \forall (u) \in \mathbb{U} \end{aligned} \tag{4.9}$$

pour tout $i \in \{1, \dots, n\}$ et $j \in \{1, \dots, m\}$

Remarque

Par définition, les systèmes coopératifs sont des systèmes monotones. Cependant des systèmes non coopératifs peuvent être monotones s'ils sont préservateurs d'ordre. Dans ce cas, il existe une matrice de changement de base qui peut transformer le système non coopératif en un système coopératif dans un nouveau système de coordonnées.

Propriétés 4.3

1. La première propriété intéressante des systèmes dynamiques coopératifs est le principe de comparaison d'opérateurs différentiels [96] qui est énoncé comme suit :

Considérons deux systèmes coopératifs :

$$\dot{x}(t) = h(x(t), u(t)) \tag{4.10}$$

$$\dot{y}(t) = g(y(t), w(t))$$

ou $x, y \in \mathbb{X}$ et $u, w \in \mathbb{U}$, on a alors :

$$(h_i(x, u) \leq g_i(x, u)) \wedge (x_{0_i} \leq y_{0_i}) \quad \forall i \in \{1, \dots, n\}, \quad \forall (x) \in \mathbb{X}, \forall (u) \in \mathbb{U}$$

$$\Rightarrow \forall i \in \{1, \dots, n\}, \quad x_i(t) \leq y_i(t) \quad \forall t \geq t_0 \tag{4.11}$$

2. La deuxième propriété est la positivité des systèmes coopératifs. En effet, soit le système d'équations différentielles linéaires positives.

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \tag{4.12}$$

où $A(t)$ est une matrice coopérative, ce qui veut dire que tous ses éléments non diagonaux sont tous positifs. Il en est de même pour les éléments de $B(t)$.

$$a_{i,j}(t) \geq 0, \quad \forall i \neq j, \quad \forall t \geq t_0$$

$$b_{i,j}(t) \geq 0, \quad \forall (i, j), \quad \forall t \geq t_0 \tag{4.13}$$

On a :

$$\forall (i, j), \quad u_{i,j} \geq 0 \text{ et } x_{0_i} \geq 0 \Rightarrow x_i(t) \geq 0, \quad \forall t > t_0, \quad \forall i \tag{4.14}$$

3. Des deux propriétés précédentes, on déduit :

Pour toute entrée $u(t) \geq 0, \forall t \geq t_0$, une équation différentielle ordinaire coopérative, vérifie la propriété suivante :

$$\forall x(t_0) \in \mathbb{X}, f(x(t_0), u(t_0)) \geq 0 \Rightarrow \forall t \geq t_0, f(x(t), u(t)) \geq 0 \quad (4.15)$$

Pour la démonstration de ces propriétés, se référer à [90]. Ces propriétés trouvent une application intéressante dans le cadre de l'intégration numérique garantie des EDO incertaines.

4.3 Intégration numérique garantie des ODE

Le principe de l'intégration numérique garantie est de résoudre des EDO incertaines pour lesquelles les conditions initiales appartiennent à un intervalle, en les transformant dans une première étape, en deux ODE certaines ou déterministes, l'une maximale et l'autre minimale.

4.3.1 Cas des systèmes non monotones

Dans le cas général où les systèmes sont non monotones, le théorème de Müller [71; 64] permet la construction de trajectoires englobantes afin d'obtenir de manière garantie, un encadrement de toutes les solutions de l'EDO non monotones et incertaines.

$$\left\{ \begin{array}{l} \dot{x}(t) = f(x, p, u(t)) \\ x_0 \in [x_0] \subseteq \mathbb{X} \\ p \in [p] \\ u(t) \in \mathbb{X} \end{array} \right. \quad (4.16)$$

Avec p un paramètre incertain de dimension r .

Dans le paragraphe suivant, nous présentons un théorème qui nous permet de vérifier les conditions [55; 63] pour lesquelles la résolution des ODE certaines permet d'obtenir un encadrement garanti de toutes les solutions du système (4.16)

Theorem 3. Soient deux vecteurs $\underline{\Theta}$ et $\overline{\Theta}$ de dimension inférieure ou égale à $r \times n$ tels que :

$$\forall i \in \{1, \dots, n\}, f_i(x, \underline{\Theta}, \cdot) \leq f_i(x, p, \cdot) \leq f_i(x, \overline{\Theta}, \cdot) \quad (4.17)$$

les solutions $\underline{x}(t)$ et $\overline{x}(t)$ des ODE coopératives certaines du système (4.16) sont données par :

$$\dot{\underline{x}}(t) = f(\underline{x}, \underline{\Theta}, u_1(t)) \quad (4.18)$$

$$\dot{\overline{x}}(t) = f(\overline{x}, \overline{\Theta}, u_2(t)) \quad (4.19)$$

Tenant compte de la propriété (4.3) des systèmes coopératifs :

si pour $\forall i \in \{1, \dots, n\}$, et si $\underline{x}_i(t_0) \leq \overline{x}_i(t_0)$

$\forall i, \forall t \geq t_0, \forall p \in [p], \forall u(t) \in [u_1(t), u_2(t)]$ et $\forall x(t_0) \in [\underline{x}(t_0), \overline{x}(t_0)]$

$$\underline{x}_i(t_0) \leq x_i(t) \leq \overline{x}_i(t_0) \quad (4.20)$$

En d'autres termes, tel qu'il a été défini en arithmétique des intervalles, on détermine une fonction d'inclusion des solutions des équations (4.16).

$$x_i(t) = [\underline{x}_i(t), \overline{x}_i(t)] \quad (4.21)$$

$\underline{x}_i(t)$ représentent les solutions minimales et $\overline{x}_i(t)$ les solutions maximales de l'équation différentielle (4.16). Cependant, afin de déterminer ces deux solutions minimales et maximales, on a besoin de bien déterminer $\underline{\Theta}$ et $\overline{\Theta}$ pour assurer un encadrement garanti. Pour cela on doit appliquer une règle de sélection [55] pour la construction de ces deux vecteurs.

Remarque

Si la fonction f est Lipschitzienne par rapport à x , alors pour tout $x_0 \in [x_0]$, $p \in [p]$ et $u(t) \in \mathbb{U}$, l'unicité de la solution est garantie.

Règle 4.1

Le principe de cette règle est le calcul du maximum \bar{f} et du minimum \underline{f} d'une fonction f sur le domaine de paramètre $[p]$, vérifiant par conséquent le théorème 3. Cette règle s'applique aux systèmes dont le signe des dérivées partielles de la fonction f par rapport aux vecteurs des paramètres p sont constants. On analyse donc le signe des dérivées partielles $\frac{\partial f_i}{\partial p_k}$

Pour $\forall i \in \{1, \dots, n\}$ et pour $\forall k \in \{1, \dots, r\}$, On définit les vecteurs $\underline{\theta}_i^k$ et $\bar{\theta}_i^k$ de la façon suivante :

$$\underline{\theta}_i^k = \begin{cases} \underline{p}_k & \text{si } \frac{\partial f_i}{\partial p_k}(\cdot, \dots, \cdot) \geq 0 \quad \forall x \in \mathbb{X}, \forall p \in [p], \forall u(t) \in \mathbb{U} \\ \bar{p}_k & \text{si } \frac{\partial f_i}{\partial p_k}(\cdot, \dots, \cdot) < 0 \quad \forall x \in \mathbb{X}, \forall p \in [p], \forall u(t) \in \mathbb{U} \end{cases} \quad (4.22)$$

$$\bar{\theta}_i^k = \begin{cases} \bar{p}_k & \text{si } \frac{\partial f_i}{\partial p_k}(\cdot, \dots, \cdot) \geq 0 \quad \forall x \in \mathbb{X}, \forall p \in [p], \forall u(t) \in \mathbb{U} \\ \underline{p}_k & \text{si } \frac{\partial f_i}{\partial p_k}(\cdot, \dots, \cdot) < 0 \quad \forall x \in \mathbb{X}, \forall p \in [p], \forall u(t) \in \mathbb{U} \end{cases} \quad (4.23)$$

Et donc on définit les vecteurs :

$$\begin{aligned} \underline{\Theta} &= [\underline{\theta}_1^1, \dots, \underline{\theta}_n^1, \dots, \underline{\theta}_n^r] \\ \text{et} & \\ \bar{\Theta} &= [\bar{\theta}_1^1, \dots, \bar{\theta}_n^1, \dots, \bar{\theta}_n^r] \end{aligned} \quad (4.24)$$

L'apparition de la borne inférieure et la borne supérieure du même paramètre dans l'une des composantes du champ de vecteur \underline{f} ou \bar{f} [84] provoque une instabilité

des ODE englobantes, ce qui produit des solutions divergentes. L'application de cette règle permet d'éviter ce problème.

Exemple 4.3

Soit un système à compartiments modélisé par les ODE coopératives incertaines suivantes :

$$\begin{cases} \dot{x}_1 = -p_4 x_1 - \frac{p_1 x_1}{1+p_2 x_2} + p_3 x_2 + u \\ \dot{x}_2 = p_4 x_1 - p_3 x_2 \end{cases} \quad (4.25)$$

Les conditions initiales incertaines x_0 appartiennent à l'intervalle suivant : $x_0 \in [1, 1.4] \times [0, 0.4]$ et les paramètres bruités $p \in [0.9, 1.1] \times [1.1, 1.3] \times [0.45, 0.55] \times [0.2, 0.3]$. L'entrée à ce système est une constante $u(t) = 0.1$ sur une durée $t = [0, 20]$

En appliquant la règle 4.1, on obtient les deux EDO englobantes suivantes :

$$\begin{cases} \dot{\underline{x}}_1 = -\bar{p}_4 \underline{x}_1 - \frac{\bar{p}_1 \underline{x}_1}{1+\bar{p}_2 \underline{x}_2} + \underline{p}_3 \underline{x}_2 + u \\ \dot{\underline{x}}_2 = \underline{p}_4 \underline{x}_1 - \bar{p}_3 \underline{x}_2 \end{cases} \quad (4.26)$$

$$\begin{cases} \dot{\bar{x}}_1 = -\underline{p}_4 \bar{x}_1 - \frac{\underline{p}_1 \bar{x}_1}{1+\underline{p}_2 \bar{x}_2} + \bar{p}_3 \bar{x}_2 + u \\ \dot{\bar{x}}_2 = \bar{p}_4 \bar{x}_1 - \underline{p}_3 \bar{x}_2 \end{cases} \quad (4.27)$$

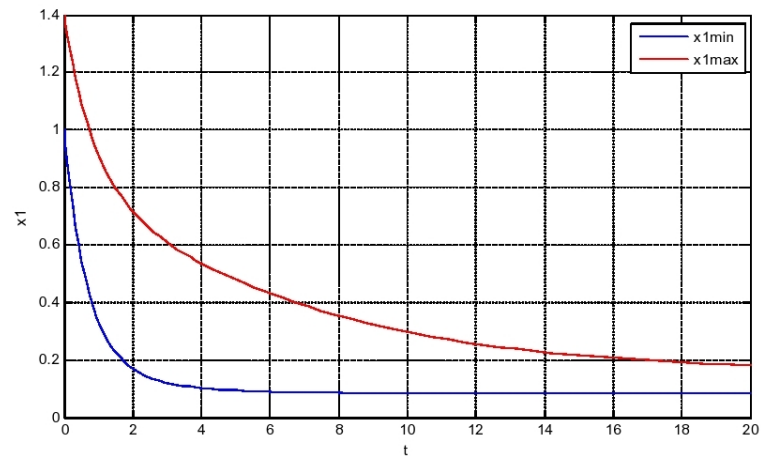
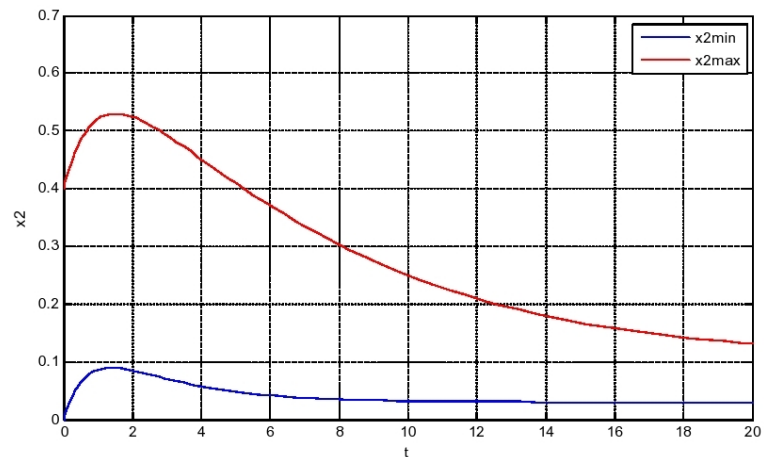
Dans ce cas les vecteurs $\underline{\Theta}$ et $\bar{\Theta}$ sont définis par :

$$\underline{\Theta} = [\underline{\theta}_1^1, \underline{\theta}_2^1, \underline{\theta}_3^1, \underline{\theta}_4^1, \underline{\theta}_3^2, \underline{\theta}_4^2]^T = [\bar{p}_1, \underline{p}_2, \underline{p}_3, \bar{p}_4, \bar{p}_3, \underline{p}_4]^T$$

et

$$\bar{\Theta} = [\bar{\theta}_1^1, \bar{\theta}_2^1, \bar{\theta}_3^1, \bar{\theta}_4^1, \bar{\theta}_3^2, \bar{\theta}_4^2]^T = [\underline{p}_1, \bar{p}_2, \bar{p}_3, \underline{p}_4, \underline{p}_3, \bar{p}_4]^T$$

Les figures suivantes (4.3) et (4.4) représentent respectivement, un encadrement garanti de toutes les solutions de $x_1(t)$ et celles de $x_2(t)$ obtenu par l'application de la règle (4.1)

FIGURE 4.3 – Trajectoires englobantes de la solution x_1 FIGURE 4.4 – Trajectoires englobantes de la solution x_2

4.3.2 Théorème de Müller

La règle (4.1) précédente s'applique au ODE coopératives, elle permet donc de construire les ODE minimales et maximales chacune séparément en appliquant le principe de comparaison (Propriétés 4.3). Cependant dans le cas générale où les fonctions sont non monotones, les ODE englobantes sont construites simultanément en appliquant le théorème suivant [71; 55; 64].

Theorem 4. Soient les équations différentielles définies par :

$$\begin{cases} \dot{X}(t) = f(X(t), U(t)) \\ X_0 \in [X_0] \subseteq X \\ P \in [P] \end{cases} \quad (4.29)$$

On définit un domaine \mathbb{T} en fonction d'un champs de vecteur f continu sur ce domaine :

$$\mathbb{T} \begin{cases} \omega(t) \leq X(t) \leq \Omega(t) \\ \underline{P} \leq P \leq \bar{P} \\ t_0 \leq t \leq t_{n_T} \end{cases} \quad (4.30)$$

Avec :

1. $\omega(t_0) = \underline{x}_0$ et $\Omega(t_0) = \bar{x}_0$
 2. $\forall i, D^\pm \omega_i(t) \leq \min_{\mathbb{T}} f_i(x, p, t)$
 3. $\forall i, D^\pm \Omega_i(t) \leq \max_{\mathbb{T}} f_i(x, p, t)$
- (4.31)

Les fonctions $\omega_i(t)$ et $\Omega_i(t)$ telles que : $i = 1, \dots, n$ sont continues sur l'intervalle $[t_0, t_{n_T}]$.

$D^- \omega_i(t)$ et $D^- \Omega_i(t)$ représentent les dérivées à gauches, $D^+ \omega_i(t)$ et $D^+ \Omega_i(t)$ représentent les dérivées à droites.

\mathbb{T}_i est un domaine de \mathbb{T} , et il est défini par,

$$\mathbb{T}_i \begin{cases} X_i = \omega_i(t) \\ \omega_j(t) \leq X_j \leq \Omega_j(t), i \neq j \\ \underline{P} \leq P \leq \bar{P} \end{cases} \quad (4.32)$$

et $\bar{\mathbb{T}}_i$ et un domaine \mathbb{T} , défini par

$$\mathbb{T}_i \begin{cases} X_i = \Omega_i(t) \\ \omega_j(t) \leq X_j \leq \Omega_j(t), i \neq j \\ \underline{P} \leq P \leq \overline{P} \end{cases} \quad (4.33)$$

ainsi $\forall X_0 \in [\underline{X}_0, \overline{X}_0]$ et $P \in [\underline{P}, \overline{P}]$, une solution de $X(t)$ existe et reste dans le domaine suivant :

$$\mathbb{X} \begin{cases} \omega(t) \leq X(t) \leq \Omega(t) \\ t_0 \leq t \leq t_{n_t} \end{cases} \quad (4.34)$$

4.3.3 Règle de construction des solutions minimales et maximales

L'application du théorème 3 nous permet de construire des solutions minimales et maximales des ODE en appliquant la règle suivante proposée par Kieffer et al. puis Ramdani et al. [55; 83; 84].

Règle 4.2

1. Pour $\forall i \in \{1, \dots, n\}$ et pour $\forall k \in \{1, \dots, r\}$, On définit les vecteurs $\underline{\theta}_i^k$ et $\overline{\theta}_i^k$ de la façon suivante :

$$\underline{\theta}_i^k = \begin{cases} \underline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(\cdot, \cdot, \cdot) \geq 0 \forall x \in \mathbb{X}, \forall p \in [p], \forall u(t) \in \mathbb{U} \\ \overline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(\cdot, \cdot, \cdot) < 0 \forall x \in \mathbb{X}, \forall p \in [p], \forall u(t) \in \mathbb{U} \end{cases} \quad (4.35)$$

$$\overline{\theta}_i^k = \begin{cases} \overline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(\cdot, \cdot, \cdot) \geq 0 \forall x \in \mathbb{X}, \forall p \in [p], \forall u(t) \in \mathbb{U} \\ \underline{p}_k \text{ si } \frac{\partial f_i}{\partial p_k}(\cdot, \cdot, \cdot) < 0 \forall x \in \mathbb{X}, \forall p \in [p], \forall u(t) \in \mathbb{U} \end{cases} \quad (4.36)$$

Avec les vecteurs :

$$\begin{aligned} \underline{\Theta} &= [\theta_1^1, \dots, \theta_n^1, \dots, \theta_n^r] \\ \text{et} \\ \overline{\Theta} &= [\bar{\theta}_1^1, \dots, \bar{\theta}_n^1, \dots, \bar{\theta}_n^r] \end{aligned} \quad (4.37)$$

2. $\forall i \in \{1, \dots, n\}$ et $j \in \{1, \dots, n\}$, On définit le vecteur $\bar{\Gamma}$ tel que : $Dim(\bar{\Gamma}) \leq (n \times n)$.
Les éléments $\bar{\gamma}_i^j$ de ce vecteur sont obtenus de la façon suivante :

$$\bar{\gamma}_i^j = \begin{cases} \Omega_i & \text{si } i = j \\ \Omega_i & \text{si } i \neq j \wedge \frac{\partial f_i}{\partial x_j} \geq 0 \\ \omega_i & \text{si } i \neq j \wedge \frac{\partial f_i}{\partial x_j} < 0 \end{cases} \quad \forall x \in \mathbb{X}, \forall p \in [p] \quad (4.38)$$

De la même façon, on obtient les éléments de $\underline{\Gamma}$

$$\underline{\gamma}_i^j = \begin{cases} \omega_i & \text{si } i = j \\ \omega_i & \text{si } i \neq j \wedge \frac{\partial f_i}{\partial x_j} \geq 0 \\ \Omega_i & \text{si } i \neq j \wedge \frac{\partial f_i}{\partial x_j} < 0 \end{cases} \quad \forall x \in \mathbb{X}, \forall p \in [p] \quad (4.39)$$

Les vecteurs $\bar{\Gamma}$ et $\underline{\Gamma}$ sont définis par :

$$\begin{aligned} \bar{\Gamma} &= [\bar{\gamma}_1^1, \dots, \bar{\gamma}_n^1, \dots, \bar{\gamma}_n^n] \\ \text{et} \\ \underline{\Gamma} &= [\underline{\gamma}_1^1, \dots, \underline{\gamma}_n^1, \dots, \underline{\gamma}_n^n] \end{aligned} \quad (4.40)$$

Ce qui va permettre d'obtenir par la suite, les équations différentielles vérifiant les conditions du théorème de Müller. On les décrit de la façon suivante :

$$i = 1, \dots, n \left\{ \begin{aligned} \dot{\omega}_i(t) &= \underline{f}_i(\underline{\Gamma}, \underline{\Theta}, t) \\ \dot{\Omega}_i(t) &= \bar{f}_i(\bar{\Gamma}, \bar{\Theta}, t) \end{aligned} \right. \quad (4.41)$$

Les solutions $\omega(t)$ et $\Omega(t)$ des équations différentielles sont données par les expressions suivantes :

$$\left\{ \begin{array}{l} \dot{\omega}_i(t) = \underline{f}_i(\gamma, \omega, \underline{p}, \bar{p}, t) \\ \dot{\Omega}_i(t) = \bar{f}_i(\gamma, \Omega, \underline{p}, \bar{p}, t) \\ \omega(t_0) = \underline{x}_0 \\ \Omega(t_0) = \bar{x}_0 \end{array} \right. \quad (4.42)$$

Sachant que :

$$\begin{aligned} \underline{f}_i(\gamma, \omega, \underline{p}, \bar{p}, t) &= \underline{f}_i(\underline{\Gamma}, \underline{\Theta}, t) \\ \text{et} & \\ \bar{f}_i(\gamma, \Omega, \underline{p}, \bar{p}, t) &= \bar{f}_i(\bar{\Gamma}, \bar{\Theta}, t) \end{aligned} \quad (4.43)$$

4.4 Identification dans le contexte à erreurs bornées

Afin de choisir la meilleure représentation mathématique décrivant un processus physique, la première étape est la modélisation. Plus tard, des tests seront effectués sur le système réel afin d'observer le comportement de ses sorties et valider ainsi le modèle. Les mesures $y(t_i)$ sont enregistrées sous un certain protocole expérimental. D'autre part, les sorties prévues par le modèle sont notées $y_m(t_i, p)$, où p est le vecteur de paramètres à identifier.

$e(t_i, p)$ représente l'erreur entre les sorties mesurées et celles prédites par le modèle $y_m(t_i, p)$.

$$e(t_i, p) = y(t_i) - y_m(t_i, p) \quad (4.44)$$

Cette erreur correspond à des erreurs structurelles ainsi que des erreurs de mesure. L'estimation dans le contexte d'erreurs bornées consiste à déterminer l'ensemble de vraisemblance \mathbb{S} de vecteurs de paramètres p tels que :

$$\mathbb{X} = \{p \in \mathbb{P} / y_m(t_i, p) \in y(t_i)\} \quad (4.45)$$

qui correspond à un problème d'inversion ensembliste.

4.5 La bibliothèque VNODELP

La bibliothèque VNODELP (A Validated Solver for Initial Value Problems in Ordinary Differential Equations) est une bibliothèque qui a été conçue par N.Nedialkov [5; 72], c'est un solveur écrit en C^{++} et utilisant la programmation littérale (LP)[57] basée sur latex afin de simplifier la compréhension et la programmation. Contrairement aux solveurs des ODE traditionnels qui calculent des solutions approximatives, ce solveur prouve l'existence et l'unicité d'une solution à un problème donné et puis calcul un encadrement garanti contenant cette solution. Le problème se présente comme suit :

considérons un problème de valeur initiale (IVP) :

Soit $y \in \mathbb{R}^n$, $t \in \mathbb{R}$ avec f une fonction différentiable.

$$\frac{\partial y}{\partial t}(t) = f(t, y), \quad y(t_0) = y_0 \quad (4.47)$$

La condition initiale peut être un vecteur d'intervalle $[y_0]$ tel que $y_0 \in [y_0]$. Si $y(t, t_0, y_0)$ représente une solution de 4.47, alors $y(t, t_0, [y_0])$ représente l'ensemble des solutions pour chaque condition initiale contenue dans $[y_0]$ tel que :

$$y(t, t_0, [y_0]) = y(t, t_0, y_0) \mid y_0 \in [y_0] \quad (4.48)$$

L'objectif de l'intégration numérique garantie basées sur l'arithmétique par intervalles, consiste à caractériser, pour $t_{end} > t_0$ à chaque instant d'une séquence temporelle croissante $t_0 < t_1 < \dots < t_f = t_{end}$, toutes les solutions y_i à l'instant t_j telles que :

$$y(t_i, t_0, [y_0]) \subseteq [y_i] \quad \forall i \quad (4.49)$$

les méthodes d'intervalles pour IVP relatives aux équations différentielles ordinaires (EDO) sont généralement basés sur des séries de Taylor, ce qui nécessite donc le calcul des coefficients de Taylor pour $y(t)$ jusqu'à un certain ordre $k \geq 1$.

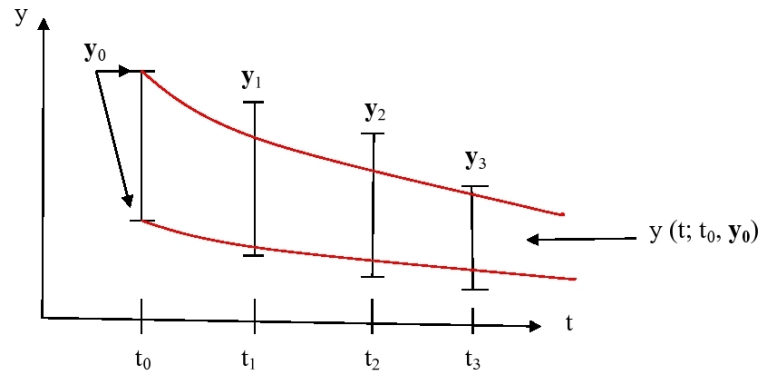


FIGURE 4.5 – Solution d'un problème de valeur initiale (IVP) par VNODELP

Dans un premier temps, VNODELP calcul les bornes a priori \tilde{y}_i telles que $y(t, t_i, y_i) \in \tilde{y}_i \forall t \in [t_i, t_{i+1}]$ et $\forall y_i \in [y_i]$. Ensuite, il détermine les bornes ajustées ou strictes y_{i+1} à l'instant t_{i+1} telle que $y(t_{i+1}, t_0, y_0) \in y_{i+1} \forall y_0 \in [y_0]$

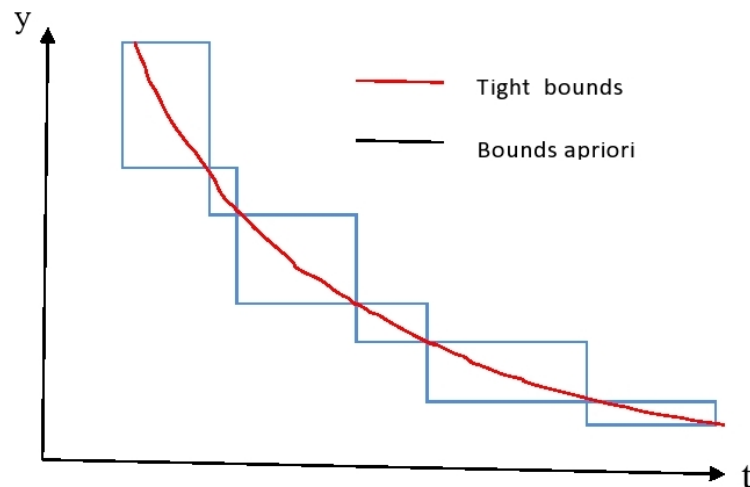


FIGURE 4.6 – Bornes a priori et bornes strictes

4.6 Application au bioréacteur

Dans la pratique, la modélisation de processus biotechnologique est difficile. Les bactéries ainsi que tous les être vivant sont décrits par des fonctions complexes impliquant des paramètres difficiles à définir. Afin de représenter les incertitudes qui affectent ces paramètres, nous nous concentrons sur l'approche par l'intervalle. Pour cette application, le modèle mathématique est utilisé pour décrire le traitement biologique des eaux usées par boues activées. L'azote ammoniacal est traité par réaction de nitrification-dénitrification, impliquant deux populations de bactéries

autotrophes. La nitrification est l'oxydation biologique de l'azote ammoniacal ; il se produit à travers deux étapes : d'abord, nitrification par des bactéries oxydant l'ammonium (AOB) et en second lieu, nitrification par des bactéries de nitrate-oxydant (NOB)

Ainsi, pour représenter l'état de ce bioréacteur, six variables sont nécessaires : les concentrations de bactéries, appelées biomasses et désignées par X_1 et X_2 , ainsi que les concentrations de substrat S_1 , S_2 et S_3 ; S_{in} est la concentration du substrat entrant. Le modèle ci-dessous représente l'évolution dynamique de ces variables d'état [30] .

$$\begin{aligned}
 \dot{S}_1 &= D(S_{in} - S_1) - k_1\mu_1(S_1)X_1 \\
 \dot{X}_1 &= (\mu_1(S_1) - D)X_1 \\
 \dot{S}_2 &= k_1\mu_1(S_1)X_1 - k_2\mu_2(S_2)X_2 - S_2D \\
 \dot{X}_2 &= (\mu_2(S_2) - D)X_2 \\
 \dot{S}_3 &= k_2\mu_2(S_2)X_2 - S_3D
 \end{aligned} \tag{4.50}$$

avec $\mu_1(S_1) = \mu_{max1} \frac{S_1}{(S_1 + k_{s1})}$ et,

$$\mu_2(S_2) = \mu_{max2} \frac{S_2}{(S_2 + k_{s2})}$$

$\mu_1(S_1)$ et $\mu_2(S_2)$ représentent les taux de croissance de la biomasse. Ils sont modélisés par une cinétique de Monod [67]. $\mu_{max1}(S_1)$ et $\mu_{max2}(S_2)$ sont les taux de croissance spécifiques maximums, k_1 et k_2 sont les coefficients stœchiométriques qui représentent les rendements respectifs de biomasses X_1 et X_2 , k_{s1} et k_{s2} sont les constantes de demi-saturation pour la croissance correspondant à l'affinité des cellules pour le substrat de chaque population bactérienne. D est le taux de dilution, tel que $D = \frac{Q_{in}}{V}$, où Q_{in} est le débit d'entrée du bioréacteur et V son volume.

4.6.1 Étude de la monotonie

L'application de la théorie de comparaison des inégalités différentielles à un système donné, nécessite la vérification de la propriété de monotonie afin d'appliquer la règle 4.1. Pour cela, on vérifie dans un premier temps si le système est coopératif, ce qui revient à vérifier que les éléments de la matrice formée par les dérivées partielles

du système 4.47, hors diagonale, sont tous positifs.

le calcul a montré qu'il existait des dérivées partielles négatives, donc le système n'est pas coopératif. Pour vérifier si notre système est monotone, reste à construire le graphe de préservation d'ordre.

4.6.2 Préservation d'ordre

Les dérivées partielles déterminées auparavant, nous permettent de tracer le graphe suivant :

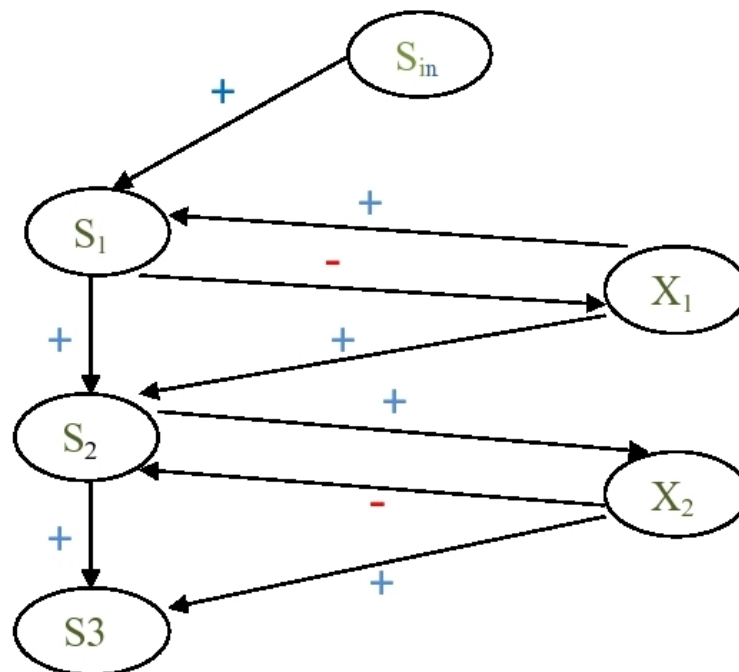


FIGURE 4.7 – Graphe de préservation d'ordre

Dans ce graphe, il existe des cycles négatifs :

1. cycle $S_1, X_1, S_2, S_1 \rightarrow \text{sign}(e_{X_1}^{S_1})\text{sign}(e_{S_2}^{X_1})\text{sign}(e_{S_1}^{S_2}) < 0$
2. cycle $S_2, X_2, S_3, S_2 \rightarrow \text{sign}(e_{X_2}^{S_2})\text{sign}(e_{S_3}^{X_2})\text{sign}(e_{S_2}^{S_3}) < 0$

Ce qui nous permet de conclure que notre système n'est ni coopératif, ni préservateur d'ordre, donc il n'est pas monotone. Afin de construire des trajectoires

englobantes, il nous est nécessaire d'appliquer le théorème de Müller.

Le but est de pouvoir construire des équations différentielles dont la solution permet d'obtenir des trajectoires maximales et minimales qui contiennent les solutions de notre système d'une manière garantie.

4.6.3 Équation des trajectoires minimales et maximales

L'application du théorème de Müller nous permet d'obtenir les équations minimale et maximales suivantes :

$$\begin{aligned}
 \overline{\dot{S}}_1 &= D(S_{in} - \overline{S}_1) - \underline{k}_1 \overline{\mu}_{1s1} \overline{X}_1 \\
 \overline{\dot{X}}_1 &= (\overline{\mu}_{1x1} - D) \overline{X}_1 \\
 \overline{\dot{S}}_2 &= \overline{k}_1 \overline{\mu}_{1s2} \overline{X}_1 - \overline{k}_2 \overline{\mu}_{2s2} \overline{X}_2 - \overline{S}_2 D \\
 \overline{\dot{X}}_2 &= (\overline{\mu}_{2x2} - D) \overline{X}_2 \\
 \overline{\dot{S}}_3 &= \overline{k}_2 \overline{\mu}_{2s3} \overline{X}_2 - \overline{S}_3 D
 \end{aligned} \tag{4.51}$$

$$\begin{aligned}
 \underline{\dot{S}}_1 &= D(S_{in} - \underline{S}_1) - \overline{k}_1 \underline{\mu}_{1s1} \underline{X}_1 \\
 \underline{\dot{X}}_1 &= (\underline{\mu}_{1x1} - D) \underline{X}_1 \\
 \underline{\dot{S}}_2 &= \underline{k}_1 \underline{\mu}_{1s2} \underline{X}_1 - \overline{k}_{2s2} \underline{\mu}_{2s2} \underline{X}_2 - \underline{S}_2 D \\
 \underline{\dot{X}}_2 &= (\underline{\mu}_{2x2} - D) \underline{X}_2 \\
 \underline{\dot{S}}_3 &= \underline{k}_2 \underline{\mu}_{2s3} \underline{X}_2 - \underline{S}_3 D
 \end{aligned}$$

4.7 Résultats

Dans un premier temps, nous avons simulé le modèle de bioréacteur avec des paramètres inspirés de la littérature [30]. Les mesures enregistrées à partir du chimostat ont été réalisées sur une période de 220 jours. La figure 4.8 représente les simulations du modèle du bioréacteur et celles des données prélevées. La figure 4.9 représente les simulations du modèle obtenu par le bibliothèque vnodelp

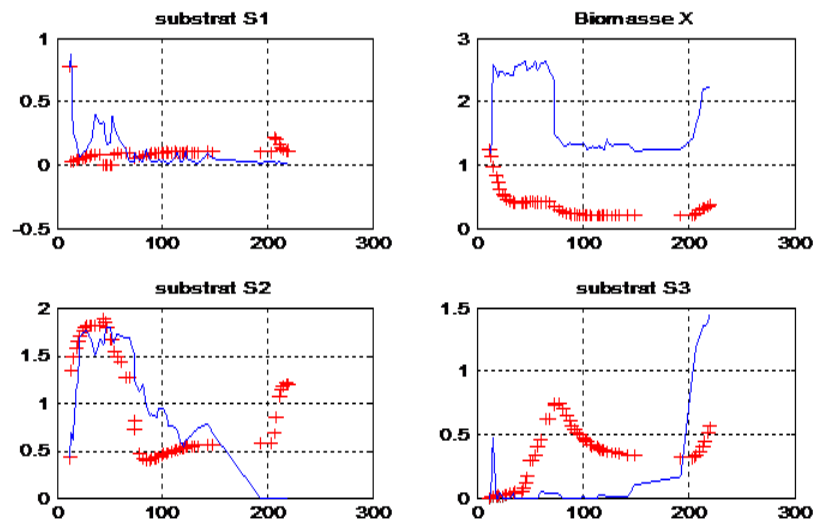
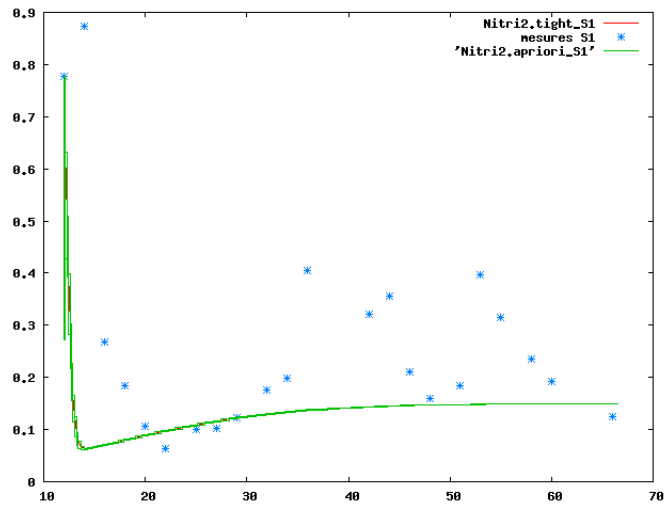
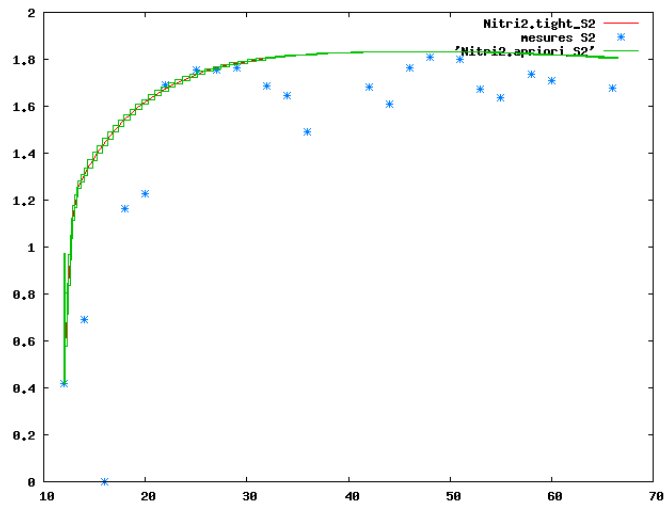


FIGURE 4.8 – Simulations des mesures enregistrées (*bleu*) et du modèle de nitrification(+)

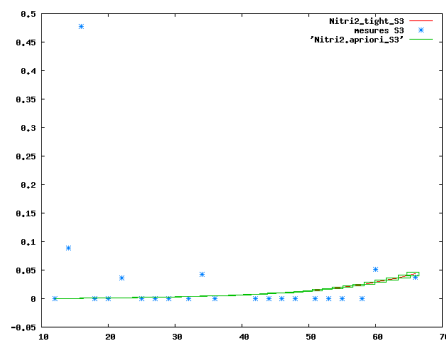
L'identification des paramètres du modèle du bioréacteur a été effectuée par l'application du critère des moindres carrés, en utilisant la fonction matlab « *lsqnonlin* », figure 4.10. Cependant, les paramètres à identifier sélectionnés pour l'identification ensembliste sont les taux de croissance maximaux $\mu_{max1}(S_1)$ et $\mu_{max2}(S_2)$ pour les deux populations bactériennes X_1 et X_2 , respectivement, ainsi que les rendements k_1 et k_2 . Les constantes de demi-saturation k_{s1} et k_{s2} sont considérées comme constantes.



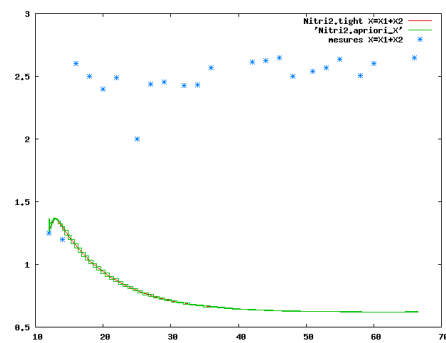
(a)



(b)



(c)



(d)

FIGURE 4.9 – Simulation du modèle avec VNODELP (a,b,c les substrats S1,S2,S3 resp, d la biomasse $X = X1 + X2$)

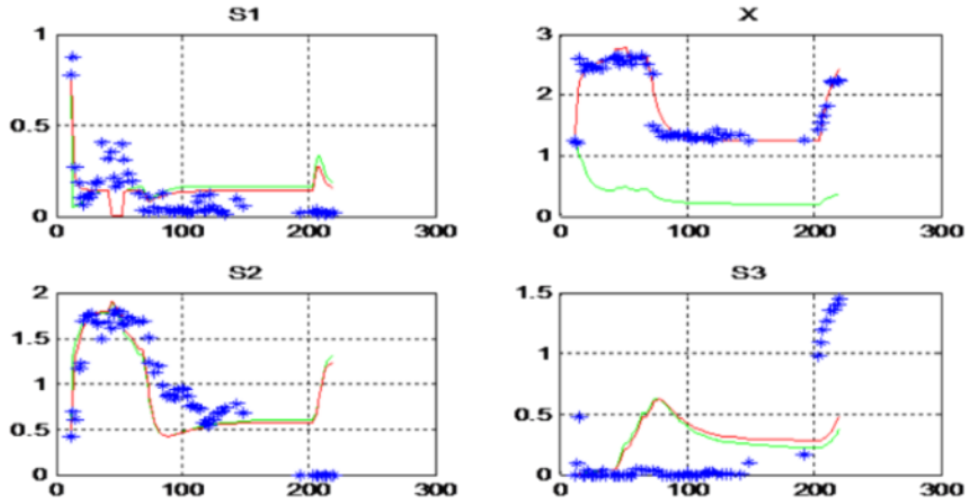


FIGURE 4.10 – Identification des paramètres par moindres carrés
 Mesures prélevées : courbe bleue étoilée
 Modèle simulé : courbe verte
 Modèle obtenu pour les valeurs identifiées par MC : courbe rouge

Les trajectoires englobantes sont obtenues par résolution des équations (4.51). La figure 4.11 représente la simulation du modèle pour la période de 2 jours à 22 jours et pour laquelle le substrat d'entrée est constant. La figure 4.12 concerne une période de 18 jours à 27 jours, pour ce cas il y'a une variation du substrat S_{in} . Les simulations sont effectuées avec les valeurs des paramètres obtenues par la méthode des moindres carrés. D'autre part, la biomasse $X = X_1 + X_2$ et substrats S_1 , S_2 et S_3 sont accessibles à la mesure à des instants discrets. On considère que ces mesures sont affectées par des erreurs additives inconnues, mais qui restent bornées par des bornes connues a priori.

$$\mathbb{Y}(t_j) = Y(t_j) + \mathbb{E}(t_j) \quad (4.52)$$

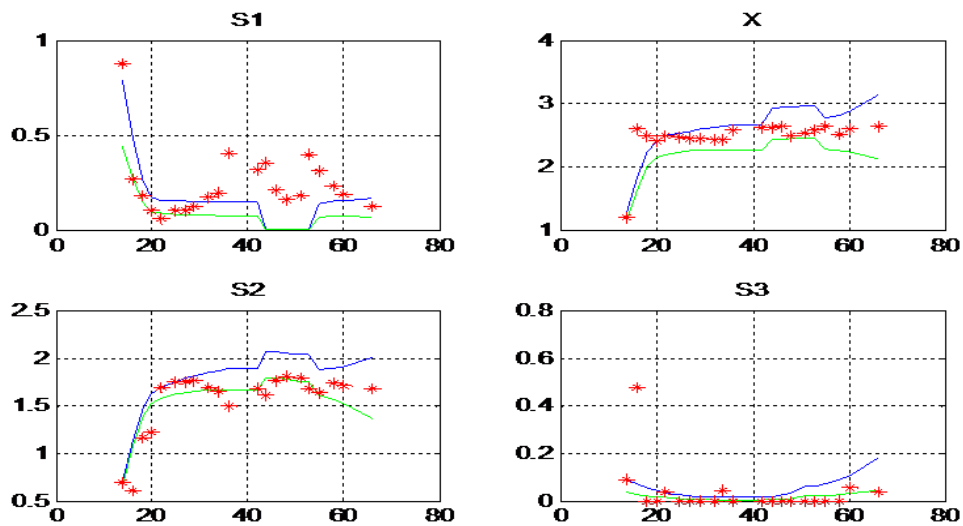
avec $\mathbb{E}(t_j)$ représente le domaine admissible borné de toutes les erreurs commises sur les mesures à l'instant t_j

$$\mathbb{X} = [1.01, 0.98] X(t_j)$$

$$\mathbb{S}_1 = [1.01, 0.9] S_1(t_j)$$

$$\mathbb{S}_2 = [1.01, 0.99] S_2(t_j)$$

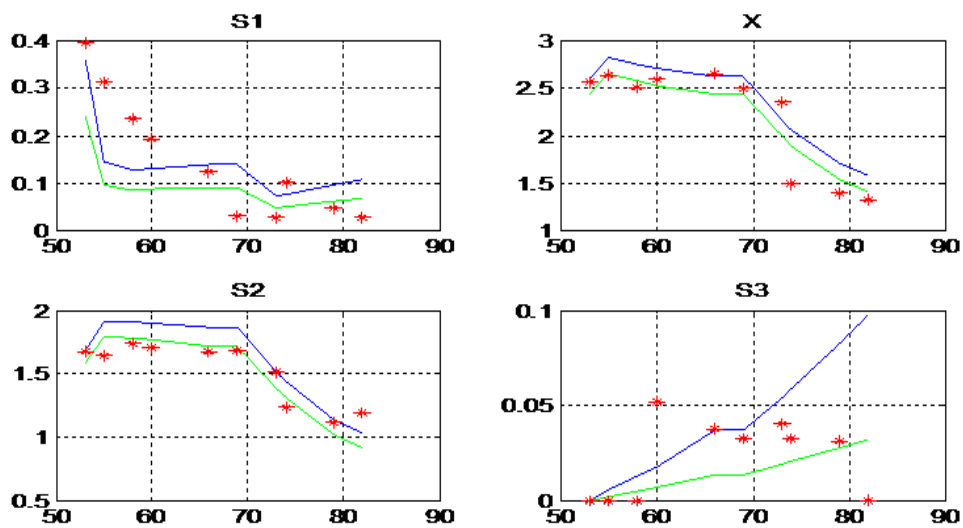
$$\mathbb{S}_3 = [1.01, 0.3] S_3(t_j) \text{ alors :}$$

FIGURE 4.11 – Trajectoires englobantes pour une période $2j$ à $22j$

En bleu : courbes max

En vert : courbe min

En rouge : les mesures réelles

FIGURE 4.12 – Trajectoires englobantes pour une période $18j$ à $27j$

En bleu : courbes max

En vert : courbe min

En rouge : les mesures réelles

Dans l'étape suivante, une procédure de l'identification des paramètres utilisant la méthode des moindres carrés via la fonction Matlab « lsqnonlin » programmé sous

Intlab a été utilisé pour trouver des courbes inclusives optimales figure 4.13, ce qui a permis de déduire un encadrement optimal pour chaque paramètre à identifier et qu'on a regroupé dans le tableau 4.1.

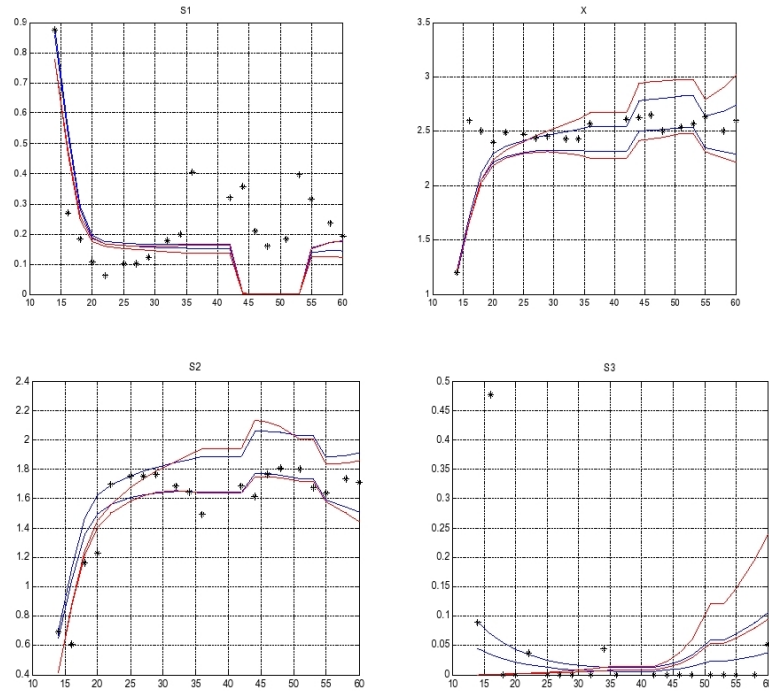


FIGURE 4.13 – Trajectoires englobantes(bleue), trajectoires englobantes optimales(rouge), mesures(*)

L'identification des paramètres par inversion ensembliste est effectuée en exécutant l'algorithme SIVIA qu'on a associé à l'intégration numérique basée sur la théorie de comparaison des inégalités différentielles. Cela a permis dans un premier temps, de caractériser pour les paramètres μ_{max1} et μ_{max2} , d'une manière garantie, l'ensemble \mathbb{S} illustré par la figure 4.14. En effet, l'approximation intérieure $\underline{\mathbb{S}}$ est décrite par l'ensemble des boîtes bleues, obtenus exactement et avec une précision ϵ . Les boîtes jaunes sont incertaines et sont traitées une seconde fois avec Sivia jusqu'à ce qu'on atteigne la précision fixée. Cependant, en augmentant la précision ϵ , le temps de simulation devient très grand jusqu'à atteindre parfois plusieurs jours, il est donc difficile d'affiner les résultats en augmentant la précision avec Sivia. Pour cette raison, nous avons associé un contracteur à l'algorithme Sivia. L'algorithme réalisé s'appelle SiviaP et il est illustré par la figure 4.15, la figure 4.16 et la figure 4.17 pour différentes précisions. SiviaP est un contracteur basé sur la projection de contraintes, utilisant des sous solveurs comme il a été décrit au chapitre I. Cet

l'algorithme a permis de réduire le temps de simulation ainsi que la longueur des intervalles relative à chaque paramètre. La projection de l'approximation extérieure $\underline{\mathbb{S}}$ sur les axes de paramètres permet d'obtenir les encadrements des paramètres de manière garantie. L'identification des paramètres μ_{max1} et μ_{max2} est effectuée en maintenant les rendements k_1 et k_2 constants et cela pour un pavé de recherche initial $p = [0.3, 0.8] \times [0.2, 0.4]$ et une précision $\epsilon = 0.01$

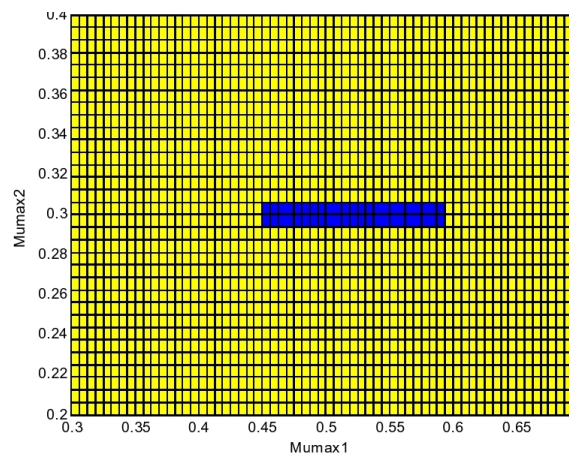


FIGURE 4.14 – Pavés solutions relatifs aux paramètres μ_{max1} et μ_{max2} obtenu par Sivia

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 84.502531s \quad \epsilon = 0.01$$

$$\mu_{max1}(S_1) = [0.4499, 0.5938]$$

$$\mu_{max2}(S_1) = [0.2937, 0.3063]$$

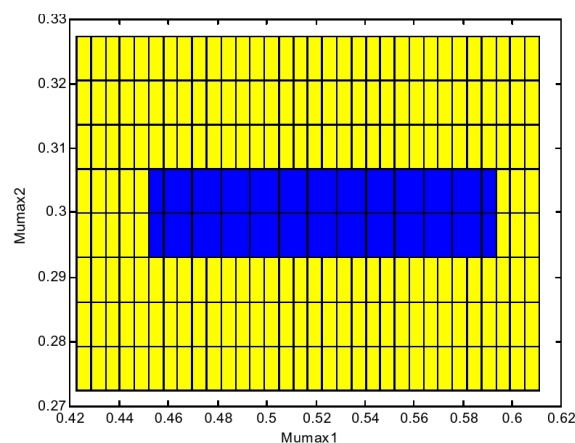


FIGURE 4.15 – Pavés solutions relatifs aux paramètres μ_{max1} et μ_{max2} obtenu par SiviaP

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 9.9522s \quad \epsilon = 0.01$$

$$\mu_{max1}(S_1) = [0.4520, 0.5933]$$

$$\mu_{max2}(S_1) = [0.2929, 0.3068]$$

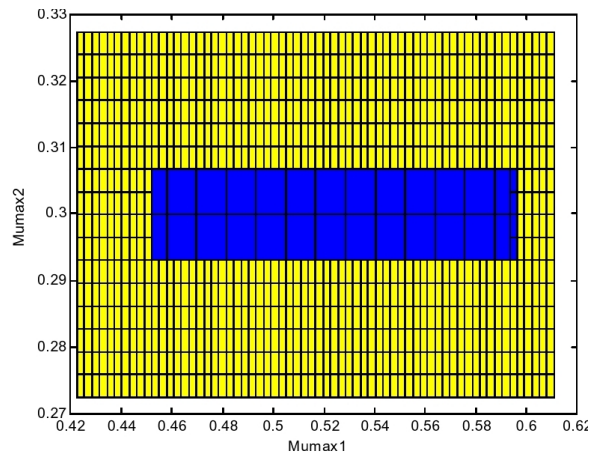


FIGURE 4.16 – Pavés solutions relatifs aux paramètres μ_{max1} et μ_{max2} obtenu par SiviaP

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 35.2934s \quad \epsilon = 0.005$$

$$\mu_{max1}(S_1) = [0.4520, 0.5963]$$

$$\mu_{max2}(S_1) = [0.2929, 0.3068]$$

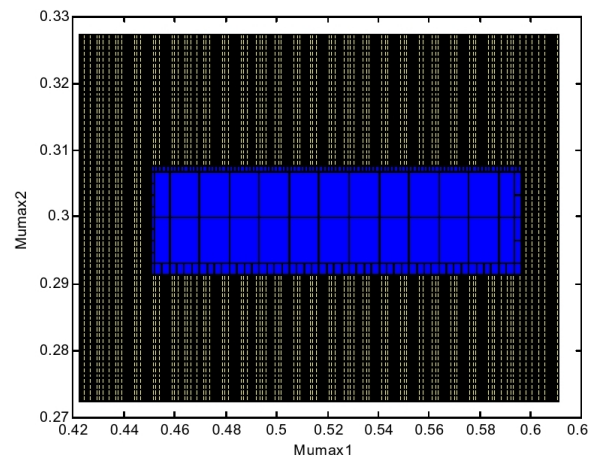


FIGURE 4.17 – Pavés solutions relatifs aux paramètres μ_{max1} et μ_{max2} obtenu par SiviaP

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 641.6982s \quad \epsilon = 0.001$$

$$\mu_{max1}(S_1) = [0.4505, 0.5970]$$

$$\mu_{max2}(S_1) = [0.2912, 0.3076]$$

Les figures suivantes illustrent les encadrements des paramètres k_1 et k_2 maintenant les paramètres μ_{max1} et μ_{max2} constants. La figure 4.18 est obtenue par Sivia pour une précision $\epsilon = 0.01$, On constate que le temps est très élevé mais on a pu le réduire par SiviaP, les figures 4.18, 4.19 et la figure 4.20 sont obtenues pour différentes précisions. Le pavé initial est : $p = [0.5, 1] \times [0.8, 2]$

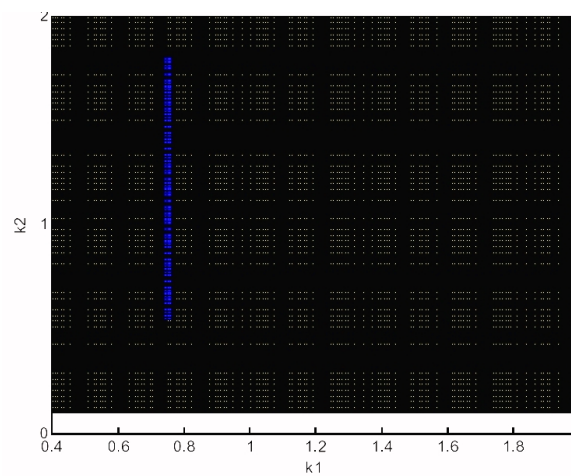


FIGURE 4.18 – Pavés solutions relatifs aux paramètres k_1 et k_2 obtenu par Sivia

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 13868.2185s \quad \epsilon = 0.01$$

$$k_1 = [0.7312, 0.7688]$$

$$k_2 = [0.5453, 1.8293]$$

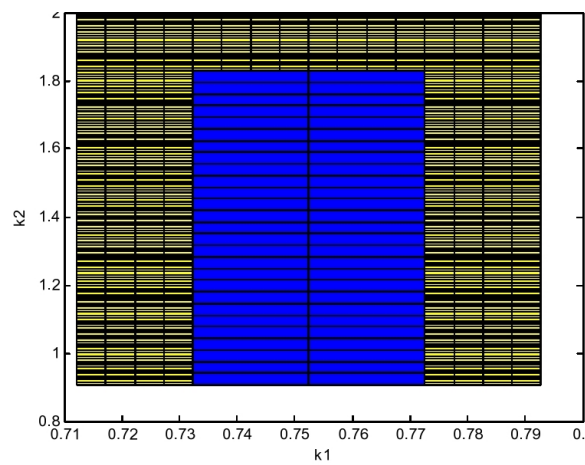


FIGURE 4.19 – Pavés solutions relatifs aux paramètres k_1 et k_2 obtenu par SiviaP

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 111.7257s \quad \epsilon = 0.01$$

$$k_1 = [0.7323, 0.7726]$$

$$k_2 = [0.9067, 1.8292]$$

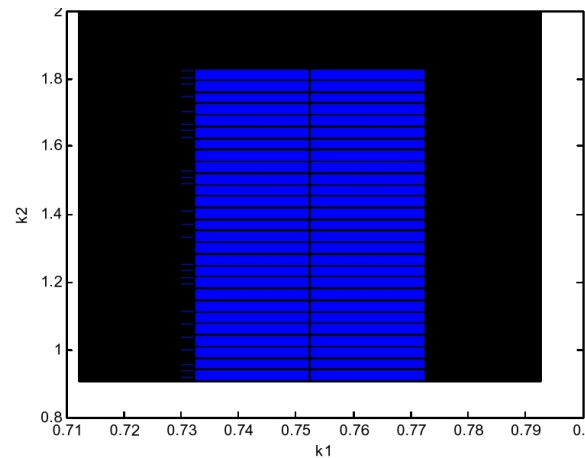


FIGURE 4.20 – Pavés solutions relatifs aux paramètres k_1 et k_2 obtenu par SiviaP

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 7810.9073s \quad \epsilon = 0.001$$

$$k_1 = [0.7291, 0.7726]$$

$$k_2 = [0.9067, 1.8335]$$

Les figures suivantes illustrent les encadrements des paramètres k_{s1} et k_{s2} pour une précision $\epsilon = 0.01$ et pour un encadrement initial $p = [0.2, 1] \times [0.5, 1]$

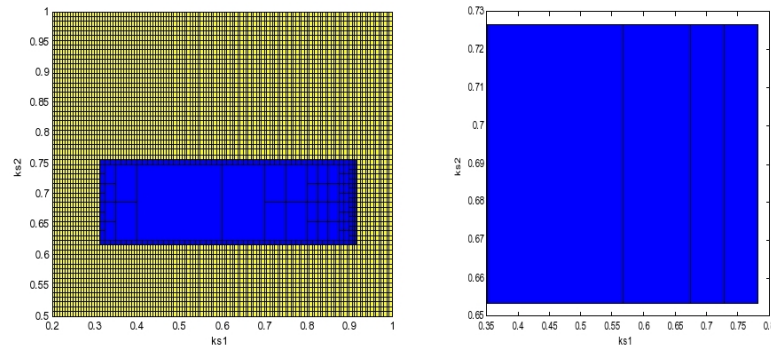


FIGURE 4.21 – Pavés solutions relatifs aux paramètres k_{s1} et k_{s2} obtenu par Sivia et SiviaP

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 9225.609117s \quad k_{s1} = [0.2437, 0.9188] \quad k_{s2} = [0.6171, 0.7579]$$

$$T = 5.084959s \quad k_{s1} = [0.3516, 0.7819] \quad k_{s2} = [0.6533, 0.7267]$$

La figure 4.22 illustre l'identification des quatre paramètres μ_{max1} , μ_{max2} , k_1 et k_2 utilisant SiviaP pour une précision $\epsilon = 0.01$, et pour une boîte de recherche initiale :

$$p = [0.2, 1.2] \times [0.2, 1] \times [0.1, 0.8] \times [0.5, 0.8]$$

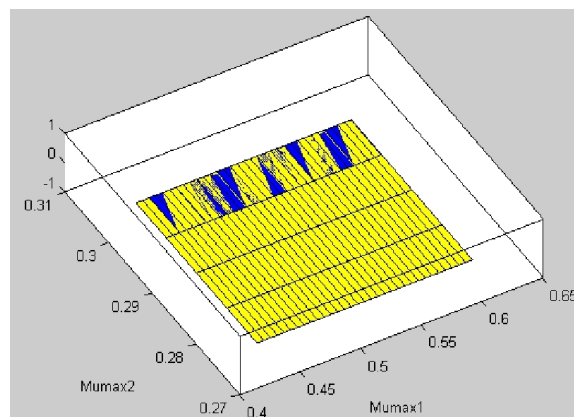


FIGURE 4.22 – Pavés solutions relatifs aux quatre paramètres μ_{max1} , μ_{max2} , k_1 et k_2 obtenu par SiviaP

Les pavés concernant les paramètres identifiés obtenus sont :

$$T = 39530.0036s \quad \epsilon = 0.01$$

$$\mu_{max1}(S_1) = [0.4613, 0.5612]$$

$$\mu_{max1}(S_1) = [0.2930, 0.3068]$$

$$k_1 = [0.7474, 0.7575]$$

$$k_2 = [0.9067, 1.8292]$$

Afin de procéder à une comparaison, nous avons regroupé les résultats dans le tableau ci-dessous.

Paramètres	Paramètres a priori	Paramètres estimés par (MC)	Trajectoires englobantes MC	SiviaP
μ_{max1}	[0.45, 0.5]	0.5049	[0.5081, 0.5463]	[0.4613, 0.5612]
μ_{max2}	0.27	0.2994	[0.2624, 0.3199]	[0.2930, 0.3068]
k_1	5.555	0.751	[0.7398, 0.7471]	[0.7474, 0.7575]
k_2	10	1.1875	[1.6704, 2.0000]	[0.9067, 1.8292]
k_{s1}	[0.3, 0.48]	0.47	[0.4360, 0.4845]	[0.47, 0.47]
k_{s2}	[0.65, 0.7]	0.69	[0.4848, 0.6452]	[0.69, 0.69]
ϵ	–	–	–	0.01
$T(s)$	–	963.4441	41.2212	39530.0036

Tableau 4.1 – Comparaison des résultats obtenus par trois méthodes

4.8 Conclusion

Cette étude nous a permis de déterminer un encadrement garanti des solutions des équations d'état d'un modèle de bioréacteur, par le biais de la résolution des équations différentielles numériques par application de la théorie de comparaison des inégalités différentielles, en utilisant la boîte à outils INTLAB de MATLAB. L'identification des paramètres de ce modèle sont obtenus par l'utilisation du critère des moindres carrés.

L'optimisation des trajectoires englobantes par le principe des moindres carrés en utilisant la fonction lsqnonlin de Matlab, nous a permis de déterminer les intervalles d'incertitude minimaux relatifs à chaque paramètre identifié.

L'algorithme Sivia a également été programmé en utilisant la bibliothèque INTLAB dans Matlab, afin de tester l'inversion ensembliste. Cette étude comparative nous a

permis de constater que les intervalles obtenus par l'inversion ensembliste sont assez larges. Ceci est dû au problème du pessimisme provoqué par les deux phénomènes décrit au chapitre II : Le premier est le problème de multioccurrence et le second est l'effet d'enveloppement, Cela rend les méthodes ensembliste plus pauvres que les méthode probabilistes. L'autre problème concernant l'inversion ensembliste, c'est temps de simulation qui est plus long jusqu'au atteindre parfois jusqu'à plusieurs jours. Cependant, l'introduction d'un contracteur associé à l'algorithme Sivia nous a permis de réduire le temps de simulation et la longueur de l'intervalle autour de chaque paramètre identifié. Nous pouvons donc conclure que les méthodes ensemblistes nous permettent de déterminer les solutions de manière garantie.

Deuxième partie

Le méthode LSCR(Leave-out Sign-dominant Correlation Regions)

4.9 Introduction

La méthode d'identification des paramètres du bioréacteur que nous allons exposer dans ce chapitre est la méthode LSCR (Leave out Sign-dominant Correlation Regions) développé par MC.CAMPI [20; 21]. Le principe est basé sur la construction d'une région de confiance Θ , à partir du calcul des fonctions de corrélation empiriques. L'un des aspects intéressant de cette méthode est que ces régions de confiance sont déterminées de manière non asymptotiques et avec une probabilité garantie, quelle que soit la taille de l'ensemble. D'autre part, les hypothèses sur le bruit qui affectent les données sont très réduites, nous pouvons simplement supposer que c'est un signal indépendant et réparti symétriquement autour de zéro, la variance peut prendre n'importe quelle valeur. Cependant, l'intensité du bruit peut influencer sur la largeur de la zone de confiance.

4.10 Principe de la méthode

La procédure de calcul des régions de confiance par la méthode LSCR est la suivante :

1. On suppose que nous essayons d'identifier les paramètres p d'un système S , nous définissons l'erreur de prédiction $\epsilon(p)$, qui représente la différence entre les données échantillonnées bruitées sur la sortie du système $y(t)$ et la sortie du modèle $y_m(t)$.

$$\epsilon_t(p) = y_t - y_m(p), \quad \text{pour } t = 1, 2, \dots, K \quad (4.53)$$

2. Sélectionner un entier $r \geq 1$. Puis, pour $t = 1 + r, \dots, N + r = K$, Calculer :

$$f_{t-r,r}^\epsilon = \epsilon_{t-r}(p) \epsilon_t(p) \quad (4.54)$$

3. Considérer un ensemble $I = \{1, \dots, M\}$, pour un certain nombre de données M , et une collection G de sous ensembles $I_i \subseteq I, i = 1, \dots, M$ qui est un groupe respectant la différence symétrique. ie $(I_i \cup I_j) - (I_i \cap I_j) \in G$, si $I_i, I_j \in G$ et alors calculer les estimations de la corrélation $E[\epsilon_{t-r}(p) \epsilon_t(p)]$, définit par :

$$g_{i,r}^\epsilon(p) = \sum_{k \in I_i} f_{k,r}^\epsilon(p) \quad i = 1, \dots, M \quad (4.55)$$

Pour $p = p_0$, sachant que w représente un bruit additif indépendant et symétriquement distribué autour de zéro. On a

$$E[\epsilon_{t-r}(p_0) \epsilon_t(p_0)] = E[\omega_{t-1} \omega_t] = 0 \quad (4.56)$$

On en conclut que, les estimations empiriques sont une séquence de variables aléatoires de moyenne nulle pour $p = p_0$. Par conséquent, nous calculons un certain nombre d'estimation de corrélation en utilisant différents sous-ensembles de données, et nous éliminons les régions dans l'espace des paramètres, où les estimations empiriques sont positifs (ou négatifs) trop de fois.

4. Soit un entier $q \geq 0$ dans un intervalle $\left[1, \frac{M+1}{2}\right]$, il s'agit de caractériser la région $\Theta_{r,q}^\epsilon$, de sorte qu'au moins q de $g_{i,r}^\epsilon(p)$ sont supérieurs à zéro et au moins q sont inférieures à zéro. La probabilité pour que p appartienne à $\Theta_{i,r}^\epsilon(p)$ est :

$$P(p \in \Theta_{r,q}^\epsilon) = 1 - \frac{2q}{M}$$

La région $\Theta_{r,q}^\epsilon$ dépend des valeurs r et q , ainsi que de l'ensemble des intervalles qui forment le groupe G . En annexe 2, on présente un bref descriptif du procédé de génération de ce groupe décrit par Gordon [34].

On peut définir $\Theta_{r,q}^\epsilon$ formellement par $:\Theta_{r,q}^\epsilon = \Theta_{r,q}^{\epsilon,+} \cap \Theta_{r,q}^{\epsilon,-}$ tel que :

$$\Theta_{r,q}^{\epsilon,+} = \left\{ p \in \mathbb{P} \text{ tel que } \sum_{i=1}^m \tau_i^{\epsilon,+}(p) \geq q \right\}, j = 1, 2, \dots$$

$$\Theta_{r,q}^{\epsilon,-} = \left\{ p \in \mathbb{P} \text{ tel que } \sum_{i=1}^m \tau_i^{\epsilon,-}(p) \geq q \right\}, j = 1, 2, \dots \quad (4.57)$$

\mathbb{P} représente le pavé de recherche initial pour le paramètre p avec :

et

$$\tau_i^{\epsilon,-} = \begin{cases} 1 & \text{si } -g_{i,r}^\epsilon(p) \geq 0 \\ 0 & \text{ailleurs} \end{cases} \quad (4.58)$$

$$\tau_i^{\epsilon,+} = \begin{cases} 1 & \text{si } g_{i,r}^\epsilon(p) \geq 0 \\ 0 & \text{ailleurs} \end{cases}$$

L'ensemble $\Theta_{r,q}^{\epsilon,+}$ contient toutes les valeurs $p \in \mathbb{P}$ de telle sorte qu'au moins une des q fonctions $g_{i,r}^\epsilon(p)$ est plus grande que zéro tandis que $\Theta_{r,q}^{\epsilon,-}$ contient toutes les valeurs $p \in \mathbb{P}$ de telle sorte qu'au moins une des q fonctions $g_{i,r}^\epsilon(p)$ est plus petite que zéro.

Considérons le modèle linéaire suivant [20] :

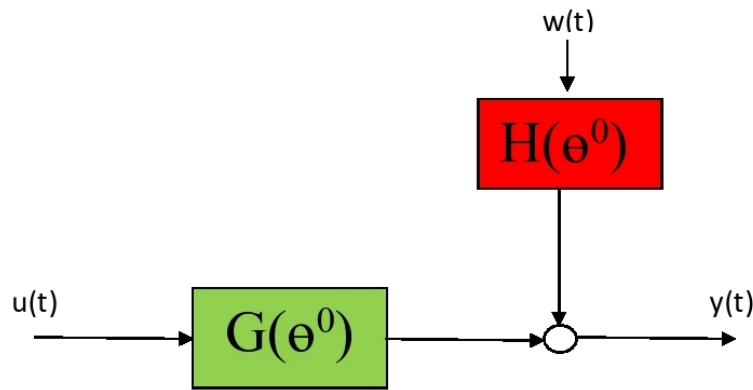


FIGURE 4.23 – Modèle linéaire

Theorem 5. *Soit :*

$$y_t = G(z^{-1}, \theta)u_t + H(z^{-1}, \theta)w_t \quad (4.59)$$

On suppose que w_t est un bruit dont la distribution est indépendante et symétriquement distribué autour de zéro. On suppose aussi que w_t et $w_t \cdot u_t$ admettent des densités pour éviter une probabilité non nulle pour que ces deux quantités prennent des valeurs égales. Le calcul des régions de confiance $\Theta_{i,r}^\epsilon(p)$ et $\Theta_{i,r}^u(p)$ étant basées sur le calcul de l'auto-corrélation de l'erreur de prédiction et l'inter-corrélation entre l'erreur de prédiction et le signal d'entrée. Alors on montre qu'un paramètre appartenant à ces régions de confiance vérifie : (Démonstration en Annexe 3) :

$$Pr[p \in \Theta_{i,r}^\epsilon(p)] = 1 - \frac{2q}{M}$$

(4.60)

$$Pr[p \in \Theta_{i,r}^u(p)] = 1 - \frac{2q}{M}$$

Les deux ensembles $\Theta_{i,r}^\epsilon$ et $\Theta_{i,r}^u$ fournissent des régions de confiance non asymptotiques. Cependant, chacun de ces ensembles seront généralement sans bornes dans certaines directions de l'espace des paramètres et donc inexploitable. Dans la pratique la région confiance peut être obtenue en faisant l'intersection d'un certain nombre n_ϵ de $\Theta_{i,r}^\epsilon$ et n_u de $\Theta_{i,r}^u$ tel que :

$$\Theta = \bigcap_{r=1}^{n_\epsilon} \Theta_r^\epsilon \bigcap_{s=1}^{n_u} \Theta_s^u \quad (4.61)$$

Le choix de n_ϵ et n_u dépend de la classe du modèle étudié ainsi qu'au nombre de données disponibles[21]. Le cas de système non-linéaire est traité dans [21]

4.11 Exemple de Campi

Pour mieux comprendre la méthode, Campi [6] décrit en détails l'exemple suivant :

Exemple 4.3

Considérons un système du premier ordre :

$$y_t + \theta_0 y_{t-1} = w_t \quad (4.62)$$

avec $\theta_0 = 0.2$ le paramètre à identifier et w_t est un processus indépendant symétriquement distribué autour de zéro, sa variance peut être choisie quelconque.

On génère 9 mesures à partir du modèle (4.62) qu'on représente sur la figure 4.24¹ et le but est d'identifier le paramètre θ_0 et de déterminer une région de confiance autour de ce paramètre.

1. <http://marco-campi.unibs.it/LSCRwebsite/index.html>

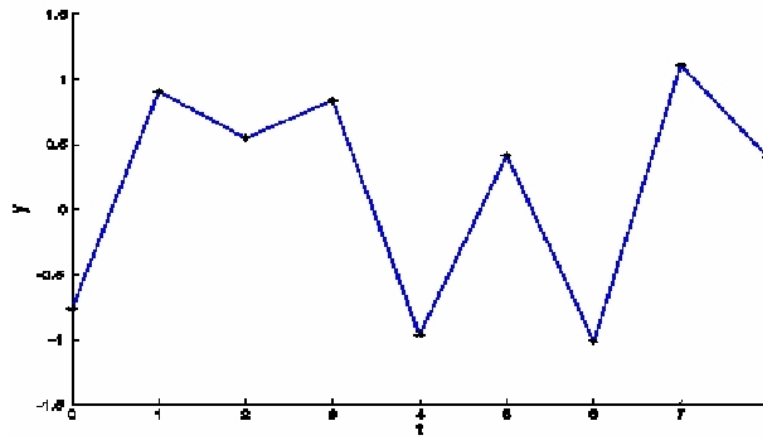


FIGURE 4.24 – Courbe des mesures

On réécrit le système avec un paramètre générique θ afin de définir un prédicteur :

$$\hat{y}_t = -\theta y_{t-1} \quad (4.63)$$

et par la suite calculer l'erreur de prédiction pour $t = 1, \dots, 8$:

$$\epsilon_t(\theta) = y(t) - \hat{y}(t) = y(t) + \theta y_{t-1} \quad (4.64)$$

ces erreurs nous permettent de calculer les fonctions suivantes :

$$f_{t-1}(\theta) = \epsilon_{t-1}(\theta)\epsilon_t(\theta) \quad \text{pour } t = 2, \dots, 8 \quad (4.65)$$

Puis nous calculons 8 moyennes de ces fonctions :

$$g_i(\theta) = \frac{1}{4} \sum_{k \in I_n} f_k(\theta) \quad (4.66)$$

ou les ensembles I_n sont des sous ensembles de $\{1, \dots, 7\}$ contenant les éléments mis en évidence dans la figure 4.25² par un rond. Par exemple, $I_1 = \{1, 2, 4, 5\}$ et $I_2 = \{1, 3, 4, 6\}$. Les fonctions $g_i(\theta)$, $i = 1, \dots, 7$, peuvent être interprétées comme fonction de corrélation empiriques de l'erreur de prédiction.

2. <http://marco-campi.unibs.it/LSCRwebsite/index.html>

	1	2	3	4	5	6	7
I_1	•	•		•	•		
I_2	•		•	•		•	
I_3		•	•		•	•	
I_4	•	•				•	•
I_5	•		•		•		•
I_6		•	•	•			•
I_7				•	•	•	•

FIGURE 4.25 – Tableau des sous ensembles I_n

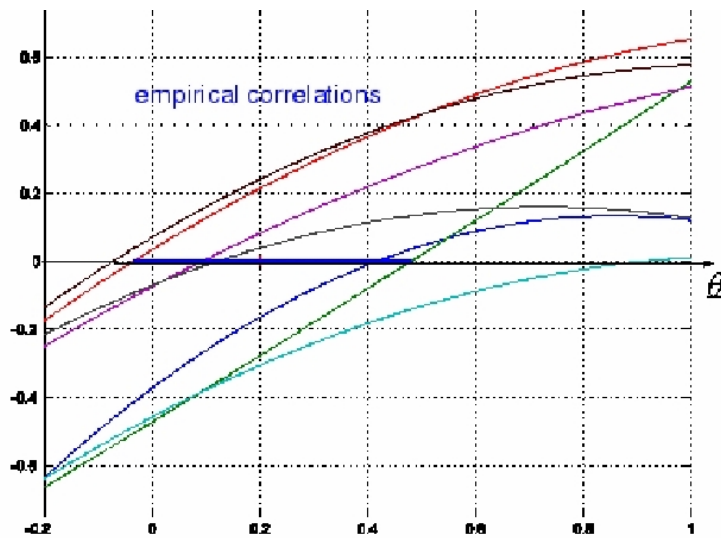
Pour $r=1$, les fonctions de corrélation relatives à ce tableau se calculent de la façon suivante :

$$f_{1,r=1}^\epsilon = \epsilon_1(\theta) \epsilon_2(\theta) + \epsilon_2(\theta) \epsilon_3(\theta) + \epsilon_4(\theta) \epsilon_5(\theta) + \epsilon_5(\theta) \epsilon_6(\theta)$$

$$f_{2,r=1}^\epsilon = \epsilon_1(\theta) \epsilon_2(\theta) + \epsilon_3(\theta) \epsilon_4(\theta) + \epsilon_4(\theta) \epsilon_5(\theta) + \epsilon_6(\theta) \epsilon_7(\theta)$$

$$f_{3,r=1}^\epsilon = \dots$$

La figure 4.25³ représente $g_i(\theta)$, la moyenne de ces fonctions de corrélation en fonction de θ

FIGURE 4.26 – Fonctions de corrélations $g_i(\theta)$

3. <http://marco-campi.unibs.it/LSCRwebsite/index.html>

Remarque

Les fonctions de corrélation $g_i(\theta)$ ont tendance à couper l'axe des abscisses à des points proches du paramètre recherché θ_0 et elles prennent des valeurs positives et négatives avec une probabilité égale. En effet, si on calcule la fonction $g_1(\theta)$ pour $\theta = \theta_0$ comme suit :

$$\begin{aligned}
g_1(\theta_0) &= \frac{1}{4} \sum_{k \in \{1,2,4,5\}} [y_k + \theta_0 y_{k-1}] [y_{k+1} + \theta_0 y_k] & (4.67) \\
&= \frac{1}{4} \sum_{k \in \{1,2,4,5\}} [y_k + \theta_0 y_{k-1} + (\theta - \theta_0) y_{k-1}] [y_{k+1} + \theta_0 y_k + (\theta - \theta_0) y_k] \\
&= \frac{1}{4} \sum_{k \in \{1,2,4,5\}} [w_k + (\theta - \theta_0) y_{k-1}] [w_{k+1} + (\theta - \theta_0) y_k] \\
&= \frac{1}{4} (\theta - \theta_0)^2 \sum_{k \in \{1,2,4,5\}} y_{k-1} y_k \\
&\quad + \left(\frac{1}{4}\right) (\theta - \theta_0) \sum_{k \in \{1,2,4,5\}} w_k y_k \\
&\quad + \left(\frac{1}{4}\right) (\theta - \theta_0) \sum_{k \in \{1,2,4,5\}} y_{k-1} w_{k+1} + \frac{1}{4} \sum_{k \in \{1,2,4,5\}} w_k w_{k+1} \\
&\quad + \frac{1}{4} \sum_{k \in \{1,2,4,5\}} w_k w_{k+1} = 0 & (4.68)
\end{aligned}$$

l'équation 4.68 représente la moyenne des fonctions de corrélation. Si cette moyenne est nulle pour $\theta = \theta_0$, donc nous pouvons conclure que la valeur estimée θ_0 se trouve aux environs où les fonctions moyennes se croisent sur l'axe de θ

Par conséquent, le principe de la méthode LSCR et d'éliminer deux fonctions de corrélation les plus à droite et deux les plus à gauche, ce qui correspond à $q = 2$. Nous obtenons sur la figure 4.26, un intervalle $[-0.04, 0.48]$ qui représente une région de confiance pour θ_0 avec une probabilité égale à $1 - \frac{2-2}{8} = 0.5$ ($q = 2$ et $n=8$ mesures).

Dans cet exemple, l'intervalle obtenu est un peu large, avec une probabilité de 50%, afin d'améliorer cette probabilité et diminuer la largeur de l'intervalle solution, il faut augmenter le nombre des mesures.

4.12 Caractérisation garantie

L'inconvénient de l'approche LSCR est la difficulté à caractériser numériquement ces régions de confiance. M.KIEFFER [56] a porté une amélioration à cette approche en la combinant avec les méthodes ensemblistes, en particulier, l'inversion ensembliste, ce qui permet de garantir les résultats. LSCR nous permet de caractériser un ensemble :

$$\Theta_q = \left\{ p \in \mathbb{P} \text{ tel que } \sum_{i=1}^m \tau_i(p) \geq q \right\} \quad (4.69)$$

où :

$$\tau_i(p) = \begin{cases} 1 & \text{si } g_{i,r}^\epsilon(p) \geq 0 \\ 0 & \text{ailleurs} \end{cases} \quad (4.70)$$

Cette expression peut être reformulée comme un problème d'inversion ensembliste

$$\Theta_q = \mathbb{P} \cap \tau^{-1}([q, m]) \quad \text{avec} \quad \tau_p = \sum_{i=1}^m \tau_i(p) \quad (4.71)$$

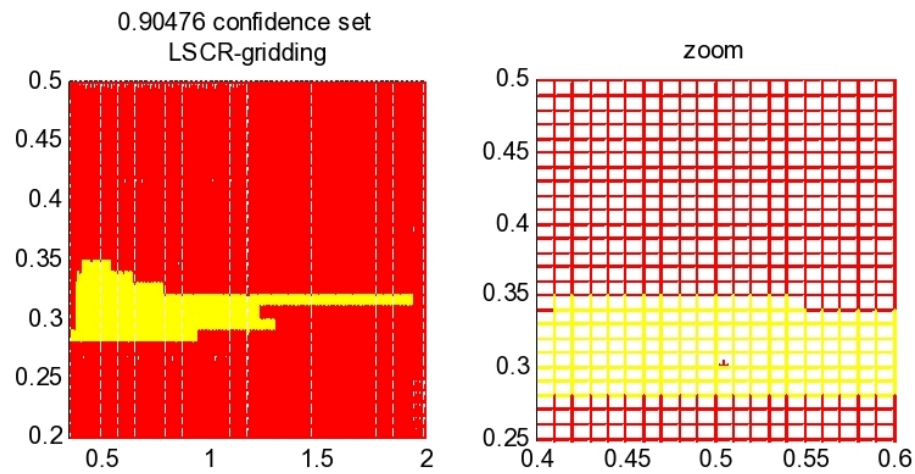
La théorie des ensembles possède deux points faibles : la première est l'augmentation exponentielle du temps de simulation avec le nombre de paramètres à estimer. Le second est le pessimisme qui a déjà été décrit dans les chapitres précédant.

4.13 Application au Bioréacteur

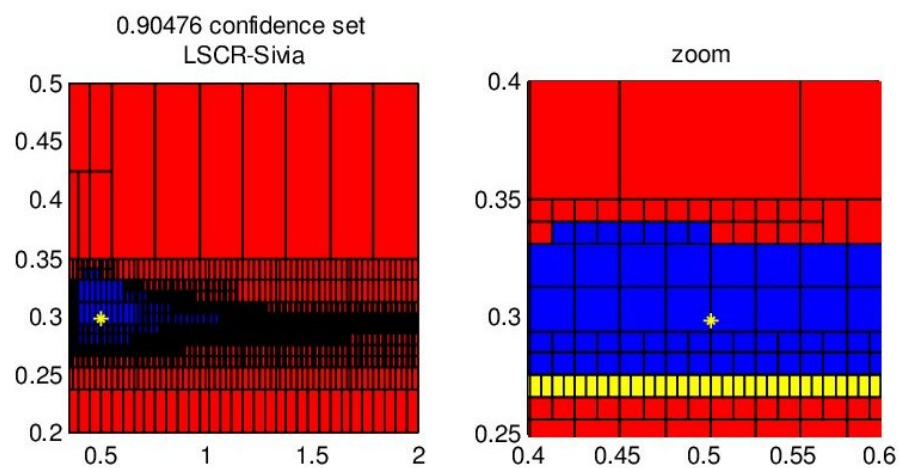
La méthode LSCR a été testée sur le même modèle de chémostat décrit auparavant. On commence par déterminer une région de confiance correspondant au taux de croissance maximum, $\mu_{max1}(S_1)$ et $\mu_{max2}(S_2)$, les autres paramètres sont considérés constants. Le pavé de recherche initial est $\mathbb{P} = [0.35, 2] \times [0.2, 1]$. Pour un nombre de mesures $n = 48$, les paramètres entiers $r=1$ et $q=3$ nous obtenons une région de confiance avec une probabilité à 90,47%. Toutes ces simulations ont été effectuées sur un Core I3 avec une précision $\epsilon = 0,01$, une variance=0.001.

La figure (4.27-a), représente la région de confiance $\Theta_{r,q}^\epsilon$ obtenue par maillage avec un pas $\epsilon = 0.01$, en $t = 1.23s$. Sur la figure (4.27-b) sont représentés, les pavés

solutions obtenus par l'algorithme Sivia, le temps de simulation est $t = 15.328s$.



(a) Gridding-LSCR



(b) Sivia :Fonction d'inclusion naturelle

FIGURE 4.27 – (a) Maillage de l'espace de recherche obtenu par LSCR, (b) Pavage de domaine avec Sivia, et leur zoom autour du paramètre apriori (*)

Les figures suivantes 4.28 représentent le résultat de l'identification des paramètres k_1 et k_2 , pour un pavé de recherche initial :

$p_0 = [0.5, 1] \times [0.5, 2]$ ont obtenu des pavés solutions en un temps de simulation $T = 35.077946s$

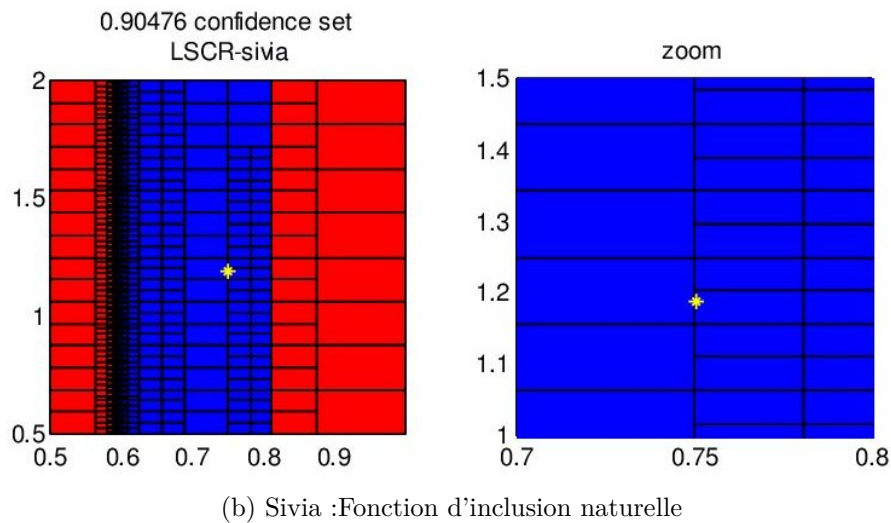
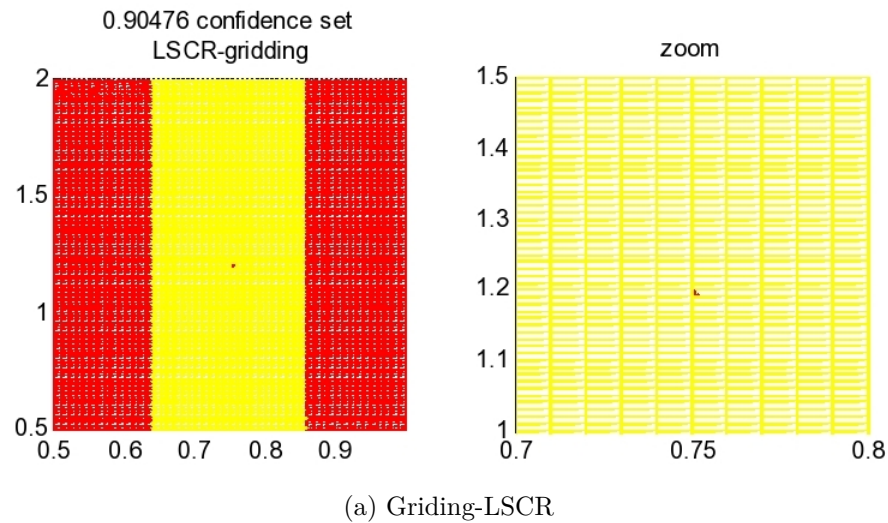
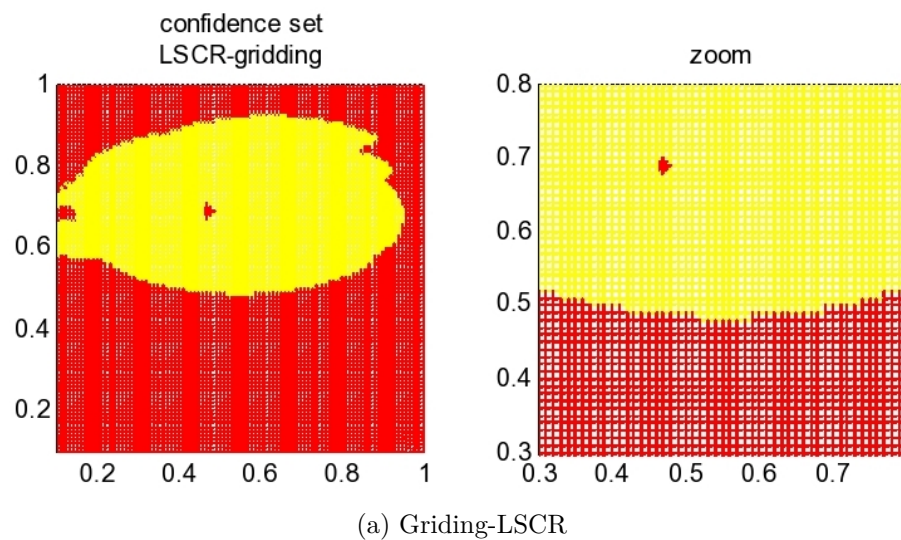


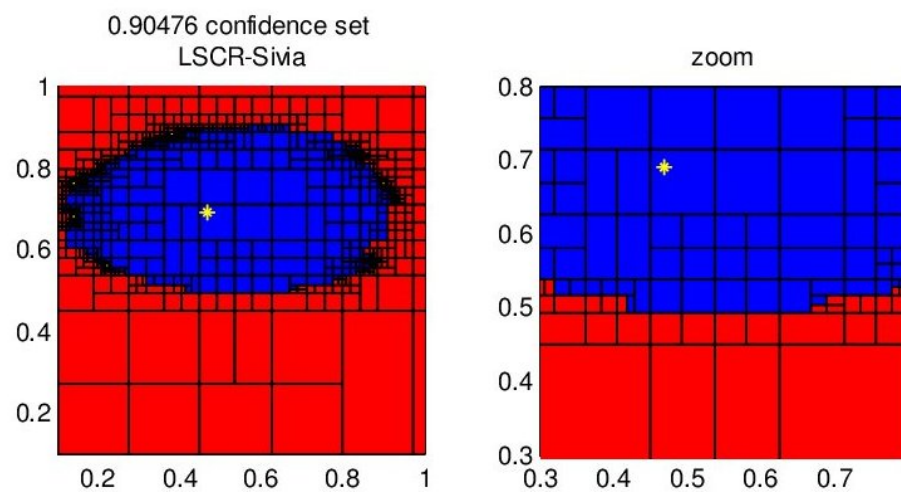
FIGURE 4.28 – (a) Maillage de l'espace de recherche obtenu par LSCR, (b) Pavage de domaine avec Sivia, et leur zoom autour du paramètre a priori (*)

Les figures suivantes 4.29 représentent le résultat de l'identification des paramètres ks_1 et ks_2 , pour un pavé de recherche initial :

$p_0 = [0.1, 1] \times [0.1, 1]$ ont obtenu des pavés solutions en un temps de simulation $T = 8.645434s$



(a) Griding-LSCR



(b) Sivia :Fonction d'inclusion naturelle

FIGURE 4.29 – (a) Maillage de l'espace de recherche obtenu par LSCR, (b) Pavage de domaine avec Sivia, et leur zoom autour du paramètre apriori (*)

Tableau 4.2 – Résultats

Paramètres	paramètres a priori	Paramètres estimés(MC)	LSCR+Sivia
μ_{max1}	[0.45, 0.5]	0.5049	[0.3951, 1.0655]
μ_{max2}	0.27	0.2994	[0.2749, 0.3032]
$k1$	5.555	0.751	[0.6484, 0.7422]
$k2$	10	1.1875	[0.5000, 2.000]
$ks1$	[0.3, 0.48]	0.47	[0.2828, 0.8946]
$ks2$	[0.65, 0.7]	0.67	[0.5359, 0.7891]

4.14 Conclusion

Les points forts de la méthode LSCR sont les hypothèses minimales concernant le bruit additif et l'obtention d'un résultat avec une probabilité garantie. Cette méthode développée par Campi et améliorée par M.Kieffer fournit des résultats garantis. En effet, l'algorithme Sivia associé évalue les approximations intérieures et extérieures des régions de confiance asymptotiques non déterminées par LSCR.

Conclusion générale et Perspectives

« C'est le devoir de chaque homme de rendre au monde au moins autant qu'il en a reçu. »

Albert Einstein, (Physicien)

L'une des préoccupations majeure de la communauté scientifique est de pouvoir décrire un système dynamique par un modèle mathématique le plus précisément et fidèlement possible. Pour cette raison, des techniques d'identification abondent dans la littérature. Selon les dires de Ljung[60] « il existe un impénétrable mais transparent écran entre la description mathématique de notre monde et le monde réel ». Ce qui veut dire qu'il est impossible d'établir des relations exactes entre le processus et son modèle mathématique mais l'on peut essayer de s'en rapprocher. Un modèle est construit sur la base de données observées, d'où la difficulté quand il s'agit d'étudier des systèmes biologiques, médicaux ou biotechnologiques quand le temps se compte en jours et quand l'intervalle d'erreur est trop large. Dans cette thèse nous testons de nouvelles techniques pour identifier les paramètres d'un chémostat. Ces techniques consistent à l'utilisation de méthodes ensemblistes basées sur l'arithmétique des intervalles. Le premier chapitre a été consacré à la description du dispositif expérimentale sur lequel s'est portée notre étude, il s'agit d'un bioréacteur en fonctionnement continu qu'on appelle chémostat. Ce dispositif modélise les deux étapes de nitrification du cycle de l'azote dans le cadre du traitement des eaux usées par boues activées. Les micro organismes nitrifiants composants la biomasse sont dominés par deux types de bactéries, Nitrosomonas (AOB) et Nitrobacter (NOB) qui suivent une cinétique de Monod. En se basant sur

le bilan de matière, le modèle mathématique du chémostat obtenu est décrit par cinq équations différentielles. Dans le chapitre suivant, Nous avons présenté les bases de l'arithmétique des intervalles, rappelé les différentes opérations ensemblistes ainsi que les types de récipients pouvant contenir ces ensembles tels que des pavés afin de les manipuler aisément. Nous avons aussi défini des notions importantes telles que la fonction d'inclusion et le pessimisme résultant qui provoque une surestimation d'un pavé solution. La solution apportée à ce problème de pessimisme est l'introduction de contracteurs. Il en existe plusieurs mais nous avons présenté en détails la contraction par projection qui est celle qu'on a utilisé dans cette thèse. Nous avons terminé ce chapitre par la définition de l'inversion ensembliste qui est la base de l'identification dans le contexte à erreurs bornées. Dans le troisième chapitre nous décrivons en détail l'identification dans le contexte à erreurs bornées. Nous avons débuté par un rappel des méthodes classiques de l'identification afin de procéder à une étude comparative avec celles utilisant les méthodes ensemblistes et déduit par la suite les avantages et inconvénients de ces techniques. Afin d'illustrer l'identification dans le contexte à erreurs bornées utilisant l'inversion ensembliste, un exemple de robot plan à deux degrés de liberté du type Scara est testé. L'essentiel de notre travail est présenté dans le quatrième chapitre. Il s'agit d'identifier certains paramètres d'un chémostat. Dans un premier temps, nous avons choisi d'identifier les taux de croissance maximaux $\mu_{max1}(S_1)$ et $\mu_{max2}(S_2)$ pour les deux populations bactériennes X_1 et X_2 , respectivement, puis les rendements k_1 et k_2 et ks_1 et enfin ks_2 Nous avons partagé ce chapitre en deux parties :

La première est consacrée à l'intégration numérique garantie utilisant la théorie de comparaison des inégalités différentielles qui nous a permis d'encadrer toutes les solutions de notre système entre deux solutions englobantes qu'on a minimisé en utilisant un critère des moindres carrés. Nous avons d'abord montré que notre système n'était ni monotone, ni préservateur d'ordre, ni coopératif, par conséquent les solutions englobantes sont obtenues par l'application d'une règle basée sur le théorème de Müller [71]. Notons qu'il est possible de rechercher une nouvelle base de coordonnées des EDO de façon à obtenir un système coopératif si le système est préservateur d'ordre, ce qui n'est pas le cas pour notre modèle. L'identification des paramètres par inversion ensembliste est effectuée en exécutant l'algorithme SIVIA qu'on a associé à l'intégration numérique basée sur la théorie de comparaison des

inégalités différentielles. Cela a permis de caractériser, d'une manière globale et garantie, l'ensemble solution.

Dans la deuxième partie de ce chapitre nous utilisons la méthode LSCR (Leave-out Sign-dominant Correlation Regions), c'est une nouvelle méthode d'identification proposée par MC.CAMPI [20]. Le principe est basé sur la construction d'une région de confiance Θ , à partir du calcul des fonctions de corrélation empiriques. L'avantage de cette méthode est que ces régions de confiance sont déterminées de manière non asymptotique et avec une probabilité garantie, quelle que soit la taille de l'ensemble. D'autre part, les hypothèses sur le bruit qui affectent les données sont très réduites, nous pouvons simplement supposer que c'est un signal indépendant et réparti symétriquement autour de zéro, la variance peut prendre n'importe quelle valeur. L'inconvénient de l'approche LSCR est la difficulté à caractériser numériquement ces régions de confiance. M.KIEFFER [56] a apporté une amélioration à cette approche en la combinant avec les méthodes ensemblistes, en particulier, l'inversion ensembliste, ce qui permet de garantir les résultats. Nous avons testé cette méthode qui nous a paru très intéressante.

Pour les deux méthodes d'identification ensemblistes utilisées, nous obtenons des encadrements pour les paramètres identifiés relativement proche et de plus ils contiennent les valeurs a priori. Cependant, l'inconvénient majeur de ces méthodes reste le pessimisme introduit pendant les calculs, ce qui a fourni des intervalles plutôt larges. Dans la première partie de notre application, on a pu atténuer l'effet de ce pessimisme par l'introduction d'un contracteur, de plus le travail a été effectué avec la toolbox Intlab de Matlab qui n'est pas le plus adapté pour l'arithmétique des intervalles, la programmation en C++ améliore nettement les résultats en termes d'encadrement et en temps de simulation. Dans la seconde partie, pour remédier à ce problème de pessimisme, on a essayé d'utiliser la forme centrée de la fonction d'inclusion qui n'a pas donné des résultats très concluants, l'introduction d'un contracteur constitue une suite à ce travail. Les perspectives à ce travail sont nombreuses puisque les recherches dans le domaine de l'ensembliste et l'arithmétique des intervalles sont en plein essor, on peut s'orienter vers la recherche de nouvelles techniques de contraction. Il existe la bibliothèque `ibex`[3] qui résout des problèmes d'optimisation non linéaires utilisant le principe de propagation de contraintes et l'arithmétique des intervalles. Avec la bibliothèque `VNODELP`, on

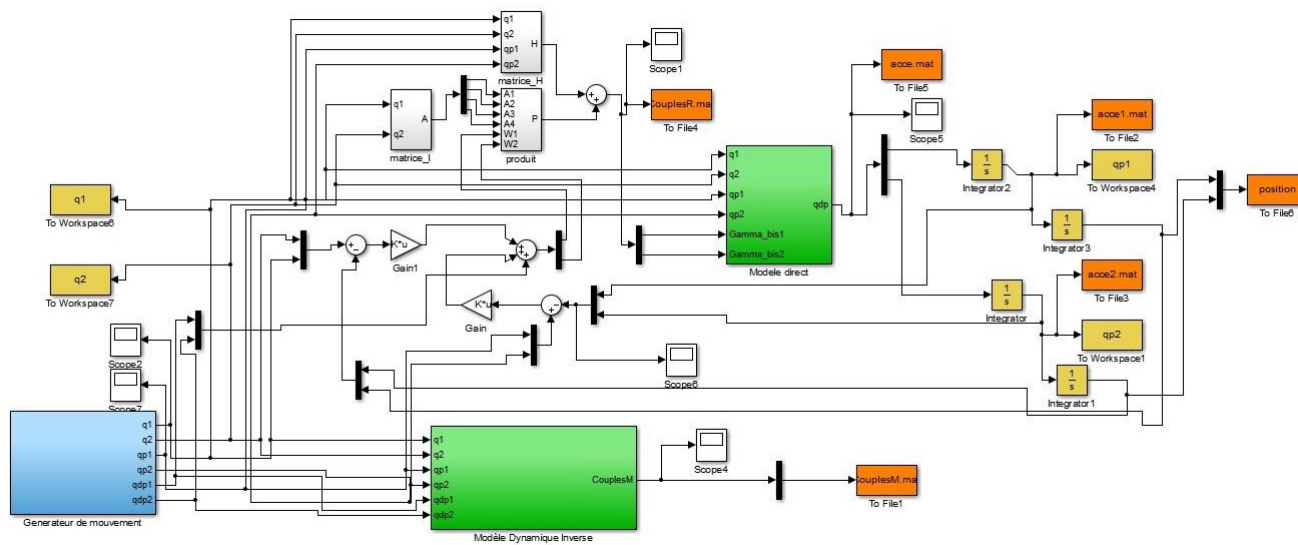
a juste procédé à l'intégration numérique garantie, comme suite à ce travail, on peut procéder à l'identification des paramètres du chémostat utilisant l'inversion ensembliste et essayer d'identifier les six paramètres du chémostats en même temps. Tester d'autres estimateurs associés aux techniques ensemblistes serait intéressant comme le filtre de KALMAN. Le calculateur influe sur les résultats, pour identifier plusieurs paramètres du chémostat en même temps, il faut bien choisir le calculateur sur lequel on travaille, plus rapide et plus précis.

Annexe A

Annexes

Annexe 1

Schéma simulink d'un modèle de robot Scara



Annexe 2

La construction de Gordon, de la matrice d'incidence d'un groupe

Soit $I = \{1, \dots, N\}$, Une matrice d'incidence pour un groupe $\{I_n\}$ de sous ensemble de I est une matrice dont les éléments (i, j) sont tels que :

$(i, j) = 1$ si $j \in \{I_n\}$ et $(i, j) = 0$ ailleurs.

L.GORDON [34] décrit la prodédure de construction de cette matrice d'incidence R de la façon suivante :

soit $I = \{1, \dots, 2^l - 1\}$ et un groupe possédant 2^l éléments.

On pose $R(1) = [1]$ et on calcule récursivement pour $k = 1, 2, \dots, l$:

$$R(2^k - 1) = \begin{bmatrix} R(2^{k-1} - 1) & R(2^{k-1} - 1) & 0 \\ R(2^{k-1} - 1) & J - R(2^{k-1} - 1) & e \\ 0^T & e^T & 1 \end{bmatrix} \quad (\text{A.2.1})$$

où J et e sont, respectivement, une matrice et un vecteur de tous les 1, et 0 est un vecteur de tous les zéros. Par suite, soit ;

$$R = \begin{bmatrix} R(2^{k-1} - 1) \\ 0^T \end{bmatrix} \quad (\text{A.2.2})$$

Gordon [34] donne également la construction de groupes lorsque le nombre de points expérimentaux est différent de $2^l - 1$

Annexe 3

Démonstration du théorème 4

Si l'on reprend l'exemple de Campi, la région de confiance est obtenue en éliminant les régions de l'espace des paramètres où toutes les fonctions $g_i(\theta)$, $i = 1, \dots, 7$ sont supérieures à zéro, ou au plus une d'entre eux est inférieure à zéro et où toutes les fonctions sont inférieures à zéro, ou au plus une d'entre eux est supérieure à zéro. Par conséquent, la vraie valeur du paramètre θ_0 tombe en dehors de la zone de confiance lorsque toutes les fonctions $g_i(\theta)$, $i = 1, \dots, 7$ sont plus grandes que $g_8(\theta) = 0$, ou tout au plus une d'entre eux est inférieure à $g_8(\theta)$ et où toutes les fonctions sont inférieures à $g_8(\theta)$, ou au plus une d'entre eux est plus grande que $g_8(\theta)$. il fut démontré [21] que chacun de ces évènements a la probabilité $1/8$ de se produire, de sorte que la probabilité totale pour que θ_0 tombe en dehors de la zone de confiance est de $4.1/8 = 0,5$ Considérons une seule condition et calculons la probabilité où toutes les fonctions $g_i(\theta)$, $i = 1, \dots, 7$ sont supérieurs à $g_8(\theta) = 0$, les autres conditions dérivent de façon similaires.

On a alors :

$$\left\{ \begin{array}{l} w_1w_2 + w_2w_3 + w_4w_5 + w_5w_6 > 0 \\ w_1w_2 + w_3w_4 + w_4w_5 + w_6w_7 > 0 \\ w_2w_3 + w_3w_4 + w_5w_6 + w_6w_7 > 0 \\ w_1w_2 + w_2w_3 + w_6w_7 + w_7w_8 > 0 \\ w_1w_2 + w_3w_4 + w_5w_6 + w_7w_8 > 0 \\ w_2w_3 + w_3w_4 + w_4w_5 + w_7w_8 > 0 \\ w_4w_5 + w_5w_6 + w_6w_7 + w_7w_8 > 0 \end{array} \right. \quad (A.3.1)$$

Pour calculer la probabilité pour que tous ces 7 inégalités soient simultanément vraies, nous comparons $g_i(\theta)$, $i = 1, \dots, 7$ avec $g_1(\theta) = 0$ au lieu de $g_8(\theta) = 0$

$$\left\{ \begin{array}{l} w_1w_2 + w_3w_4 + w_4w_5 + w_6w_7 > w_1w_2 + w_2w_3 + w_4w_5 + w_5w_6 \\ w_2w_3 + w_3w_4 + w_5w_6 + w_6w_7 > w_1w_2 + w_2w_3 + w_4w_5 + w_5w_6 \\ w_1w_2 + w_2w_3 + w_6w_7 + w_7w_8 > w_1w_2 + w_2w_3 + w_4w_5 + w_5w_6 \\ w_1w_2 + w_3w_4 + w_5w_6 + w_7w_8 > w_1w_2 + w_2w_3 + w_4w_5 + w_5w_6 \\ w_2w_3 + w_3w_4 + w_4w_5 + w_7w_8 > w_1w_2 + w_2w_3 + w_4w_5 + w_5w_6 \\ w_4w_5 + w_5w_6 + w_6w_7 + w_7w_8 > w_1w_2 + w_2w_3 + w_4w_5 + w_5w_6 \\ 0 > w_1w_2 + w_2w_3 + w_4w_5 + w_5w_6 \end{array} \right. \quad (A.3.2)$$

En déplaçant tous les termes à gauche, on a :

$$\left\{ \begin{array}{l} -w_2w_3 + w_3w_4 - w_5w_6 + w_6w_7 > 0 \\ -w_1w_2 + w_3w_4 - w_4w_5 + w_6w_7 > 0 \\ -w_4w_5 - w_5w_6 + w_6w_7 + w_7w_8 > 0 \\ -w_2w_3 + w_3w_4 - w_4w_5 + w_7w_8 > 0 \\ -w_1w_2 + w_3w_4 - w_5w_6 + w_7w_8 > 0 \\ -w_1w_2 - w_2w_3 + w_6w_7 + w_7w_8 > 0 \\ -w_1w_2 - w_2w_3 - w_4w_5 - w_5w_6 > 0 \end{array} \right. \quad (A.3.3)$$

Si l'on remplace w_2 par \tilde{w}_2 et w_5 par \tilde{w}_5

$$\left\{ \begin{array}{l} \tilde{w}_2w_3 + w_3w_4 + \tilde{w}_5w_6 + w_6w_7 > 0 \\ w_1\tilde{w}_2 + w_3w_4 + w_4\tilde{w}_5 + w_6w_7 > 0 \\ w_4\tilde{w}_5 + \tilde{w}_5w_6 + w_6w_7 + w_7w_8 > 0 \\ \tilde{w}_2w_3 + w_3w_4 + w_4\tilde{w}_5 + w_7w_8 > 0 \\ w_1\tilde{w}_2 + w_3w_4 + \tilde{w}_5w_6 + w_7w_8 > 0 \\ w_1\tilde{w}_2 + \tilde{w}_2w_3 + w_6w_7 + w_7w_8 > 0 \\ w_1\tilde{w}_2 + \tilde{w}_2w_3 + w_4\tilde{w}_5 + \tilde{w}_5w_6 > 0 \end{array} \right. \quad (A.3.4)$$

On remarque que nous obtenons les même équation que A.3.1, ceci est dû à la propriété de groupe des sous ensembles I_i . Nous pouvons conclure que les équations

(A.3.1) et (A.3.2) sont satisfaits pour la même probabilité, puisque les w_t sont réparties de façon symétrique. Les mêmes résultats sont obtenus si nous comparons les équations avec $g_2(\theta_0)$, $g_3(\theta_0)$, ... etc au lieu de $g_1(\theta_0)$. Étant donné que ces 8 possibilités sont disjointes et ont la même probabilité pour se produire, alors on conclue aussi que chaque événement a exactement une probabilité de $1/8$ pour se produire.

Bibliographie

- [1] <http://dutita0.twi.tudelft.nl/users/vuik/wi211/disasters.html>. 17
- [2] <http://www.boost.org/>. 77
- [3] <http://www.emn.fr/z-info/ibex/>. 77, 135
- [4] <http://www.ensta-bretagne.fr/jaulin/demo.html>. 77
- [5] <http://www.cas.mcmaster.ca/~nedialk/vnodelp/>. 103
- [6] <http://marco-campi.unibs.it/LSCRwebsite/index.html>. 124
- [7] John F Andrews. A mathematical model for the continuous culture of microorganisms utilizing inhibitory substrates. *Biotechnology and Bioengineering*, 10(6) :707–723, 1968. 33
- [8] AC Anthonisen, RC Loehr, TBS Prakasam, and EG Srinath. Inhibition of nitrification by ammonia and nitrous acid. *Journal (Water Pollution Control Federation)*, pages 835–852, 1976. 28
- [9] Xavier Baguenard. *Propagation de contraintes sur les intervalles Application à l'étalonnage des robots*. PhD thesis, INRIA SOPHIA ANTIPOLIS, 2005. 77
- [10] Xavier Baguenard, Massa Dao, Luc Jaulin, and Wisama Khalil. Méthodes ensemblistes pour l'étalonnage géométrique. *JOURNAL EUROPEEN DES SYSTEMES AUTOMATISES.*, 37(9) :1059–1074, 2003. 19
- [11] Georges Bastin and Denis Dochain. On-line estimation of microbial specific growth rates. *Automatica*, 22(6) :705–709, 1986. 33

- [12] Naveen Bhat and Thomas J McAvoy. Use of neural nets for dynamic modeling and control of chemical process systems. *Computers & Chemical Engineering*, 14(4) :573–582, 1990. [20](#)
- [13] E Bock and HP Koops. The genus nitrobacter and related genera. *Balows A, Trâ-uper HG, Dworkin M, Harder W, Schleifer KH, editors. The prokaryotes. New York : Springer, pages 2302 – –9, 1992.* [27](#)
- [14] Eberhard Bock, Hans-Peter Koops, Heinz Harms, and B Ahlers. The biochemistry of nitrifying organisms. *Variations in autotrophic life*, pages 171–200, 1991. [29](#)
- [15] Daisy Bougard. *Traitement biologique d’effluents azotés avec arrêt de la nitrification au stade nitrite*. PhD thesis, École nationale supérieure agronomique (Montpellier), 2004. [27](#)
- [16] I Braems, N Ramdani, A Boudenne, M Kieffer, L Jaulin, L Ibos, E Walter, and Y Candau. New set-membership techniques for parameter estimation in presence of model uncertainty. In *Proceedings of the 5th International Conference on Inverse Problems in Engineering : Theory and Practice*, volume 11, page 15th, 2005. [19](#)
- [17] Isabelle Braems. Méthodes ensemblistes garanties pour l’estimation de grandeurs physiques. *Phd of*, 2002. [58](#)
- [18] Isabelle Braems, Fabienne Berthier, Luc Jaulin, Michel Kieffer, and Eric Walter. Guaranteed estimation of electrochemical parameters by set inversion using interval analysis. *Journal of Electroanalytical Chemistry*, 495(1) :1–9, 2000. [19](#)
- [19] Isabelle Braems, Luc Jaulin, Michel Kieffer, Nacim Ramdani, and Eric Walter. Reliable parameter estimation in presence of uncertain variables that are not estimated. In *Proceedings of the 13th IFAC Symposium on System Identification SYSID*, volume 3, pages 1856–1861, 2003. [19](#)
- [20] Marco C Campi and Erik Weyer. Guaranteed non-asymptotic confidence regions in system identification. *Automatica*, 41(10) :1751–1764, 2005. [19](#), [121](#), [123](#), [135](#)

- [21] Marco C Campi and Erik Weyer. Identification with finitely many data points : The lscr approach. In *System Identification*, volume 14, pages 46–64, 2006. [19](#), [121](#), [124](#), [139](#)
- [22] MC Campi and Erik Weyer. Estimation of confidence regions for the parameters of arma models guaranteed non-asymptotic results. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 6, pages 6009–6014. IEEE, 2003. [19](#)
- [23] Jean-Pierre Canler, Jean-Marc Perret, Philippe Duchène, and Eric Cotteux. *Aide au diagnostic des stations d'épuration par l'observation microscopique des boues activées*. Editions Quae, 2011. [25](#)
- [24] Niels Jacob Carstensen and Henrik Madsen. *Identification of wastewater processes*. PhD thesis, Technical University of DenmarkDanmarks Tekniske Universitet, AdministrationAdministration, Office for Study Programmes and Student AffairsAfdelingen for Uddannelse og Studerende, 1993. [20](#)
- [25] Gilles Chabert and Luc Jaulin. Contractor programming. *Artificial Intelligence*, 173(11) :1079–1100, 2009. [62](#)
- [26] DE Contois. Kinetics of bacterial growth : relationship between population density and specific growth rate of continuous cultures. *Microbiology*, 21(1) :40–50, 1959. [33](#)
- [27] Denis Couillard and Shucaï Zhu. Control strategy for the activated sludge process under shock loading. *Water Research*, 26(5) :649–655, 1992. [20](#)
- [28] Massa Dao. *Caractérisation d'ensembles par des méthodes intervalles : applications en automatique*. PhD thesis, Angers, 2006. [77](#)
- [29] Jean-Guillaume Dumas, Thierry Gautier, Mark Giesbrecht, Pascal Giorgi, Bradford Hovinen, Erich Kaltofen, B David Saunders, Will J Turner, Gilles Villard, et al. Linbox : A generic library for exact linear algebra. In *Proceedings of the 2002 International Congress of Mathematical Software, Beijing, China*, pages 40–50. World Scientific Pub, 2002. [77](#)

- [30] Maxime Dumont. *Apports de la modélisation des interactions pour une compréhension fonctionnelle d'un écosystème : application à des bactéries nitrifiantes en chémostat*. PhD thesis, Montpellier 2, 2008. [35](#), [36](#), [38](#), [105](#), [107](#)
- [31] I Enbutsu, K Baba, N Hara, K Waseda, and S Nogita. Integration of multi ai paradigms for intelligent operation support systems—fuzzy rule extraction from a neural network. *Water Science & Technology*, 28(11-12) :333–340, 1993. [20](#)
- [32] JM Garrido, WAJ Van Benthum, MCM Van Loosdrecht, and JJ Heijnen. Influence of dissolved oxygen concentration on nitrite accumulation in a biofilm airlift suspension reactor. *Biotechnology and bioengineering*, 53(2) :168–178, 1997. [28](#)
- [33] Maxime Gautier. *Contribution à la modélisation et à l'identification des robots*. PhD thesis, Nantes, 1990. [75](#)
- [34] Louis Gordon et al. Completely separating groups in subsampling. *The Annals of Statistics*, 2(3) :572–578, 1974. [122](#), [138](#)
- [35] Jean-Luc Gouzé and Karl P Haderler. Monotone flows and order intervals. *Nonlinear World*, 1 :23–34, 1994. [20](#)
- [36] Jean-Luc Gouzé, A Rapaport, and Mohamed Zakaria Hadj-Sadok. Interval observers for uncertain biological systems. *Ecological modelling*, 133(1) :45–56, 2000. [20](#)
- [37] MZ Hadj-Sadok and JL Gouzé. Estimation of uncertain models of activated sludge processes with interval observers. *Journal of Process Control*, 11(3) :299–310, 2001. [20](#)
- [38] Eldon Hansen and G William Walster. *Global optimization using interval analysis : revised and expanded*, volume 264. CRC Press, 2003. [19](#), [57](#)
- [39] ER Hansen and RI Greenberg. An interval newton method. *Applied Mathematics and Computation*, 12(2) :89–98, 1983. [49](#)
- [40] C Hellinga, AAJC Schellen, J Wf Mulder, MCM Van Loosdrecht, and JJ Heijnen. The sharon process : an innovative method for nitrogen removal from

- ammonium-rich waste water. *Water science and technology*, 37(9) :135–142, 1998. [28](#)
- [41] Mogens Henze, CPL Grady Jr, Willi Gujer, GvR Marais, and T Matsuo. Activated sludge model no. 1 : IAWPRC scientific and technical report no. 1. *IAWPRC, London*, 1987. [20](#)
- [42] Morris W Hirsch and Hal Smith. Monotone dynamical systems. *Handbook of differential equations : ordinary differential equations*, 2 :239–357, 2006. [87](#)
- [43] Bailey James and F Ollis David. *Biochemical engineering fundamentals*. McGraw Hill Book Company, 1986. [33](#)
- [44] Luc Jaulin. Interval constraint propagation with application to bounded-error estimation. *Automatica*, 36(10) :1547–1552, 2000. [59](#)
- [45] Luc Jaulin. Le calcul ensembliste par analyse par intervalles et ses applications. *Habilitation à diriger des recherches (HDR)*, 2000. [67](#), [68](#)
- [46] Luc Jaulin. Path planning using intervals and graphs. *Reliable computing*, 7(1) :1–15, 2001. [19](#)
- [47] Luc Jaulin. A nonlinear set membership approach for the localization and map building of underwater robots. *IEEE Transactions on Robotics*, 25(1) :88–98, 2009. [19](#)
- [48] Luc Jaulin, Isabelle Braems, and Eric Walter. Interval methods for nonlinear identification and robust control. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 4, pages 4676–4681. IEEE, 2002. [62](#)
- [49] Luc Jaulin, Michel Kieffer, Olivier Didrit, and Eric Walter. *Applied Interval Analysis*. Springer, 2001. [55](#), [56](#), [60](#)
- [50] Luc Jaulin and Eric Walter. Guaranteed nonlinear parameter estimation from bounded-error data via interval analysis. *Mathematics and computers in simulation*, 35(2) :123–137, 1993. [19](#)
- [51] John C Kabouris and Aris P Georgakakos. Stochastic control of the activated sludge process. *Water Science and Technology*, 24(6) :249–255, 1991. [20](#)

- [52] Wisama Khalil and Etienne Dombre. *Modélisation, identification et commande des robots*. Hermès science publ., 1999. [75](#)
- [53] Wisama Khalil and JF Kleinfinger. A new geometric notation for open and closed-loop robots. In *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, volume 3, pages 1174–1179. IEEE, 1986. [73](#)
- [54] Michel Kieffer. *Estimation ensembliste par analyse par intervalles Application à la localisation d'un véhicule*. PhD thesis, Université Paris Sud-Paris XI, 1999. [19](#), [54](#)
- [55] Michel Kieffer and Eric Walter. Guaranteed nonlinear state estimation for continuous-time dynamical models from discrete-time measurements. *IFAC Proceedings Volumes*, 39(9) :685–690, 2006. [19](#), [94](#), [95](#), [98](#), [100](#)
- [56] Michel Kieffer and Eric Walter. Guaranteed characterization of exact non-asymptotic confidence regions in nonlinear parameter estimation. In *IFAC Symposium on Nonlinear Control Systems*, pages 56–61, Toulouse, France, September 2013. [20](#), [128](#), [135](#)
- [57] Donald E Knuth. Literate programming. *CSLI Lecture Notes, Stanford, CA : Center for the Study of Language and Information (CSLI), 1992*, 1, 1992. [103](#)
- [58] Ulrich W Kulisch and Willard L Miranker. *A new approach to scientific computation*, volume 7. Elsevier, 2014. [19](#)
- [59] H Kunze and D Siegel. A graph theoretical approach to monotonicity with respect to initial conditions ii. *Nonlinear Analysis : Theory, Methods & Applications*, 35(1) :1–20, 1999. [89](#)
- [60] Lennart Ljung. *System identification - theory for the user*, 1999. [133](#)
- [61] Eugene Loh and G. William Walster. Rump's example revisited. *Reliable Computing*, 8(3) :245–248, 2002. [18](#)
- [62] C Marcelli and P Rubbioni. A new extension of classical müller's theorem. *Nonlinear Analysis : Theory, Methods & Applications*, 28(11) :1759–1767, 1997. [87](#)

- [63] Nacim Meslem. *Atteignabilité hybride des systèmes dynamiques continus par analyse par intervalles : application à l'estimation ensembliste*. PhD thesis, Université Paris-Est, 2008. [90](#), [91](#), [94](#)
- [64] Nacim Meslem and Nacim Ramdani. Interval observer design based on nonlinear hybridization and practical stability analysis. *International Journal of Adaptive Control and Signal Processing*, 25(3) :228–248, 2011. [19](#), [94](#), [98](#)
- [65] Mario Milanese and Antonio Vicino. Estimation theory for nonlinear models and set membership uncertainty. *Automatica*, 27(2) :403–408, 1991. [19](#)
- [66] Mario Milanese and Antonio Vicino. Optimal estimation theory for dynamic systems with set membership uncertainty : an overview. In *Bounding Approaches to System Identification*, pages 5–27. Springer US, 1996. [19](#)
- [67] Jacques Monod. *La technique de culture continue : theorie et applications*. 1950. [31](#), [37](#), [105](#)
- [68] R Moore. *Interval arithmetic*. Prentice-Hall, Englewood Cliffs (NJ), USA, 1966. [19](#)
- [69] Ramon E Moore, R Baker Kearfott, and Michael J Cloud. *Introduction to interval analysis*. Siam, 2009. [19](#)
- [70] Ramon E Moore and RE Moore. *Methods and applications of interval analysis*, volume 2. SIAM, 1979. [19](#)
- [71] Max Müller. Über das fundamentaltheorem in der theorie der gewöhnlichen differentialgleichungen. *Mathematische Zeitschrift*, 26(1) :619–645, 1927. [87](#), [94](#), [98](#), [134](#)
- [72] Nediialko S Nediialkov. Vnode-lpâa validated solver for initial value problems in ordinary differential equations. *Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada, L8S 4K1, Technical Report CAS-06-06-NN, November 2006a*. Retrieved from : <http://www.cas.mcmaster.ca/~nedialk/vnodelp>, 2006. [77](#), [103](#)

- [73] Nedialko S Nedialkov, Kenneth R Jackson, and George F Corliss. Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105(1) :21–68, 1999. [19](#)
- [74] A. Neumaier. *Interval Methods for Systems of Equations*. Cambridge Middle East Library. Cambridge University Press, 1990. [19](#), [49](#), [57](#)
- [75] Aaron Novick and Leo Szilard. Experiments with the chemostat on spontaneous mutations of bacteria. *Proceedings of the National Academy of Sciences*, 36(12) :708–719, 1950. [30](#)
- [76] Gustaf Olsson, Bengt Andersson, Bengt Göran Hellstrom, Hans Holmström, Lars Gunnar Reinius, and Petr Vopatek. Measurements, data analysis and control methods in wastewater treatment plants—state of the art and future trends. *Water Science & Technology*, 21(10-11) :1333–1345, 1989. [20](#)
- [77] Jens Ejnar Parkum. *Recursive identification of time-varying systems*. IMSOR, 1992. [20](#)
- [78] Ph Poignet, Nacim Ramdani, O Vivas, et al. Robust estimation of parallel robot dynamic parameters with interval analysis. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 6, pages 6503–6508. IEEE, 2003. [19](#)
- [79] Claudine Presse. *Identification des paramètres dynamiques des robots*. PhD thesis, 1994. [76](#)
- [80] JI Prosser. Autotrophic nitrification in bacteria. *Advances in microbial physiology*, 30 :125–181, 1990. [27](#)
- [81] Tarek Raissi. *Méthodes ensemblistes pour l'estimation d'état et de paramètres*. PhD thesis, Université Paris XII Val de Marne, 2004. [47](#)
- [82] Tarek Raissi, Nacim Ramdani, and Yves Candau. Parameter estimation for non-linear continuous-time systems in a bounded error context. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 3, pages 2240–2245. IEEE, 2003. [19](#), [56](#)

- [83] Nacim Ramdani. Analyse et estimation robustes des systèmes hybrides non linéaires, une approche ensembliste. In *CIFA 2012*, pages 991–1002, 2012. [19](#), [100](#)
- [84] Nacim Ramdani, Nacim Meslem, Tarek Raissi, and Yves Candau. Set-membership identification of continuous-time systems. In *System Identification*, volume 14, pages 446–451, 2006. [96](#), [100](#)
- [85] Nacim Ramdani and Nedialko S Nedialkov. Computing reachable sets for uncertain nonlinear hybrid systems using interval constraint-propagation techniques. *Nonlinear Analysis : Hybrid Systems*, 5(2) :149–162, 2011. [19](#)
- [86] Nacim Ramdani and Philippe Poignet. Robust Dynamic Experimental Identification of Robots with Set Membership Uncertainty. *IEEE/ASME Transactions on Mechatronics*, 10(2) :253–256, 2005. [19](#)
- [87] Nacim Ramdani and Philippe Poignet. Short papers_. *IEEE/ASME TRANSACTIONS ON MECHATRONICS*, 10(2) :253, 2005. [19](#)
- [88] Nacim Ramdani, Tarek Raissi, Yves Candau, and Laurent Ibos. Complex interval constraint propagation for non linear bounded-error parameter identification. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 3590–3595. IEEE, 2005. [19](#)
- [89] Siegfried M Rump. *INTLABâinterval laboratory*. Springer, 1999. [77](#)
- [90] Hal L Smith. *Monotone dynamical systems : an introduction to the theory of competitive and cooperative systems*. Number 41. American Mathematical Soc., 2008. [87](#), [92](#), [94](#)
- [91] S.M.Rump. Algorithms for verified inclusions -theory and practice. In Reliability in Computing In R.E. Moore, editor, editor, *volume 19 of Perspectives in Computing*, page 109â126. Academic Press, 1988. [18](#)
- [92] Aki Sorsa, Riikka Peltokangas, and Kauko Leiviskä. Real-coded genetic algorithms and nonlinear parameter identification. In *Intelligent Systems, 2008. IS'08. 4th International IEEE Conference*, volume 2, pages 10–42. IEEE, 2008. [20](#)

-
- [93] Teruo Sunaga. Theory of an interval algebra and its application to numerical analysis. *Japan Journal of Industrial and Applied Mathematics*, 26(2-3) :125–143, 2009. [19](#)
- [94] Michel Villain. *Automatique : Systèmes asservis linéaires*. Ellipses, 1996. [63](#)
- [95] Eric Walter and Michel Kieffer. Sur quelques idées fausses ayant des conséquences en identification. *J. Européen des Systèmes Automatisés*, 42(2) :181–210, 2008. [16](#)
- [96] Wolfgang Walter. Differential inequalities and maximum principles : theory, new methods and applications. *Nonlinear Analysis : Theory, Methods & Applications*, 30(8) :4695–4711, 1997. [87](#), [93](#)
- [97] S Winogradsky. Recherches sur les organismes de la nitrification. *Ann. Inst. Pasteur*, 4(257) :11–12, 1890. [28](#)
- [98] Rosalind Cecily Young. The algebra of many-valued quantities. Springer, 1931. [=http://www.cs.utep.edu/interval-comp/young.pdf](http://www.cs.utep.edu/interval-comp/young.pdf). [19](#)