

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

MEMOIRE

Présenté pour l'obtention du diplôme de MASTER

En : Génie électrique _ Spécialité : Automatique

Par : Attoui Atiya

Propagation de contraintes pour l'identification des paramètres d'un bioréacteur

Soutenu publiquement, le //2020, devant le jury composé de :

Dr. Boumediene BENYAHYA

Maitre de conférences, Université de Tlemcen

Président

Dr. Mounira BENALLAL

Maitre de conférences, Université de Tlemcen

Examinatrice

Dr. Salima BORSALI née KAHOUADJI

Maitre de conférences, Université de Tlemcen

Encadreur

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 nous nous intéressons à l'identification des paramètres d'un modèle de bioprocédé aérobie de traitement des eaux usées. 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.

MOTS CLES : Identification, Bioprocédé Aérobie, Arithmétique, biotechnologie

Abstract

The aim of this work is to explore other ways of identifying parameters, applied in the field of biotechnology. This choice is justified for several reasons, such as the various computer bugs caused by rounding errors. The set approach based on interval arithmetic is well suited in these cases, i.e. when the error interval introduced during data acquisition is too large as is the case in biology or science we are interested in identifying the parameters of an aerobic bioprocess model for wastewater treatment. Set methods are well suited for biotechnology applications due to the large error interval introduced during measurement acquisition.

KYE WORDS : identifying, aerobic bioprocess, arithmetic, biotechnology

REMERCIEMENTS

Je tiens à exprimer toute ma reconnaissance à ma directrice de mémoire Madame :

BORSALI née KAHOUADJI SALIMA

Je la remercie de m'avoir encadré, orienté, aidé et conseillé, et surtout ses judicieux conseils, qui ont contribué à alimenter ma réflexion.

Je tiens tout d'abord à remercier les membres du jury, *Dr B.Benyahia* et *Dr.M.Benallel* pour leur présence, pour leur lecture attentive de ma thèse ainsi que pour les remarques qu'ils m'adresseront lors de cette soutenance afin d'améliorer mon travail.

L'enseignement de qualité dispensé par le **Master « M II »** a également su nourrir mes réflexions et a représenté une profonde satisfaction intellectuelle, merci donc aux enseignants chercheurs.

Je remercie également toute l'équipe pédagogique de l'université de TLEMCEM et les intervenants professionnels responsables de ma formation pour avoir assuré la partie théorique de celle-ci.

Je remercie mes très chers parents et ma femme qui ont toujours été là pour moi, je remercie mes sœurs, et mes frères, pour leurs encouragements.

Table des matières	04
Table des figures	06
Introduction Générale	07
Chapitre I L'arithmétique du l'intervalle	
1.1 Introduction	09
1.2 Notions sur L'arithmétique des intervalles	09
1.2.1 Définition	09
1.2.2 Propriétés des intervalles	10
1.2.3 Opérations élémentaires	10
1.2.4 Propriétés algébriques (perdues)	12
1.3 Les pavés	12
1.3.1 Propriétés des pavés	13
1.4 Définition d'un sous pavage	14
1.4.1 Fonction d'inclusion	14
1.4.2 Test d'inclusion	15
1.5 Pessimisme	15
1.6 Conclusion	18
Chapitre II Méthodes ensembliste	
2.1 Introduction	19
2.2 Opérations sur les ensemblistes	19
2.2.1 Ensembles et sous-ensembles	19
2.3 Intersection et réunion	21
2.3.1 Intersection	21
2.3.2 Réunion	21
2.4 Applications	22
2.4.1 Premières définitions	22

2.4.2	Composition d'applications.....	24
2.5	Image directe et réciproque	25
2.5.1	Bijections	26
2.6	Algorithme ensembliste	28
2.6.1	Inversion ensembliste	28
2.6.2	L'algorithme SIVIA	29
2.7	Conclusion.....	30

Chapitre III Les contracteurs

3.1	Introduction	31
3.2	Problèmes de satisfaction de contraintes.....	32
3.2.1	Projection de contraintes	32
3.3	Graphe acyclique orienté DAG.....	33
3.3.1	Avantages de l'utilisation d'un DAG.....	33
3.3.2	Graphe simple ou multigraphes.....	33
3.3.3	Connexité	34
3.4	Les contraintes sous forme DAG.....	35
3.5	Conclusion.....	37

CHAPITRE IV Application au bioréacteur

4.1	Introduction.....	38
4.2	Approche Algébrique.....	38
4.3	Application au bioréacteur.....	39
4.3.1	Mise en équation.....	40
4.3.2	Modèle mathématique.....	40
4.4	Résultats et discussion.....	42
4.5	Conclusion.....	52
4.6	Conclusion générale.....	53

Bibliographie.....	55
---------------------------	-----------

ANNEXE	58
---------------------	-----------

TABLE DES FIGURES

FIGURE 1-1 : Composant de vecteurs intervalles	13
FIGURE 1-2 : Effet d'enveloppement conséquent à la rotation d'un pave	16
FIGURE 2-1 : Illustration de l'inclusion	20
FIGURE 2-2 : Illustration de l'intersection	21
FIGURE 2-3 : Illustration de la réunion	22
FIGURE 2-4 : Image d'un élément	23
FIGURE 2-5 : Antécédents d'un élément	23
FIGURE 2-6 : Antécédents par la fonction carrée.....	24
FIGURE 2-7 : Diagramme sagittal d'une composée.....	25
FIGURE 2-8 : Image directe	25
FIGURE 2-9 : Image réciproque	26
FIGURE 2-10 : Représentation d'une application bijective	27
FIGURE 2-11 : Sivia.....	29
FIGURE 3-1 : Multigraphes, avec une boucle et les liens doubles	34
FIGURE 3-2 : Graphe simple avec 3 composantes connexes	34
FUGIRE 3-3 : Représentation des contraintes sous forme d'arbres	35
FIGURE 3-4 : Représentation des contraintes sous forme de DAG	36
FIGURE 4-1 : Le Bioréacteur	39
FIGURE 4-2 : Les deux étapes de la nitrification.....	41
FIGURE 4-3 : Simulations des mesures enregistrées et du modèle de Nitrification(+).	44
FIGURE 4-41 : DAG relative à expression de k_1	45
FIGURE 4-42 : DAG relative à expression de k_2	46
FIGURE 4-51 : Rendement K_1	48
FIGURE 4-52 : Rendement K_2	48
FIGURE 4-6 : Les DAG relatifs aux expressions de μ_{max1} et μ_{max2}	49
FIGURE 4-71 : Vitesse de croissance maximale μ_{max1}	51
FIGURE 4-72 : Vitesse de croissance maximale μ_{max2}	51

Introduction Générale

L'arithmétique des intervalles est un outil mathématique introduit pour la première fois par MOORE [1]. Il en a eu l'idée au printemps 1958, et une année plus tard il publie un article sur l'arithmétique d'intervalles. Son mérite est, qu' à partir d'un principe simple, donner une méthode générale pour calculer les erreurs de calculs, et aussi pour contrôler la précision des calculs numériques introduits par les erreurs d'arrondi de la représentation flottants des réels. Ses travaux ont été suivis par ceux de NEUMAIER [2] et HANSEN [3], avec une telle arithmétique, il est possible à la fois de tenir compte des incertitudes sur les données et de retourner un encadrement par intervalles ce qui augmente la fiabilité des calculs.

Le principe fondamental de cette arithmétique consiste à remplacer tout nombre réel par un intervalle le contenant puis à effectuer les calculs sur des intervalles, tout intervalle calculé produit un résultat contenant la valeur exacte.

Le premier intérêt de l'arithmétique des intervalles est de pouvoir prendre en compte les incertitudes de mesure dans le cas où les données sont des valeurs expérimentales. Le second intérêt est de permettre de les manipuler sur ordinateur, en utilisant le calcul flottant, des quantités qui ne sont pas exactement représentables comme par exemple le nombre π qui pourra être représenté par l'intervalle $[3.14, 3.15]$ sur une machine qui calcule en basse 10 avec 3 chiffres de mantisse. Cet intervalle contiendra le résultat de calcul contenant la valeur exacte de π . Cette propriété de résultat garantie constitue l'avantage essentiel de l'arithmétique par intervalles.

En résumé, notre travail ici consiste à appliquer d'autres méthodes de génération de données applicables à la biotechnologie. Cette analyse est justifiée par de nombreux motifs, tels que les erreurs de la plupart des différents appareils de calcul en raison de la présence de règles approximatives qui contribuent à l'apparition de certaines erreurs inattendues.

Une approche spécifique basée sur l'analyse d'intervalle est bien adaptée Pour étudier ces systèmes biologiques

Le mémoire est organisé de la manière suivante :

Le premier chapitre est consacré à exposer cette arithmétique des intervalles. Des exemples d'outils faisant intervenir l'analyse par intervalles dans leur méthode sont illustrés, les caractéristiques de ces opérations et les méthodes de calcul sur celles-ci sont illustrées, puis le

résultat de leur utilisation est mentionné dans le but de limiter l'intervalle d'erreur au système à lui appliquer.

Dans le deuxième chapitre, nous allons aborder les fondements de base des méthodes ensembliste [8]. Motivée par sa simplicité dans la manipulation et dans les calculs, la représentation par des intervalles (ou vecteurs intervalles) est choisie pour modéliser les incertitudes influant sur le système biologique donné.

Dans le troisième chapitre, nous avons défini les contractions et comment modélise sous forme des graphes acyclique orienté et la propagation et rétropropagation pour cherche une solution adéquate, et pour résoudre un problème de chémostat anaérobie dans le chapitre suivant,

Dans le dernier chapitre L'une des propriétés les plus importantes des chémostat est que les micro-organismes peuvent se développer physiologiquement stables dans des conditions environnementales constantes. Dans cet état d'équilibre (concentration d'oxygène dissous, concentrations de nutriments et de produits, densité cellulaire, etc.), le chercheur peut contrôler les conditions environnementales. Utiliser l'une des méthodes appropriées pour stabiliser le bioréacteur et trouver des solutions appropriées pour réduire les erreurs de manière appropriée, comme l'application de calculs d'intervalle à l'aide de méthodes ensemblistes

Chapitre I

1 L'arithmétique de l'intervalle

1-1 Introduction

L'arithmétique par intervalles est un système de calcul avec des intervalles plutôt qu'avec des nombres. Comme l'arithmétique des intervalles produit toujours des intervalles contenant l'ensemble de toutes les valeurs de résultat possibles, des algorithmes d'intervalles ont été développés pour effectuer des calculs étonnamment difficiles. Problèmes NP- durs en mathématiques et en informatique, l'arithmétique des intervalles est une méthode de calcul consistant à manipuler des intervalles, par opposition à des nombres (par exemple entiers ou flottants), dans le but d'obtenir des résultats plus rigoureux. Cette approche permet de borner les erreurs d'arrondi et ainsi de développer des méthodes numériques qui fournissent des résultats fiables. L'arithmétique des intervalles est une branche de l'arithmétique des ordinateurs.

Dans l'arithmétique des intervalles, un nombre réel x est représenté par une paire de nombres flottants $(x_{\text{inf}}, x_{\text{sup}})$. Dire que x est représenté par cette paire signifie que x appartient à l'intervalle $[x_{\text{inf}}, x_{\text{sup}}]$, autrement dit que les assertions $x_{\text{inf}} \leq x$ et $x \leq x_{\text{sup}}$ sont vraies. La notion d'égalité d'un nombre et de sa représentation disparaît, sauf si $x_{\text{inf}} = x_{\text{sup}}$.

On appelle *largeur* de la représentation la quantité $w(x) = x_{\text{sup}} - x_{\text{inf}}$, qui mesure l'incertitude de la représentation.

1.2 Notions sur L'arithmétique des intervalles

1.2.1 Définition

Un intervalle x est un ensemble connexe, borné, fermé de borne inférieure \underline{x} de borne supérieure \bar{x} qui sont des nombres réels, l'intervalle $[x]$ est représenté de manière suivante :

$$[x] = [\underline{x}, \bar{x}] := \{\forall x \in \mathbb{R}, \underline{x} \leq x \leq \bar{x}\} \dots \dots \dots (1 - 1)$$

L'ensemble de tous les intervalles \mathbb{R} est noté \mathbb{IR} , soit $[x] \in \mathbb{IR}$

1.2.2 Propriétés des intervalles

Tout nombre réel x sera confondu avec l'intervalle $[x, \bar{x}]$ correspondant. Les intervalles seront désormais repérés par des caractères majuscules et les caractères minuscules désigneront des nombres réels ou entiers.

$$\{X = [1,3] \text{ ou } = \left[\frac{\pi}{2}, \frac{3\pi}{2} \right] \} \quad \dots \dots \dots (1 - 2)$$

- a) *Borne inférieure de* $[X]$: $\inf([X]) = \underline{x}$
- b) *Borne supérieure de* $[X]$: $\sup([X]) = \bar{x}$
- c) *Milieu d'intervalle de* $[X]$: fonction définie de \mathbb{I} dans \mathbb{IR}

$$\left\{ \begin{array}{l} \text{mid}[X] = \frac{\bar{x} + \underline{x}}{2}, \quad \forall X \in \mathbb{IR} \\ \text{par extention aux forme vectorielle} \\ \text{mid}([X]) = (\text{mid}([x_1]), \dots \dots \dots, \text{mid}([x_n]))^T \quad X \in \mathbb{IR}^n \end{array} \right. \quad (1 - 3)$$

- d) *Largeur d'intervalle* $([X])$: Fonction définie de \mathbb{I} dans \mathbb{IR} ou dans le cas vectoriel de \mathbb{I}^n dans \mathbb{IR}^n

$$\left\{ \begin{array}{l} w([X]) = \bar{x} - \underline{x}, \quad \forall X \in \mathbb{IR} \\ \text{par extention aux forme vectorielle} \\ w([X]) = (w([x_1]), \dots \dots \dots, w([x_n]))^T \quad X \in \mathbb{IR}^n \end{array} \right. \quad (1 - 4)$$

- e) *Rayon de l'intervalle* $([X])$: $\mathbf{rad}([X]) = \frac{\bar{x} - \underline{x}}{2}, \forall x \in \mathbb{IR} \quad (1 - 5)$

1.2.3 Opérations élémentaires

Les opérations mathématiques élémentaires sont étendues aux intervalles, le résultat d'une opération entre deux intervalles obtenus en travaillant seulement sur leurs bornes.

Soit $[x], [y] \in \mathbb{IR}$, et une opération binaire $\odot \in \{+, -, *, /\}$

On définit : $[x] \odot [y] = \{ x \odot y \mid x \in [X] \mid y \in [Y] \}$

Définition : le résultat d'une opération \odot entre deux intervalles $\begin{cases} [X] = [\underline{x}, \bar{x}] \\ [Y] = [\underline{y}, \bar{y}] \end{cases}$ (1-6)

• $[X] + [Y] = [\underline{x} + \underline{y}, \bar{x} + \bar{y}]$ (1-61)

• $[X] - [Y] = [\underline{x} - \bar{y}, \bar{x} - \underline{y}]$ (1-62)

• $[X] * [Y] = [\min(\underline{x} * \underline{y}, \underline{x} * \bar{y}, \bar{x} * \underline{y}, \bar{x} * \bar{y}), \max(\underline{x} * \underline{y}, \underline{x} * \bar{y}, \bar{x} * \underline{y}, \bar{x} * \bar{y})]$ (1-63)

• $[X]^2 = \begin{cases} [\min(\underline{x}^2, \bar{x}^2), \max(\text{idem})] \text{ si } 0 \notin [X] \\ [0, \max(\underline{x}^2, \bar{x}^2)] \text{ si } 0 \in [X] \end{cases}$ (1-64)

• $\left[\frac{1}{X}\right] = \left[\min\left(\frac{1}{\underline{x}}, \frac{1}{\bar{x}}\right), \max(\text{idem}) \text{ si } 0 \notin [X]\right]$ (1-65)

• $\left[\frac{X}{Y}\right] = [\underline{x}, \bar{x}] * \frac{1}{[\underline{y}, \bar{y}]} \text{ si } 0 \notin [Y]$ (1-66)

• $e^{[X]} = [e^{\underline{x}}, e^{\bar{x}}]$ (1-67)

Exemple :

Expression polynomiale :

$$\begin{cases} fx = x^3 - 2x^2 + x - 3, \\ x = [-5, 2] \end{cases} \quad (1-7)$$

$$\begin{aligned} fx &= [-5, 2]^3 - 2[-5, 2]^2 + [-5, 2] - 3 \\ &= [-125, 8] - 2[0, 25] + [-5, 2] - 3 \\ &= [-125, 8] - [0, 50] + [-5, 2] - 3 \\ &= [-183, 7] \end{aligned}$$

Démonstration :

$$\begin{aligned} &[-5, 2]^3 \neq [-5, 2]^2 * [-5, 2]^1 \\ &[-5, 2]^3 = [-5, 2]^n \quad \text{tel que } n = 3 \end{aligned}$$

En effet :

$$X^n = \begin{cases} [1, 1] \text{ si } n = 0 \\ \begin{cases} [(\underline{x})^n, (\bar{x})^n] \text{ si } \underline{x} \geq 0 \text{ ou } n \text{ impair} \\ [(\bar{x})^n, (\underline{x})^n] \text{ si } \bar{x} \leq 0 \text{ et } n \text{ pair} \end{cases} \\ [0, \max((\underline{x})^n, (\bar{x})^n)] \text{ si } 0 \in [X] \text{ et } n \text{ pair} \end{cases}$$

1.2.4 Propriétés algébriques (perdues)

- la soustraction n'est pas la réciproque de l'addition

$$\text{si } x = [2,3] \text{ et } y = [-3,5] \Rightarrow \begin{cases} z = (x + y) \\ = [-1,8] \end{cases}$$

$$\begin{cases} (z - y) = [-1,8] - [-3,5] = [-6,11] \\ x = [2,3] \in (z - y) = [-6,11] \end{cases}$$

Ou encore :

$$(x - x) = [2,3] - [2,3] = [-1,1] \neq 0$$

- La division n'est pas la réciproque de la multiplication

$$\text{six } x = [2,3] \Rightarrow \frac{x}{x} = \frac{[2,3]}{[2,3]} = \left[\frac{2}{3}, \frac{3}{2}\right] \ni 1$$

- La multiplication n'est pas égale à l'élevation carrée.

$$\text{six } x = [-3,2] \begin{cases} [x] * [x] = [-6,9] \\ [x]^2 = [0,9] \end{cases}$$

- La multiplication est sous-distributive par rapport à l'addition.

$$\text{si } \begin{cases} x = [-2,3], y = [1,4], z = [-2,1] \\ x * (y + z) \neq (x * y) + (x * z) \end{cases}$$

1.3 Les pavés

Soit $\mathbb{I}\mathbb{R}^n$ l'ensemble des intervalles de \mathbb{R}^n un pavé $[x]$ est un vecteur intervalle de dimension(n) représenté de la manière suivante :

$$([X]) = ([\underline{x}_1, \bar{x}_1] \dots \dots \dots [\underline{x}_n, \bar{x}_n]) \dots \dots \dots (1 - 8)$$

Les opérateurs élémentaires définis précédemment pour les intervalles sont également redéfinies pour les vecteurs d'intervalles.

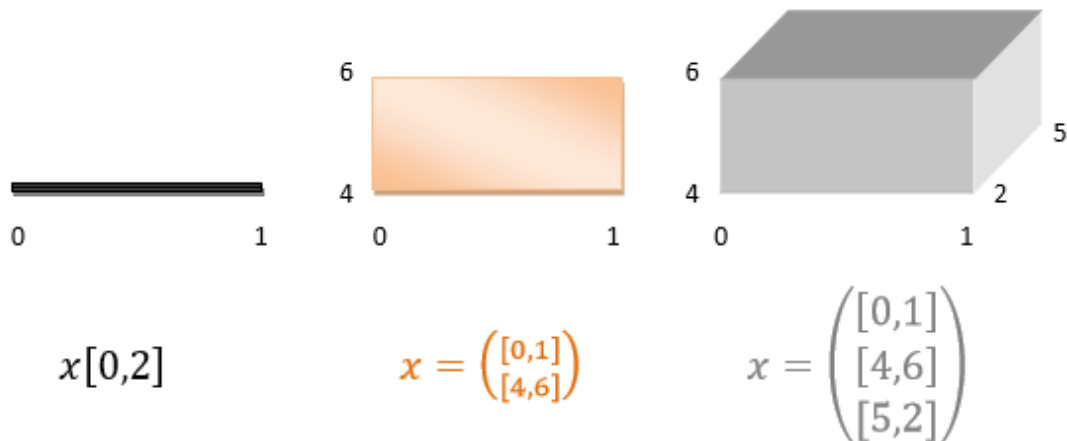


FIGURE 1-1 : Composant de vecteurs intervalles

1.3.1 Propriétés des pavés

- *Borne inférieure du pavé : $[\underline{\mathbf{x}}]$*

$$\inf([X]) = (\underline{x}_1, \underline{x}_2, \dots, \dots, \dots, \underline{x}_n)^T T: \text{transpose}$$

- *Borne supérieure du pavé : $[\overline{\mathbf{x}}]$*

$$\sup([X]) = (\overline{x}_1, \overline{x}_2, \dots, \dots, \dots, \overline{x}_n)^T T: \text{transpose}$$

- *La largeur du pavé : $[X]$*

$$w[X] = \max_{j=1}^n (\overline{x}_j - \underline{x}_j) \geq 0$$

Exemple : soit le pavé $[x] = ([1,2] * [-1,3])$

$$w[X] \begin{cases} = \max_{j=1}^2 (\overline{x}_j - \underline{x}_j) \geq 0 \\ = \max((2 - 1), (3 - (-1))) \\ = \max(1, 4) \\ = 4 \end{cases}$$

- *Le milieu du pavé : $[\mathbf{x}]$*

$$\text{mid}([X]) = (\text{mid}[x_1], \dots, \dots, \dots, \text{mid}[x_n])^T$$

- *Le volume du pavé : $[\mathbf{x}]$*

$$\text{vol}[X] = ((\overline{x}_1 - \underline{x}_1) * (\overline{x}_2 - \underline{x}_2) * \dots * (\overline{x}_n - \underline{x}_n))$$

$$vol[x] = \prod_{i=1}^n [(\underline{x}_i - \bar{x}_i)] \dots\dots\dots (1 - 9)$$

NB1 : La largeur de l'ensemble vide : $\phi : w(\phi) = -\infty$

NB2 : Un intervalle dont les deux bornes sont égales est dit dégénéré

$$\underline{x} = x = \bar{x}$$

1.4 Définition d'un sous pavage

Un sous pavage $x \in \mathbb{R}^n$ est une union de pavé : $([x])_{i=1}^n$ dont les intérieurs sont disjoints. Lorsque un sous pavage \hat{P} recouvre tout le pavé de $[x]$, 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.

1.4.1 Fonction d'inclusion

Considérons deux ensembles ainsi que deux récipients $[x \& y]$, contenant les deux ensemble $x \& y$ respectivement, soit un sous ensemble x_1 de x si $x_1 \in X$, on peut étendre une fonction $F : X \Rightarrow Y$ Aux récipients tel que :

$$[F](X_1) = [\{F(x) | x \in X_1\}]$$

Et encore $[F](X_1) \supset F(X_1)$ (1 - 10)

Ce qui introduit la notion de fonction d'inclusion, considérons une fonction \mathcal{F} de $\mathbb{R}^n \Rightarrow \mathbb{R}^m$, la fonction intervalle $[F]$ de $\mathbb{I}\mathbb{R}^n \Rightarrow \mathbb{I}\mathbb{R}^m$ est une fonction d'inclusion

$$F \text{ si } \begin{cases} \forall [x] \in \mathbb{R}^n, F[x] \subseteq [F]([x]) \\ \text{ou} \\ [F]([x]) \supset (\{F(x) | x \in [x]\}) \end{cases} \quad (1 - 11)$$

Propriétés

- La convergence : Une fonction d'inclusion $[F]$ d'une fonction F est convergente si pour une suite de pavé $[x]_k$

$$\lim_{i \rightarrow \infty} w([x]_k) = 0 \Rightarrow \lim_{i \rightarrow \infty} w([F][x]_k) = 0 \quad (1 - 12)$$

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

$$[x] \subset [y] \Rightarrow [F]([x]) \in [F]([y]) \quad (1-13)$$

1.4.2 Test d'inclusion

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

$$IB = \{\emptyset, [0], [1], \dots, [0,1]\}$$

Où l'élément $\begin{cases} [\Phi]: \text{pour impossible} \\ [1]: \text{pour vrai} \end{cases} \begin{cases} [0]: \text{pour faux} \\ [0,1]: \text{pour indéterminé} \end{cases} \quad (1-14)$

On définit une fonction test $t: \mathbb{R}^n \rightarrow B$ avec $B = \{[0], [1]\}$

Comme pour les réels, la notion de fonction $[t]: \mathbb{IR}^n \rightarrow IB$

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

$$\begin{cases} [t]([x]) = 1 \Rightarrow \forall x \in [x]: t(x) = 1 \\ [t]([x]) = 0 \Rightarrow \forall x \in [x]: t(x) = 0 \end{cases} \quad (1-15)$$

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

$$\begin{cases} (\mathbf{x}_1, \mathbf{x}_2)^t \rightarrow (x_1 + x_2 \leq 10) \\ \text{ce qui veut dire :} \\ \mathbf{t}(\mathbf{x}) = \begin{cases} 1 : \text{si } (x_1 + x_2) \leq 10 \\ 0 : \text{si } (x_1 + x_2) > 10 \end{cases} \end{cases}$$

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

$$[t](x) = \begin{cases} [1] : & \text{si } (x_1 + x_2) \leq 10, \\ [0] : & \text{si } (x_1 + x_2) > 10, \\ [0,1] : & \text{ailleurs.} \end{cases}$$

On peut réécrire aussi : $[t](x) \Leftrightarrow ([x_1], [x_2]) \leq 10 \quad (1-16)$

1.5 Pessimisme

Lorsque le résultat des opérations entre intervalles n'est pas minimal, on parle alors de pessimisme, le pessimisme est une conséquence de deux phénomènes, le premier et le phénomène d'enveloppement, il provoque une surestimation du pavé résultant, l'exemple suivant permet de mieux comprendre cet effet :

Exemple :

Soit le pavé $\mathbf{x} = [\mathbf{x}_1] \times [\mathbf{x}_2]$, si l'on fait subir à ce pavé des rotations successives à l'aide de la matrice \mathbf{A} définie par :

$$\Rightarrow \mathbf{A} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \text{ avec } \theta = \frac{\pi}{4} \quad (1 - 23)$$

La représentation de ce pavé image est un rectangle qui l'enveloppe, Après plusieurs rotations, ce rectangle augmente, ce qui provoque une surestimation du résultat finale.

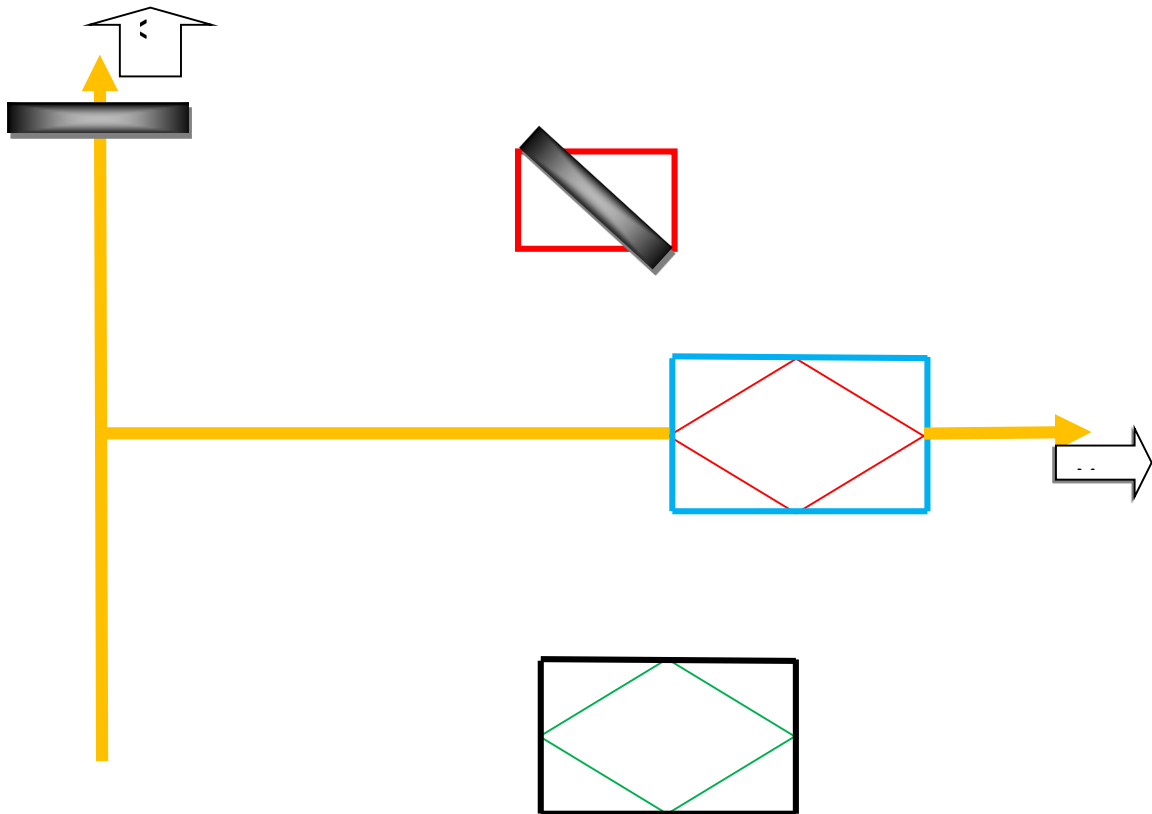


FIGURE 1-2 : Effet d'enveloppement conséquent à la rotation d'un pave

Le deuxième phénomène est celui de dépendance appelé aussi 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 : $X - Y = \{x - y \text{ , } x \in X | y \in Y\}$ est différent de l'ensemble $\{x - x | x \in X\} = 0$

Exemple: *soit* $[x] = [-1,1]$ *alors* :

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

1.6 Conclusion

Dans cette section, nous nous sommes concentrés sur la mention de certaines des opérations arithmétiques qui est une méthode très sûre et efficace pour l'évaluation de mesures qui sont souvent en tachées d'erreurs. 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. Nous souhaiterions en l'appliquant à notre modèle fortement non linéaire d'obtenir les résultats garantis.

Chapitre II

2 Méthode ensembliste

2.1 Introduction

Pour obtenir un résultat garanti de la solution d'un problème donné, il faut mettre en œuvre une méthode qui prend en considération les incertitudes omniprésentes sur les données d'un problème. Dans ce chapitre, nous allons présenter et confronter une approche qui prend en compte ces incertitudes qui sont l'approche ensembliste.

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

Cette partie sera consacrée à la présentation des diverses opérations que l'on peut effectuer sur des ensembles, en particulier l'intersection, la réunion et le passage au complémentaire

2.2.1 Ensembles et sous-ensembles

Les objets qui constituent un ensemble seront appelés les éléments de cet ensemble. En général, nous désignerons les ensembles avec des majuscules et leurs éléments avec des minuscules.

Notation : Pour signifier que x est un élément d'un ensemble E , nous noterons $x \in E$. Le symbole \in le sigle d'appartenance. Au contraire, si x n'est pas un élément d'un ensemble E , nous noterons $x \notin E$.

Il y a deux moyens d'écrire les ensembles, en extension et en compréhension.

Exemple 2.1. Ensemble écrit en extension

Ensemble des cinq continents : {Afrique, Amérique, Europe, Asie, Océanie}.

Dans le cas d'un ensemble infini on pourra avoir recours à des points de suspension. Par exemple, l'ensemble des entiers naturels peut s'écrire $\{0, 1, 2 \dots\}$

Exemple 2.2. Ensemble écrit en compréhension

Un ensemble composé de réels vérifiant une certaine inégalité : $x \in \mathbb{R}, x^2 \geq 8$
 $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ Et $\{n \in \mathbb{N}, 0 \leq n \leq 9\}$ désignent le même ensemble.

Nous allons maintenant définir quelques ensembles très particuliers.

Définitions : L'ensemble vide est un ensemble qui ne contient aucun élément. Il est unique et est noté \emptyset .

Un singleton est un ensemble constitué d'un seul élément, et donc de la forme $E = \{x\}$. On prendra soin de distinguer l'élément x de l'ensemble $\{x\}$.

Une paire est un ensemble constitué de deux éléments, et donc de la forme $E = \{x, y\}$.

En guise d'exemples, présentons les ensembles les plus couramment utilisés en mathématiques élémentaires.

Exemple 2.3. Les ensembles numériques usuels

Ensemble des entiers naturels : $\mathbb{N} = \{0, 1, 2, 3, \dots\}$.

Ensemble des entiers relatifs : $\mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$.

Ensemble des rationnels : $\mathbb{Q} = \frac{p}{q}$

$Q, p \in \mathbb{Z}, q \in \mathbb{N}, q \neq 0$.

Ensemble des nombres réels : \mathbb{R} .

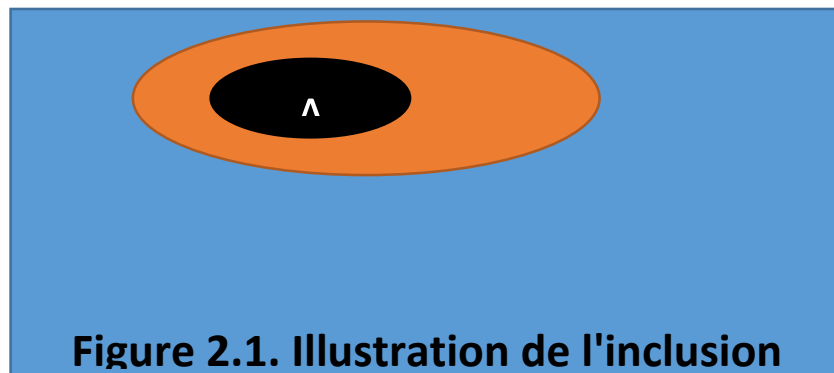
Ensemble des nombres complexes : \mathbb{C} .

Définition : On dit qu'un ensemble A est inclus dans un ensemble B si tous les éléments de A sont éléments de B . On note alors $A \subset B$.

On peut aussi utiliser les formulations A est un sous-ensemble de B ou encore A est une partie de B .

Avec les notations du premier chapitre, l'inclusion peut également s'écrire comme une **implication** :

$$A \subset B \Leftrightarrow (\forall x, x \in A \Rightarrow x \in B) \quad (2-1)$$



Propriétés

1. Pour tous ensemble A , on a $A \subset A$ et $\emptyset \subset A$.
2. Soient A et B deux ensembles.

$$\text{On a } A = B \Leftrightarrow ((A \subset B) \wedge (B \subset A)) \quad (2-2)$$

Cette égalité justifie la méthode dite de double inclusion : pour montrer que deux ensembles sont égaux, on montre que le premier est inclus dans le second et que réciproquement le second est inclus dans le premier.

2.3 Intersection, réunion

Cette sous-partie sera consacrée à l'étude de deux opérations binaires portant sur les ensembles, l'intersection et la réunion.

2.3.1 Intersection

Définition : On appelle intersection de deux ensembles A et B , l'ensemble noté $A \cap B$ constitué des éléments qui appartiennent à A et à B .

Avec les notations du premier chapitre, l'intersection peut également s'écrire comme une conjonction :

$$A \cap B = \{x, (x \in A) \wedge (x \in B)\} \quad (2-3)$$

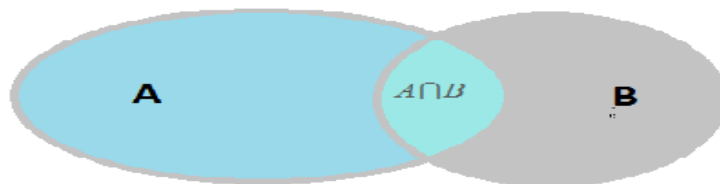


Figure 2.2. Illustration de l'intersection

2.3.2 Réunion

On appelle réunion de deux ensembles A et B l'ensemble noté $A \cup B$ constitué des éléments qui appartiennent à A ou à B .

Avec les notations du premier chapitre, la réunion peut également s'écrire comme une disjonction :

$$A \cup B = \{x, (x \in A) \vee (x \in B)\} \quad (2-4)$$

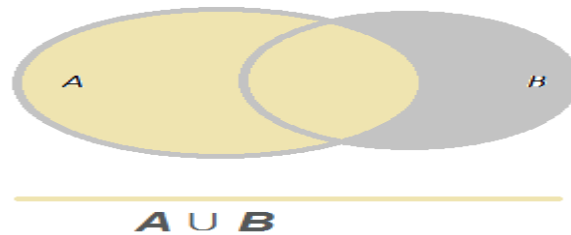


Figure 2.3. Illustration de la réunion

Voici quelques propriétés vérifiées par l'opération de réunion.

Propriétés

Soient A , B et C des ensembles.

On a :

$A \cup B = B \cup A$, la réunion est commutative.

$(A \cup B) \cup C = A \cup (B \cup C)$, la réunion est associative.

$A \subset (A \cup B)$ et $B \subset (A \cup B)$. (2-5)

$A \cup A = A$.

$A \cup \emptyset = A$, \emptyset est un élément neutre pour la réunion.

Si $A \subset B$, alors $A \cup B = B$.

Un élément neutre pour une opération binaire est défini par le fait que le résultat de l'opération entre l'élément neutre et n'importe quel élément est ce dernier. En quelque sorte il n'influe pas sur le résultat de l'opération.

Comme par exemple 1 pour la multiplication des réels et donc l'ensemble vide pour la réunion.

2.4 Applications

Nous allons dans cette seconde partie définir la notion d'application, puis nous intéresser aux cas particuliers des surjections, Injections et bijections.

2.4.1 Premières définitions

Soient E et F deux ensembles.

Définir une application f de E dans F consiste à associer à tout élément x de E un élément unique y de F .

Cet élément y , noté $f(x)$, est appelé l'image de x par f .

L'élément x est un antécédent de y par f .

E est l'ensemble de départ de f et F en est l'ensemble d'arrivée.

Comme lors de la première partie, nous réaliserons régulièrement des schémas afin d'illustrer les concepts nouvellement définis.

Pour représenter une application, le schéma utilisé s'appelle un diagramme sagittal.

En voici un premier à propos de la notion d'**image**.

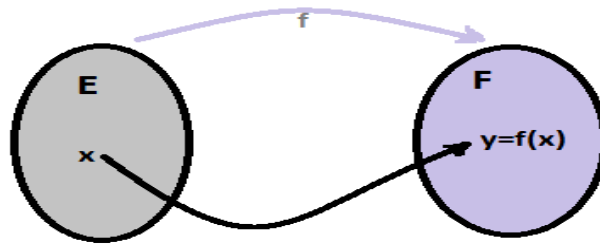


Figure 2.4. Image d'un élément

Ici, y est l'image de x par f , $y=f(x)$.

Bien comprendre que la définition précédente implique que tout élément de l'espace de départ possède une image et que celle-ci est unique.

Et en voici un second pour bien visualiser la définition d'antécédent.

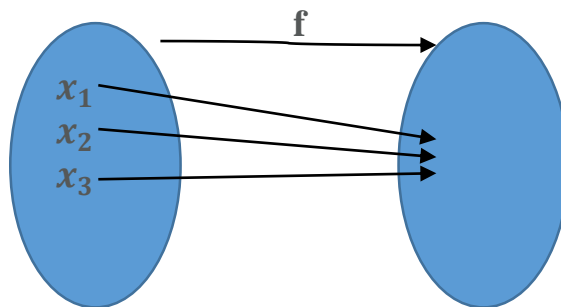


Figure 2.5. Antécédents d'un élément

Dans ce cas de figure, x_1 , x_2 et x_3 sont des antécédents de y par f ,

$$f x_1 = f x_2 = f x_3 = y.$$

Bien comprendre que la définition précédente implique que tout élément de l'espace d'arrivée peut avoir zéro ou plusieurs antécédents.

Pour notre premier exemple concret, nous allons utiliser la très connue fonction carré.

Exemple 2.4. Antécédents par la fonction carrée

Considérons l'application f de \mathbb{R} dans \mathbb{R} définie par $f(x) = x^2$.

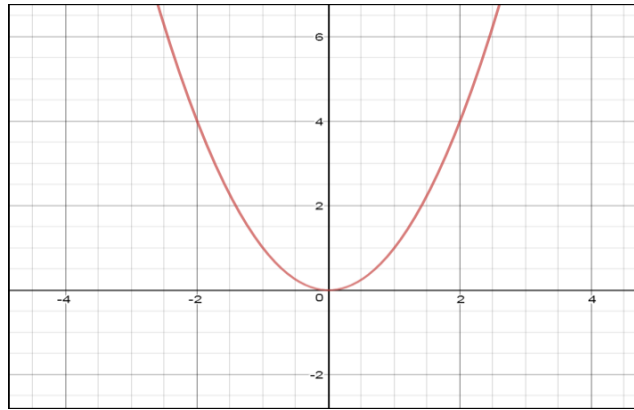


Figure 2.6 : Antécédents par la fonction carrée

On cherche à déterminer les antécédents de n'importe quel réel par f . Pour cela on doit distinguer les trois cas de figure suivants :

Tout réel strictement négatif n'admettra aucun antécédent dans \mathbb{R} par f .

0 admet un unique antécédent.

Tout réel strictement positif admet deux antécédents dans \mathbb{R} par f .

La définition suivante est celle d'une application bien particulière, elle laisse en effet les éléments d'un ensemble inchangés.

Définition : Soit E un ensemble l'application de E dans E qui a un élément x associe x est appelée application identité de E . On la note généralement Id_E .

Par définition, on a donc $\forall x \in E, \text{Id}_E(x) = x$

2.4.2 Composition d'applications

Il est tout à fait possible d'enchaîner les applications, c'est-à-dire en appliquer une seconde aux images provenant d'une première.

Définition :

Soient E, F et G trois ensembles, f une application de E dans F et g une application de F dans G l'application de E dans G notée $g \circ f$ et définie par

$\forall x \in E, (g \circ f)(x) = g(f(x))$ est appelée composée de f par g .

➤ A l'oral on prononce "g rond f".

Représentons cette opération de composition par un diagramme sagittal.

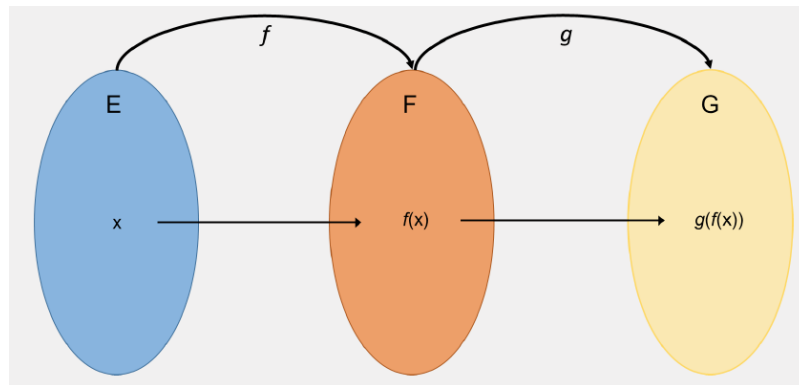


Figure 2.7. Diagramme sagittal d'une composée

2.5 Image directe et image réciproque

On va dans cette sous-partie donner des définitions ne portant plus seulement sur des éléments, mais sur des sous-ensembles des espaces de départ et d'arrivée. Plus précisément, on va voir comment ils sont transformés par l'application.

Définitions :

Soient E et F deux ensembles, et f une application de E dans F .

Soit A un sous-ensemble de E et B un sous-ensemble de F .

On appelle image directe de A par f l'ensemble noté $f(A)$ des images par f des éléments de A . Il s'agit donc d'un sous-ensemble de F dont voici l'expression littérale

$$f(A) = \{y \in F, \exists x \in A, y = f(x)\} = \{f(x), x \in A\} \quad (2-6)$$

On appelle image réciproque de B par f l'ensemble noté $f^{-1}(B)$ des antécédents par f des éléments de B . Il s'agit donc d'un sous-ensemble de E dont voici l'expression littérale

$$f^{-1}(B) = \{x \in E, f(x) \in B\} \quad (2-7)$$

Des diagrammes sagittaux nous permettront de mieux appréhender ces deux définitions un peu plus complexes que les précédentes.

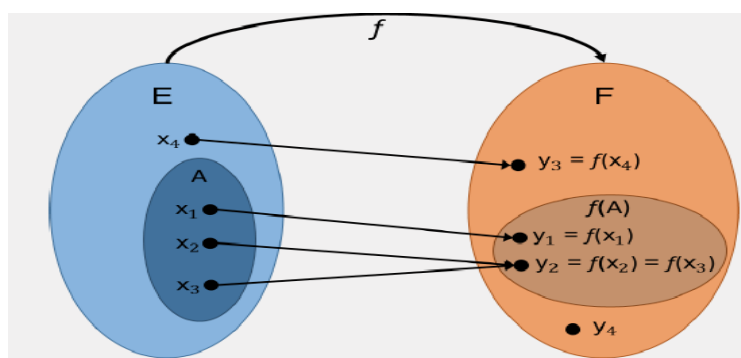


Figure 2.8. Image directe

L'image directe par f du sous-ensemble A de E constitué des éléments x_1 , x_2 et x_3 est le sous-ensemble de F composé des images de ces trois éléments, y_1 et y_2 .

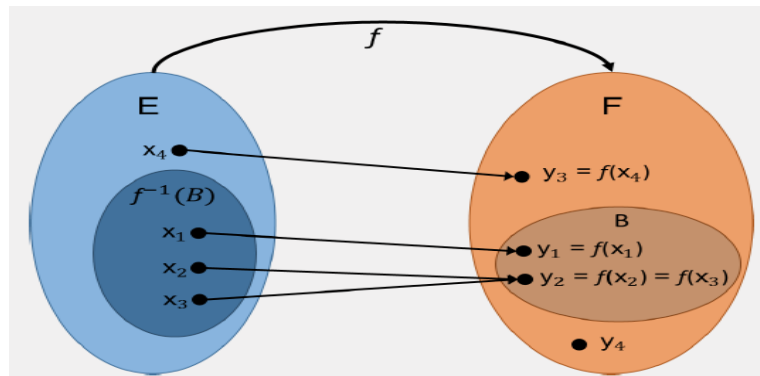


Figure 2.9. Image réciproque

L'image réciproque par f du sous-ensemble B de F constitué des éléments y_1 et y_2 est le sous-ensemble de E composé des antécédents de ces deux éléments, x_1 , x_2 et x_3 .

Utilisons de nouveau la fonction carrée comme exemple.

Exemple 2.5 :

Images directes et images réciproques par la fonction carrée

Considérons l'application f de \mathbb{R} dans \mathbb{R} définie par $f(x) = x^2$.

Nous allons exprimer quelques images directes et réciproques de sous-ensembles de \mathbb{R} par f . Que le lecteur prenne bien le temps de comprendre chaque cas de figure, en s'aidant si besoin est de la représentation graphique de la fonction carré rappelée dans l'exemple 2.4.

$$f([1; 2]) = [1; 4].$$

$$f([-4; -3]) = [9; 16].$$

$$f([-1; 2]) = [0; 4].$$

$$f^{-1}([0; 4]) = [-2; 2].$$

$$f^{-1}([1; 4]) = [-2; -1] \cup [1; 2].$$

$$f^{-1}([-12; -4]) = \emptyset.$$

$$f^{-1}([-3; 16]) = [-4; 4].$$

2.5.1 Bijections

La notion de bijection va regrouper celles de surjection et d'injection.

Définition :

Soient E et F deux ensembles, et f une application de E dans F .

On dit que f est bijective si tout élément de l'ensemble d'arrivée admet un antécédent et un seul par f .

On dit aussi que f réalise une bijection de E sur F .

Réalisons le diagramme sagittal.

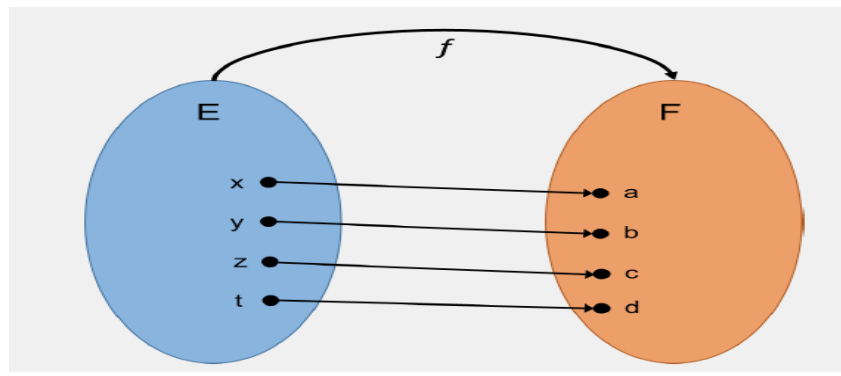


Figure 2.10. Représentation d'une application bijective

L'application f est bijective car tous les éléments de l'ensemble d'arrivée ont exactement un antécédent par f .

Comme nous en avons pris l'habitude, utilisons la fonction carrée comme exemple.

Exemple 2.6 : Bijective de la fonction carrée

L'application f de \mathbb{R} dans \mathbb{R}_+ définie par $f(x) = x^2$ n'est pas bijective car l'élément 4 de l'espace d'arrivée admet deux antécédents par f , à savoir - 2 et 2.

L'application g de \mathbb{R}_+ dans \mathbb{R}_+ définie par $f(x) = x^2$ est par contre bijective.

Obtenir la petite caractérisation suivante d'une application bijective est très simple.

Propriété.

En pratique, on se sert toujours de l'assertion précédente pour montrer qu'une application est bijective.

Si l'on dispose d'une application bijective f , on peut très bien imaginer une application qui à tout élément de l'ensemble d'arrivée associe son unique antécédent par f . Cette remarque conduit à la définition suivante.

Définition :

Soient E et F deux ensembles, et f une application bijective de E dans F .

On appelle application réciproque de f , notée f^{-1} , l'application de F dans E définie pour tout y de F par

$$x = f^{-1}(y) \Leftrightarrow (y = f(x)) \quad (2-8)$$

Le terme inverse est parfois employé à la place de réciproque mais il s'agit d'une incorrection. L'application réciproque d'une application bijective f réalise donc en quelque sorte le travail opposé à celui de f . Il n'est donc pas surprenant que leur composition soit l'application identité.

Propriété

Soient E et F deux ensembles, f une application bijective de E dans F et f^{-1} son application réciproque.

On a
$$f^{-1} \circ f = Id_E \quad \text{et} \quad f \circ f^{-1} = Id_F \quad (2-9)$$

Reprenons une dernière fois la fonction carrée en exemple.

Exemple 2.7. Réciproque de la fonction carrée

L'application g de \mathbb{R}_+ dans \mathbb{R}_+ définie par $f(x) = x^2$ est bijective et son application réciproque est l'application g^{-1} de \mathbb{R}_+ dans

\mathbb{R}_+ définie par $g^{-1} = \sqrt{x}$.

2.6 ALGORITHME ENSEMBLISTE

Le calcul ensembliste est une notion un peu idéale car on ne sait pas vraiment calculer avec les ensembles. Dans nos applications, nous développerons des algorithmes ensemblistes qui utiliseront le calcul ensembliste et qui est donc, eux aussi, seront idéaux.

2.6.1 Inversion ensembliste

Considérons un ensemble dans R^n et une fonction $f: R^n, R^m$. Soit Y un sous ensemble de R^m défini comme suit : $X = f(Y)$

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

$$S = \{x \in R^n | f(x) \in Y\} = f^{-1}(Y) \quad (2 - 10)$$

On peut résoudre un problème d'inversion ensembliste par le biais de l'algorithme SIVIA (Set Inversion Via Interval Analysis) [11]

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 :

$$\underline{S} < S < \bar{S} \quad (2 - 11)$$

L'approximation, ou encadrement, intérieure \underline{S} représente l'ensemble des pavés dits acceptables ou faisables. C'est l'ensemble contenant toutes les solutions recherchées. Cependant, il peut y avoir des solutions admissibles qui ne sont pas contenues dans \underline{S} . Si on démontre que $[x] \cap \bar{S} = \emptyset$; 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 long du plus grand coté 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.

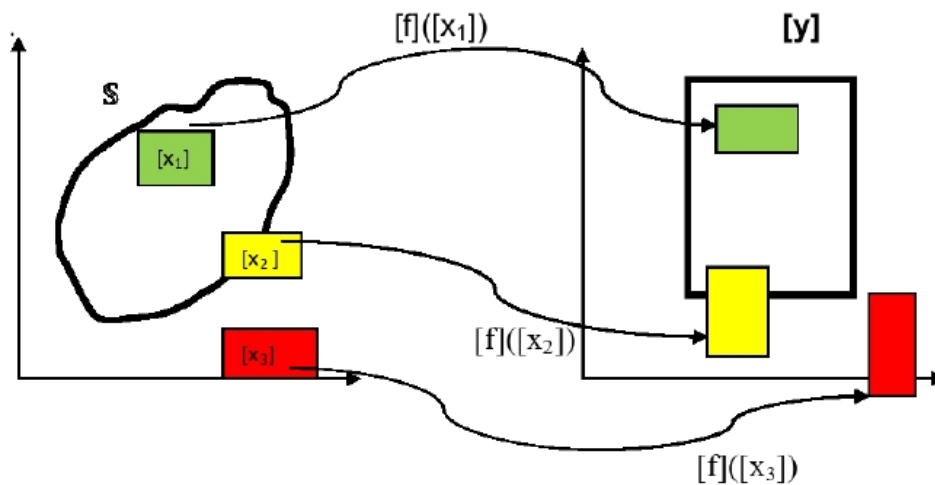


Figure 2-11 Sivia

La description de l'algorithme Sivia est donc :

2.6.2 L'algorithme SIVIA

$$\left\{ \begin{array}{l} \mathbf{SIVIA} (\text{entrée} : [t], [x], \epsilon; \text{sorties} \underline{S}, \overline{S}), \\ \quad \mathbf{1.} \text{ si } [t]([x]) = [0], \text{ rejeter } [x]; \\ \quad \mathbf{2.} \text{ si } [t]([x]) = [1], \underline{S} := \underline{S} \cup [x], \overline{S} := \overline{S} \cup [x]; \\ \quad \mathbf{3.} \text{ si } w([x]) \leq \epsilon, \overline{S} := \overline{S} \cup [x]; \\ \quad \mathbf{4.} \text{ bissecter } [x] \text{ en } ([x_1], [x_2]) \\ \quad \quad \mathbf{SIVIA}(e: [t], x_1, \epsilon; s: \underline{S}, \overline{S}); \\ \quad \quad \mathbf{SIVIA}(e: [t], x_2, \epsilon; s: \underline{S}, \overline{S}); \end{array} \right. \quad (2 - 12)$$

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 [28 ; 29] permettant de réduire les domaines solutions.

2.7 CONCLUSION

En général, on considère que la valeur moyenne des résidus dans le cas d'un bon fonctionnement est proche de zéro (le plus proche possible du fonctionnement théorique

Ce type d'estimation, avec une technique de sous pavages, a été vu comme un problème d'inversion ensembliste [Jaulin \[\]](#). Une recherche vaste et approfondie a été faite sur le sujet, ce qui a donné suite à [Jaulin \[\]](#), dans le cadre des observateurs ensemblistes

Ce chapitre a été consacré en premier lieu à relever les motivations de l'utilisation de les méthodes ensembliste. Celles-ci sont dues principalement à la capacité de mieux prendre en compte les incertitudes sur les données d'un problème et sur les méthodes et les moyens de résolution.

Chapitre III

2 Les contracteurs

3.1 Introduction

Les techniques de réduction des domaines intervalles ont des propriétés communes qu'il est intéressant de regrouper dans la notion de contracteur [6], le principe est d'augmenter le temps de traitement d'un pavé pour diminuer le temps effectif de résolution des problèmes. Cette diminution du temps effectif de résolution est très liée au problème. Dans le pire des cas l'ajout de ces contracteurs diminuent l'efficacité de l'algorithme mais dans certains problèmes l'efficacité est augmentée. La notion de contracteur a l'avantage de regrouper un très large éventail de méthodes de réduction des domaines intervalles. Les exemples pour les CSP linéaires sont le contracteur de Gauss Seidel ou le contracteur par programmation linéaire. Pour les CSP non-linéaires, le contracteur de Newton agit sur des systèmes uniquement carrés et différentiables, le contracteur par linéarisation parallèle sur des systèmes rectangulaires et le contracteur par projection de contraintes n'a pas de restriction sur la forme du CSP. Dans la suite de l'étude seule le contracteur utilisant la propagation de contraintes sera détaillée pour suivre ce qui a été codé durant la thèse. Plus de détails sur les autres types de contracteurs peuvent être trouvés dans [7] et [8].

L'outil de contraction ou de réduction est apparu à cause de la difficulté rencontrée lors des opérations de bisection d'un pavé, par exemple dans l'algorithme SIVIA pour un nombre de direction $N = 20$, on aura à faire $2^{20} = 1048576$ bisections. Cet outil permet de réduire le nombre de bisections à effectuer. L'une des applications conçue pour cette technique est « INTERVAL PEELER » [11].

Définition 3.1

L'opérateur $C_x: R^n \rightarrow R^m$ (est un contracteurs pour le sous ensemble X de R^n s'il satisfait les propriétés de contractance et de complétude suivantes :

$$\forall [x] \in R^n, \begin{cases} C_x([x]) \subset [x] & \text{contractance} \\ C_x([x]) \cap X = [x] \cap X & \text{complétude} \end{cases} \dots \dots \dots (3 - 1)$$

3.2 Problèmes de satisfaction de contraintes

Les problèmes de satisfaction de contraintes ou CSP sont des problèmes mathématiques où l'on cherche des états ou des objets satisfaisant un certain nombre de contraintes ou de critères.

Les CSP font l'objet de recherches intenses à la fois en intelligence artificielle et en recherche opérationnelle. De nombreux CSP nécessitent la combinaison d'heuristiques et de méthodes d'optimisation combinatoire pour être résolus en un temps raisonnable. Ils sont notamment au cœur de la programmation par contraintes

$$\begin{cases} f \subset R^n \rightarrow R^m & : \text{fonction} \\ X & : \text{domaine de variable initiale} \\ x = (x_1, x_2, x_3) & : \text{vecteur variable} \end{cases} \dots \dots \dots (3-2)$$

Nous considérons un système d'équations

$$f(x) = 0 \Leftrightarrow \begin{cases} f_1(x_1, x_2, \dots, x_n) = 0 \\ \vdots \\ f_j(x_1, x_2, \dots, x_n) = 0 \quad (n \leq m) \\ \vdots \\ f_m(x_1, x_2, \dots, x_m) = 0 \end{cases} \dots \dots \dots (3-3)$$

Nous appelons problème de satisfaction de contraintes \mathcal{H} , la détermination de toutes les solutions du système d'équations à résoudre (3-3), contenues dans le domaine de recherche initial X .

On note : $\mathcal{H} : (f(x) = 0, x \in X \dots \dots \dots (3-4)$

Si 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 :

$$S = \bigcap_1^n S_j \dots \dots \dots (3-5)$$

3.2.1 Projection de contraintes

Considérons un ensemble S , nous notons $Cs([x])$ le plus petit pavé contenant $[x]$.

L'opérateur de contraction Cs est en mesure de remplacer une boîte $[x]$ par un pavé $S \setminus [x]$ 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-résolveurs [10].

3.3 Graphe acyclique orienté DAG

Le DAG n'est pas un outil spécifique aux méthodes par intervalles [11]. Les méthodes de création de DAG sont ainsi largement documentées en particulier en informatique pour la réalisation de compilateurs. Nous présenterons donc ces méthodes dans l'optique d'expliquer le fonctionnement des DAG et leur lien avec les contraintes. La liaison entre contraintes et DAG pour réaliser les algorithmes de propagation rétro-propagation permet donc d'optimiser le traitement des CSP. Ceci ne résout pas les problèmes de convergence lente (algorithme de complexité pseudo-polynomiale) ni les blocages sur des pavés fixes mais permet d'optimiser le temps de calcul, ce qui n'est pas négligeable en particulier pour les CSP de grandes dimensions.

Notre utilisation des DAG pour la propagation de contraintes a été publiée dans JESA [12], depuis d'autres chercheurs ont détaillé leur propre approche en particulier pour l'optimisation globale [13].

3.3.1 Avantages de l'utilisation d'un DAG

Utiliser des DAG est intéressant puisque cela apporte de nombreux avantages quant à la gestion de ces derniers comme nous allons le voir ensemble. L'utilisation d'un DAG apporte des avantages de rapidité de déploiement de transactions, qui ne sont pas architecturés par blocs. Mais comme expliqué plus haut les DAG ne sont pas les principales caractéristiques qui les utilisent et ce sont d'autres paramètres qui apportent leur confort d'utilisation comme des frais proches de zéro.

3.3.2 Graphe simple ou multigraphes :

Une boucle est une arête qui relie un nœud à lui-même.

Un lien double caractérise l'existence de plusieurs arêtes entre deux nœuds donnés.

Un graphe possédant l'une ou l'autre de ces caractéristiques est dit multigraphe. Un graphe ne possédant aucune des deux est dit graphe simple. Nous travaillerons exclusivement sur des graphes simples par soucis de simplicité : ils couvrent la plupart des utilisations, et sont plus simples à traiter que les multigraphes dans le cas général.

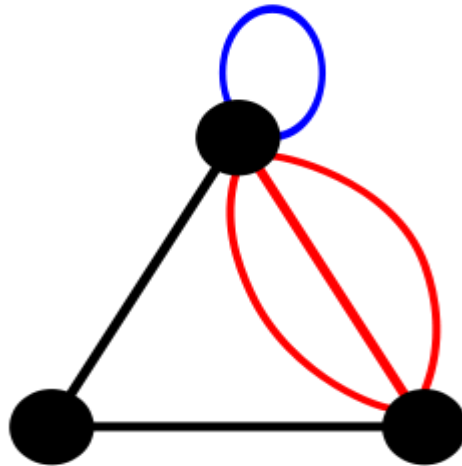


Figure3-1 : Multigraphes, avec une boucle en bleue et les liens double en rouge

3.3.3 Connexité

Un graphe est dit connexe lorsqu'il existe un chemin entre toute paire de nœuds. Une composante connexe d'un graphe est un sous-graphe connexe de ce graphe. Un sous-graphe est un sous-ensemble de nœuds du graphe, avec une partie de leurs arêtes associées. Ainsi, sur le dessin ci-dessous, vous ne voyez qu'un seul et unique graphe, comportant 3 composantes connexes.

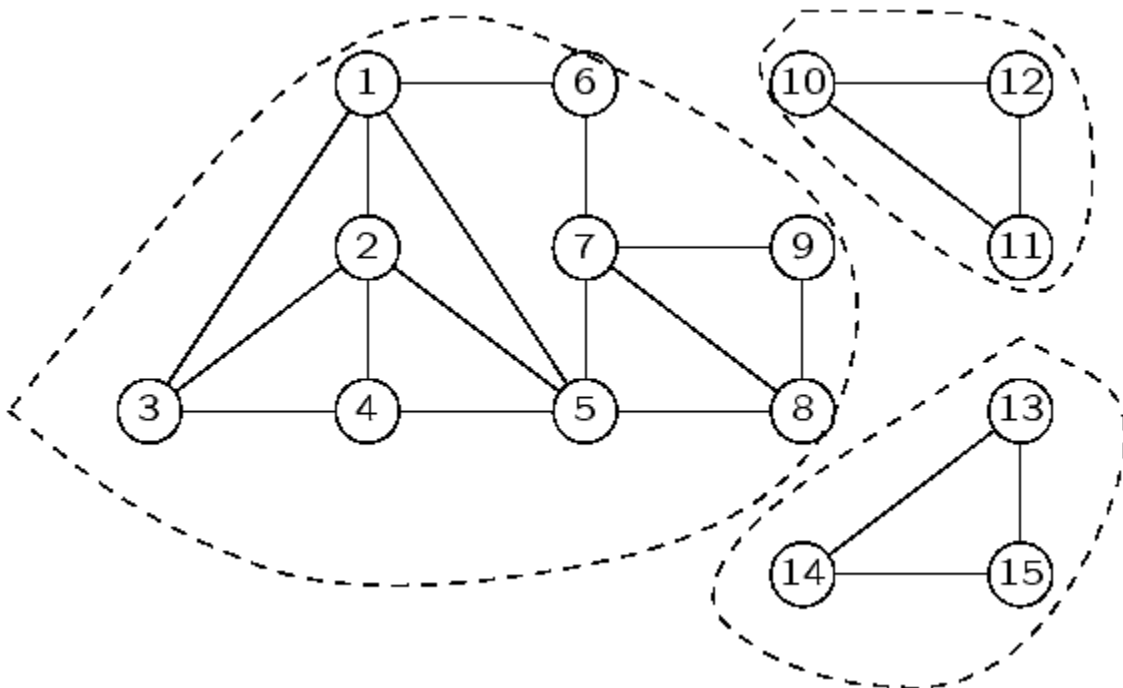


Figure 3-2 : Graphe simple avec 3 composantes connexes

3.4 Les contraintes sous forme DAG

L'ensemble de ces définitions de la théorie des graphes permettent de comprendre ce que sont les DAG. Reste à illustrer comment ces outils vont nous permettre de représenter les contraintes.

Prenons l'exemple suivant :

$$\begin{cases} y_1 = \sin((x_1 + x_2).x_2). \sin(x_1) \\ y_2 = \max((x_1 + x_2).x_2, x_2.(x_2 + x_3)) \\ y_3 = \sin((x_2 + x_3).x_2) \end{cases} \dots\dots\dots (3 - 6)$$

Ces trois contraintes peuvent se mettre sous la forme de trois arbres comme l'illustre la figure 3-3.

Pour chaque arbre, chaque nœud qui n'est pas feuille correspond à une contrainte primitive.

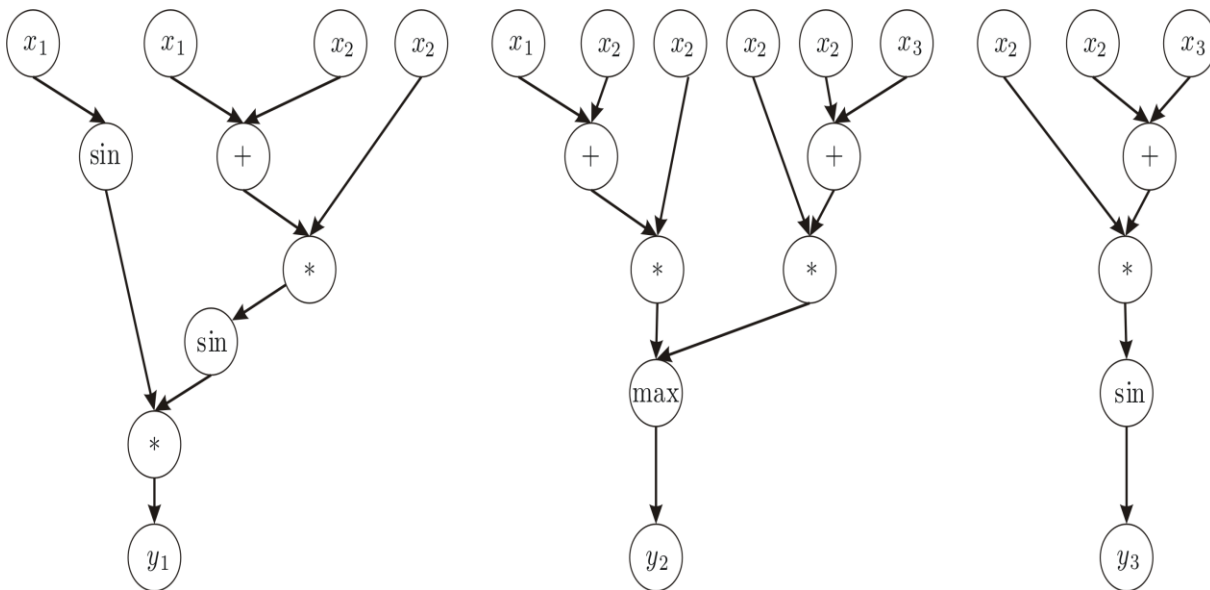


FIGURE 3-3 : Représentation des contraintes sous forme d'arbres

Nous obtenons donc la décomposition suivante :

$$\left. \begin{array}{l|l|l} a1 = x1 + x2 & a5 = x2 + x3 & a9 = x2 + x3 \\ a2 = a1 * x2 & a6 = x2 * a5 & a10 = a9 + x3 \\ a3 = \sin(a2) & a7 = x1 + x2 & . \\ a4 = \sin(x1) & a8 = a7 * x2 & . \\ y1 = a3 * a4 & y2 = \max(a6, a8) & y3 = a9 * x2 \end{array} \right\} \dots \dots \dots (3 - 7)$$

Nous avons vu dans cette partie que c'est sur cette décomposition que se basent les algorithmes de propagations de contraintes. Dans cette décomposition, les variables intermédiaires ne contiennent pas d'information et ont pour domaine initial : $] - \infty, +\infty[$

La mise en place du DAG permet de supprimer des redondances inutiles. Les variables a1 et a7 correspondant aux contraintes : $a1 = x1 + x2$ et $a7 = x1 + x2$, contiennent le même domaine $[x1] + [x2]$ lors de la propagation. Après la rétro-propagation, les domaines sont différents mais seront à nouveau les mêmes à la propagation suivante. La création du DAG permet ainsi de rassembler ces deux contraintes primitives en une seule et ainsi de gagner en temps de calcul.

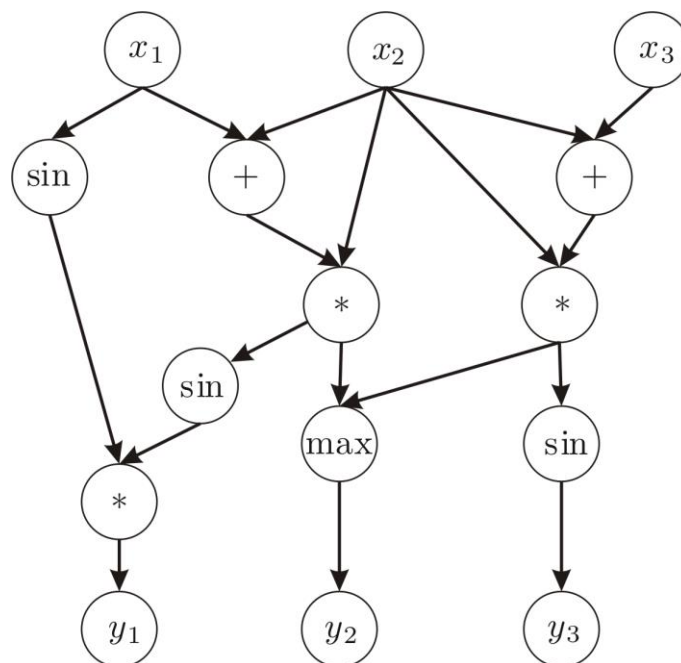


FIGURE 3-4 : Représentation des contraintes sous forme de DAG.

3.5 CONCLUSION

L'outil de contraction ou de réduction est apparu à cause de la difficulté rencontrée lors des opérations de bisection d'un pavé, par exemple dans l'algorithme SIVIA pour un nombre de direction $N = 20$, on aura à faire $2^{20} = 1048576$ bisections. Cet outil permet de réduire le nombre de bisections à effectuer. L'une des applications conçue pour cette technique est « [INTERVAL PEELER](#) » (voir [Annexe](#)).

CHAPITRE IV

4 Application au bioréacteur

4.1 Introduction

Dans le présent travail, nous avons recours à l'approche algébrique différentielle des problèmes généraux d'observation pour étudier l'identification et l'observation des processus de digestion anaérobie.

4.2 Approche Algébrique

L'approche algébrique différentielle des problèmes d'observation remonte vers les années 90 [16]. Le principe de cette approche, tel que clarifié pour la première fois dans [17], est qu'une quantité, disons x , d'un système est observable par rapport à une autre, disons y (qui est supposée être disponible dans un certain intervalle de temps), si chaque composante de x est une solution d'une équation algébrique (non différentielle) avec des coefficients dépendant éventuellement de y et d'un nombre fini de ses dérivées temporelles. La théorie s'applique aux modèles de systèmes uniquement en termes d'équations algébriques différentielles mais qui peuvent être implicites dans les variables à observer.

C'est un fait que les modèles de processus biotechnologiques sont souvent décrits en termes d'équations algébriques différentielles. Les seules expressions non polynomiales qui entrent dans ces modèles sont souvent des expressions rationnelles. Comme l'approche algébrique différentielle de base peut gérer de telles expressions rationnelles. En résumé, l'approche algébrique différentielle de l'observabilité non linéaire [18], [19] peut aborder les questions d'identification et d'observabilité qui se posent dans processus biotechnologiques [20] [21]. Les principales limitations qui peuvent être rencontrées en suivant cette approche sont la disponibilité de ressources de calcul qui sont suffisantes pour les calculs suggérés.

L'approche algébrique différentielle est parmi les rares qui fournissent des tests explicites d'observabilité. Il existe de nombreuses méthodes de décision algébrique différentielle. La plus intéressante utilise des notions d'ensemble de caractéristiques. Le lecteur est renvoyé à [17] pour plus de détails et références concernant les jeux de caractéristiques. Ici, il suffit de mentionner le fait que chaque modèle algébrique différentiel a un ensemble de caractéristiques qui, en principe, peut être calculé en nombre fini d'opérations consistant en l'addition, la multiplication et la

dérivation de polynômes différentiels formels. Des ensembles caractéristiques de systèmes d'équations algébriques différentielles sont définis par rapport aux classements. Et un classement est simplement un ordre total sur les variables et leurs dérivés. En résumé, l'identifiabilité d'un coefficient θ (ou, plus généralement, l'observabilité d'une variable z) par rapport à une variable supposée mesurée w est vérifiée en exécutant des algorithmes d'ensemble de caractéristiques à travers les équations du système relativement un classement qui ordonne w et toutes ses dérivées avant θ (ou z) et toutes ses dérivées, et toute dérivée de ces deux variables avant toute variable potentiellement restante du système.

4.3 Application au bioréacteur

Les bioréacteurs de laboratoires figure 4-1 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.

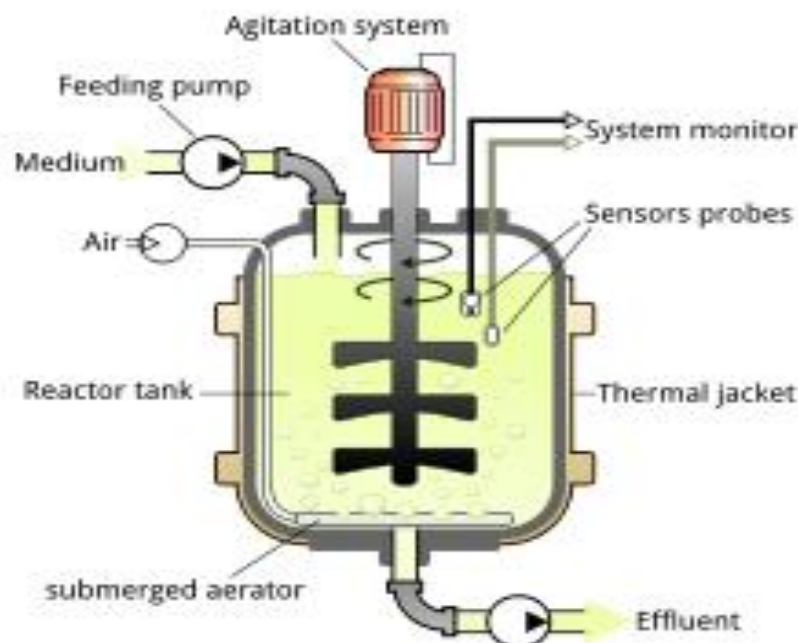


Figure 4-1 – Le Bioréacteur

4.3.1 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 [26] qui s'exprime en fonction de la concentration du substrat S comme suit :

$$\mu = \frac{\mu_{max} * S}{S + K_s} \quad (4 - 1)$$

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.

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 [23 ; 3]. Le modèle de Cantois [24] 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 [1] est proposé.

Dans le cadre de ce mémoire, on a utilisé le modèle de Monod qui fut le premier à modéliser une population microbienne. En se basant sur le bilan de matière.

4.3.2 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. Nous nous intéressons l'identification des paramètres d'un modèle de bioprocédé aérobie de traitement des eaux usées. Dans ce cadre, on utilise un chémostat qui réalise le processus de nitrification. La nitrification est un processus biologique responsable de la production de nitrates dans l'environnement [4]. Elle est réalisée en aérobie par des micro-organismes autotrophes et s'effectue en deux étapes :

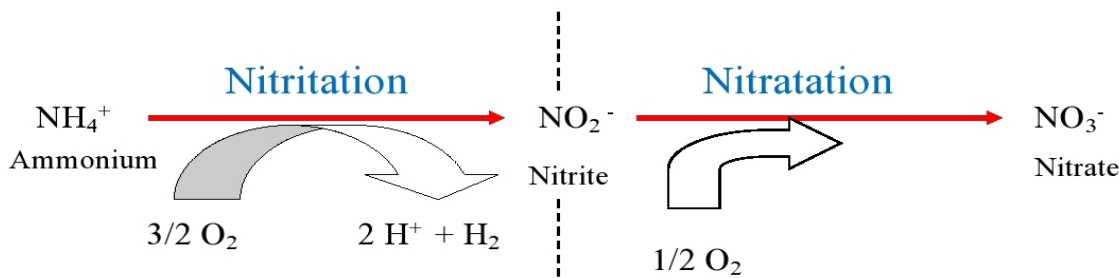


Figure 4.2 – Les deux étapes de la nitrification

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 nitrification est réalisée par des bactéries nitritantes X_1 qui transforme azote ammoniacal (S_1) en nitrites (S_2) et que la nitratation 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 [25].

$$\dot{S}_1 = D(S_{in} - S_1) - K_1\mu_1(S_1)X_1$$

$$\dot{X}_1 = (\mu_1(S_1) - D)X_1$$

$$\dot{S}_2 = K_1\mu_1(S_1)X_1 - K_2\mu_2(S_2)X_2 - S_2D \quad (4 - 2)$$

$$\dot{X}_2 = (\mu_2(S_2) - D)X_2$$

$$\dot{S}_3 = K_2\mu_2(S_2)X_2 - S_3D$$

Avec
$$\mu_1(S_1) = \mu_{max} \frac{S_1}{S_1 + K_{S1}} \quad (4 - 3)$$

Et
$$\mu_2(S_2) = \mu_{max} = \frac{S_2}{S_2 + K_{S2}}$$

$\mu_1(S_1)$ et $\mu_2(S_2)$ sont les taux de croissance de la biomasse. Ils sont modélisés par la cinétique de Monod [26]. $\mu_{max1}(S_1)$ 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

$$D = Q_{in}V \quad (4 - 4)$$

Où Q_{in} est le débit d'entrée du bioréacteur et V son volume.

4.4. Résultats et discussion

L'observabilité des rendements k_1, k_2 du système décrit par les équations (1 au 8) est testée en calculant l'ensemble caractéristique de l'ensemble suivant de polynômes différentiels.

$$\left\{ \begin{array}{l} \dot{S}_1 = D(S_{in} - S_1) - k_1 \mu_1(s_1) X_1 \\ \dot{X}_1 = (\mu_1(s_1) - D) X_1 \\ \dot{S}_2 = k_1 \mu_1(s_1) X_1 - k_2 \mu_2(s_2) X_2 - S_2 D \\ \dot{X}_2 = (\mu_2(s_2) - D) X_2 \\ \dot{S}_3 = k_2 \mu_2(s_2) X_2 - S_3 D \\ \mu_1(S_1 + k_{s1}) = \mu_{\max 1} S_1 \\ \mu_2(S_1 + k_{s2}) = \mu_{\max 2} S_2 \\ y_1 = X_1 \\ y_2 = X_2 \\ y_3 = S_1 \\ y_4 = S_2 \\ y_5 = S_3 \end{array} \right. \quad (4 - 5)$$

Par rapport au classement

$$\{ \{y_1, y_2, y_3, y_4, y_5, D, S_{in}\}, \{k_1, k_2\}, \{X_1, X_2, S_1, S_2, S_3, \mu_1, \mu_2\} \}$$

La solution obtenue par le logiciel :

$$- S_{in} D + D y_5 + D y_4 + D y_3 + \dot{y}_5 + \dot{y}_4 + \dot{y}_3 (1)$$

$$-\dot{S}_{in} D^2 y_1 - \dot{S}_{in} D \dot{y}_1 - S_{in} \dot{D} y_1 + S_{in} D^2 \dot{y}_1 + S_{in} D \ddot{y}_1 - \dot{D} \dot{y}_3 y_1 + \dot{D} y_3 \dot{y}_1 + D^2 y_3 \dot{y}_1 - D^2 y_3 \ddot{y}_1 + D \ddot{y}_3 y_1 - D y_3 \ddot{y}_1 + \ddot{y}_3 \ddot{y}_1 - \dot{y}_3 \dot{y}_1 \quad (2)$$

$$S_{in} \dot{D} D \dot{y}_2 y_1 + S_{in} \dot{D} D y_2 \dot{y}_1 + S_{in} D^3 \dot{y}_2 y_1 - S_{in} D^3 y_2 \dot{y}_1 + S_{in} D^2 \ddot{y}_2 y_1 - S_{in} D^2 y_2 \dot{y}_1 + S_{in} D \ddot{y}_2 \dot{y}_1 - S_{in} D \dot{y}_2 \ddot{y}_1 + \dot{D} D \dot{y}_4 y_2 y_1 + \dot{D} D y_4 \dot{y}_2 y_1 + \dot{D} D y_3 \dot{y}_2 y_1 - \dot{D} D y_3 \dot{y}_1 y_2 - \dot{D} \dot{y}_4 y_2 \dot{y}_1 + \dot{D} \dot{y}_2 y_4 \dot{y}_1 + \dot{D} \dot{y}_2 y_1 \dot{y}_3 - \dot{D} \dot{y}_3 y_2 \dot{y}_1 + D^3 \dot{y}_4 y_2 y_1 - D^3 y_4 \dot{y}_2 y_1 - D^3 y_3 \dot{y}_2 y_1 + y_3 y_2 \dot{y}_1 + D^2 \ddot{y}_4 y_2 y_1 + D^2 \dot{y}_4 y_2 \dot{y}_1 - D^2 \ddot{y}_2 + y_4 y_1 - D^2 \dot{y}_2 y_4 \dot{y}_1 - D^2 \dot{y}_3 y_1 \dot{y}_2 \dots + D^2 \dot{y}_3 y_2 \dot{y}_1 - D^2 y_3 \ddot{y}_2 y_1 + D^2 y_3 y_2 \ddot{y}_1 + D \ddot{y}_4 \dot{y}_2 y_1 + D \ddot{y}_4 y_2 \dot{y}_1 - D \dot{y}_4 \ddot{y}_2 y_1 - D y_4 \ddot{y}_2 \dot{y}_1 - \dot{y}_3 \ddot{y}_2 y_1 + D \dot{y}_3 y_2 \ddot{y}_1 - D y_3 \ddot{y}_2 \dot{y}_1 + D y_3 \dot{y}_2 \ddot{y}_1 + \ddot{y}_4 \dot{y}_2 \dot{y}_1 - \dot{y}_4 \dot{y}_1 - \dot{y}_2 \dot{y}_3 \dot{y}_1 + \dot{y}_1 \dot{y}_3 \dot{y}_2 (3)$$

$$- S_{in} D + D y_3 + D y_1 k_1 + \dot{y}_3 + \dot{y}_1 k_1 \quad (4)$$

$$-S_{in} D + k_2 D y_2 + k_2 \dot{y}_2 + D y_4 + D y_3 + \dot{y}_4 + \dot{y}_3 \quad (5)$$

$$-X_1 + y_1 \quad (6)$$

$$-X_2 + y_2 \quad (7)$$

$$-S_1 + y_3 \quad (8)$$

$$y_4 - S_2 \quad (9)$$

$$y_5 - S_3 \quad (10)$$

$$-D y_1 - \dot{y}_1 + y_1 \mu_1 \quad (11)$$

$$-D y_2 - \dot{y}_2 + y_2 \mu_2 \quad (12)$$

Lemme

Les rendements k_1 et k_2 sont observables par rapport à $y_1, y_2, y_3, y_4, y_5, D, S_{in}$ puisque le polynôme différentiel qui introduit k_1 et k_2 (les lignes 4 et 5 ci-dessus, obtenu par le logiciel) sont d'ordre 0 en $(k_1$ et $k_2)$. De plus k_1 et k_2 sont identifiables par rapport à $y_1, y_2, y_3, y_4, y_5, D, S_{in}$.

Nous obtenons les expressions de k_1 et k_2 que nous allons exploiter pour appliquer le principe de propagation de contrainte.

$$\left\{ \begin{array}{l} k_1 = \frac{D(S_{in} - y_3) - \dot{y}_3}{D y_1 + \dot{y}_1} \\ k_2 = \frac{D(S_{in} - y_3) - \dot{y}_3}{D y_1 + \dot{y}_1} \end{array} \right. \quad (4 - 6)$$

Dans un premier temps, nous avons simulé le modèle de bioréacteur avec des paramètres inspirés de la littérature [1]. Les mesures enregistrées à partir du chémostat ont été réalisées sur une période de 220 jours. La figure 4-3 représente les simulations du modèle du bioréacteur et celles des données prélevées (voir simulink en Annexe)

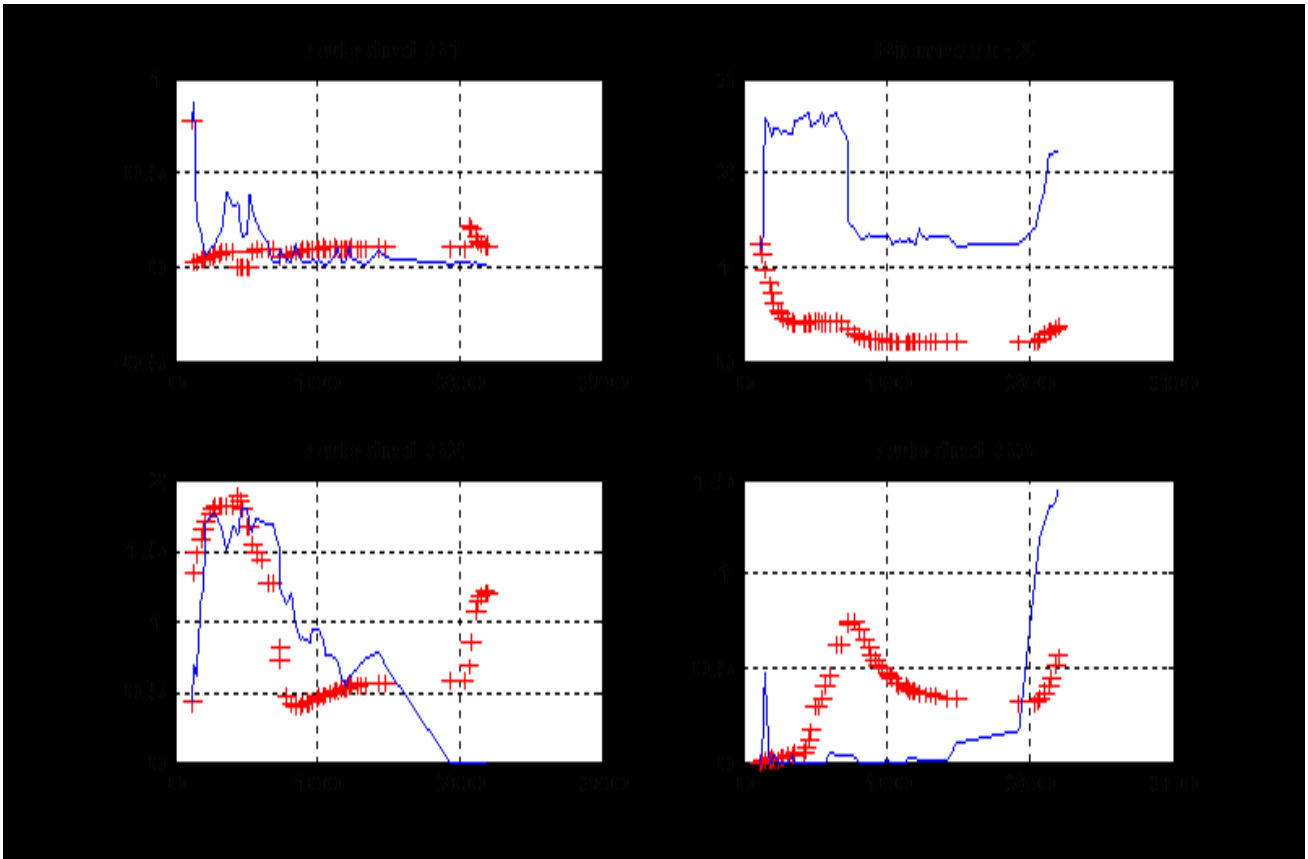


FIGURE 4-3 : Simulations des mesures enregistrées (*bleu*) et du modèle de Nitrification(+)

Les DAG relatifs aux expressions de k_1 et k_2 :

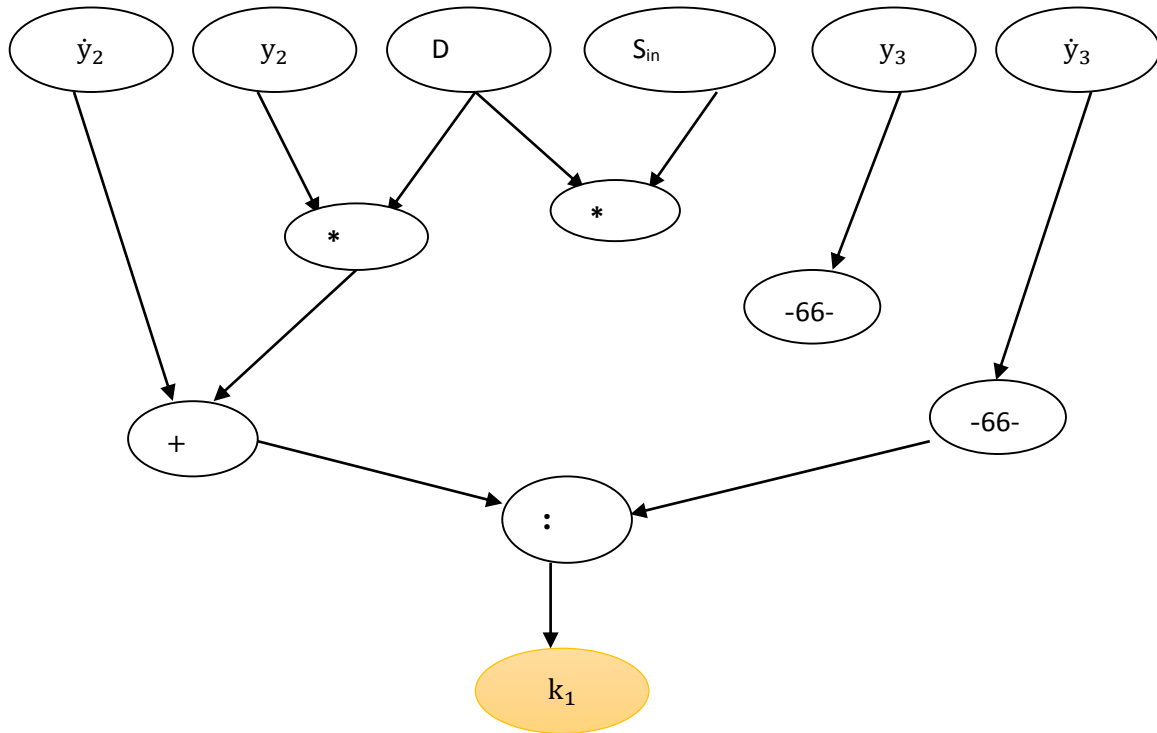


FIGURE 4-41 : DAG relatif à l'expression de k_1

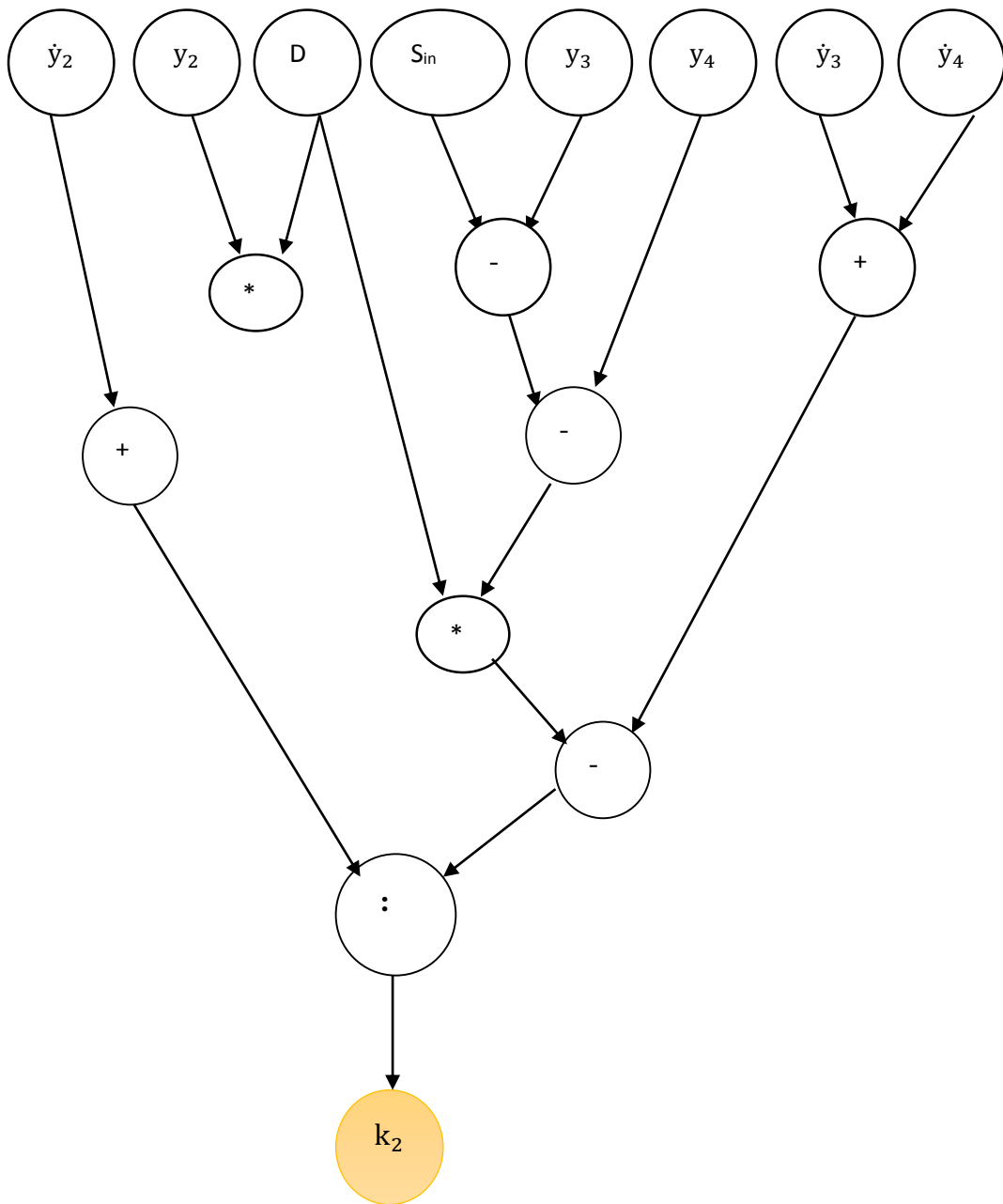


FIGURE 4-42 : DAG relative à expression de k_2

On exprime les deux expressions de k_1 et k_2 sous forme de fonctions primitives :

Propagation de contraintes

$$a_1 = S_{in} - S_1;$$

$$a_2 = D * a_1;$$

$$a_3 = a_2 - \dot{S}_1;$$

$$a_4 = D * X_1;$$

$$a_5 = a_4 + \dot{X}_1;$$

$$k_1 = (a_3/a_5) \cap k_1;$$

$$a_7 = S_{in} - S_2;$$

$$a_8 = a_7 - S_1;$$

$$a_9 = D * a_8;$$

$$a_{10} = \dot{S}_2 + \dot{S}_1;$$

$$a_{11} = a_9 - a_{10};$$

$$a_{12} = D * X_2;$$

$$a_{13} = a_{12} + \dot{X}_2;$$

$$k_2 = (a_{11}/a_{13}) \cap k_2;$$

Rétropropagation de contraintes

$$\dot{X}_2 = (a_{13} - a_{12}) \cap \dot{X}_2$$

$$a_{12} = (a_{13} - \dot{X}_2) \cap a_{12}$$

$$a_{11} = (a_{13} * k_2) \cap a_{11}$$

$$a_{13} = (a_{11}/k_2) \cap a_{13}$$

$$\dot{X}_2 = (a_{13} - a_{12}) \cap \dot{X}_2$$

$$X_2 = (a_{12}/D) \cap X_2$$

$$a_9 = (a_{11} + a_{10}) \cap a_9$$

$$a_{10} = (a_9 - a_{11}) \cap a_{10}$$

$$\dot{S}_2 = (a_{10} - \dot{S}_1) \cap \dot{S}_2$$

$$\dot{S}_1 = (a_{10} - \dot{S}_2) \cap \dot{S}_1$$

$$a_8 = (a_9/D) \cap a_8$$

$$a_7 = (a_8 + S_1) \cap a_7$$

$$S_1 = (a_7 - a_8) \cap S_1$$

$$S_{in} = (a_7 + S_2) \cap S_{in}$$

$$S_2 = (S_{in} - a_7) \cap S_2$$

$$a_3 = (a_5 * k_1) \cap a_3$$

$$a_5 = (a_3/k_1) \cap a_5$$

$$a_4 = (a_5 - \dot{X}_1) \cap a_4$$

$$\dot{X}_1 = (a_5 - a_4) \cap \dot{X}_1$$

$$X_1 = (a_4/D) \cap X_1$$

$$a_2 = (a_3 + \dot{S}_1) \cap a_2$$

$$\dot{S}_1 = (a_2 - a_3) \cap \dot{S}_1$$

$$a_1 = (a_2/D) \cap a_1$$

$$S_{in} = (a_1 + \dot{S}_1) \cap S_{in}$$

$$S_1 = (S_{in} - a_1) \cap S_1$$

$$\left\{ \begin{array}{l} X_1 = 0,975 * X \\ X_2 = 0,035 * X \\ Erreur e = 0,01 \end{array} \right.$$

Tous les intervalles des \mathbf{a}_i sont choisis $[-\infty, +\infty]$, \mathbf{k}_1 et \mathbf{k}_2 sont initialisés avec $[0, +\infty]$

Les intervalles obtenus par simulation sur Matlab, utilisant la toolbox Intlab

$$\begin{cases} \mathbf{k}_1 = [\mathbf{0.6266}, & \mathbf{0.7884}] \\ \mathbf{k}_2 = [\mathbf{0.0630}, & \mathbf{7.3394}] \end{cases}$$

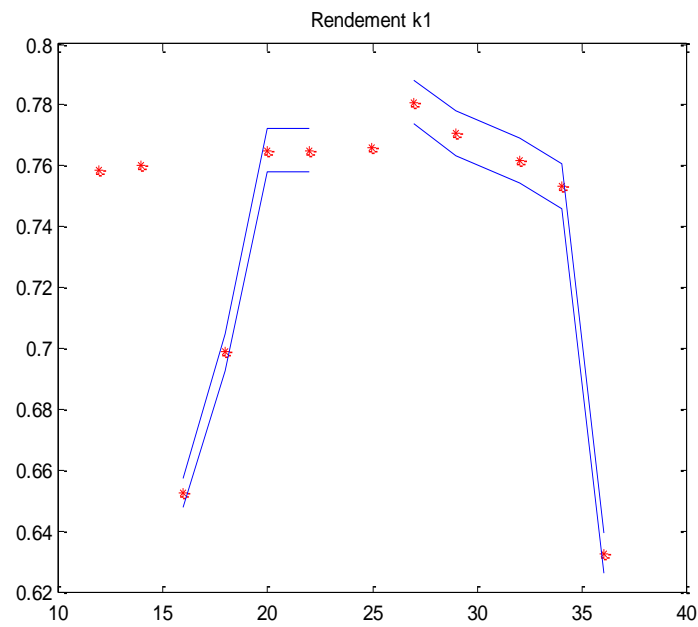


FIGURE 4-51 : Rendement K₁

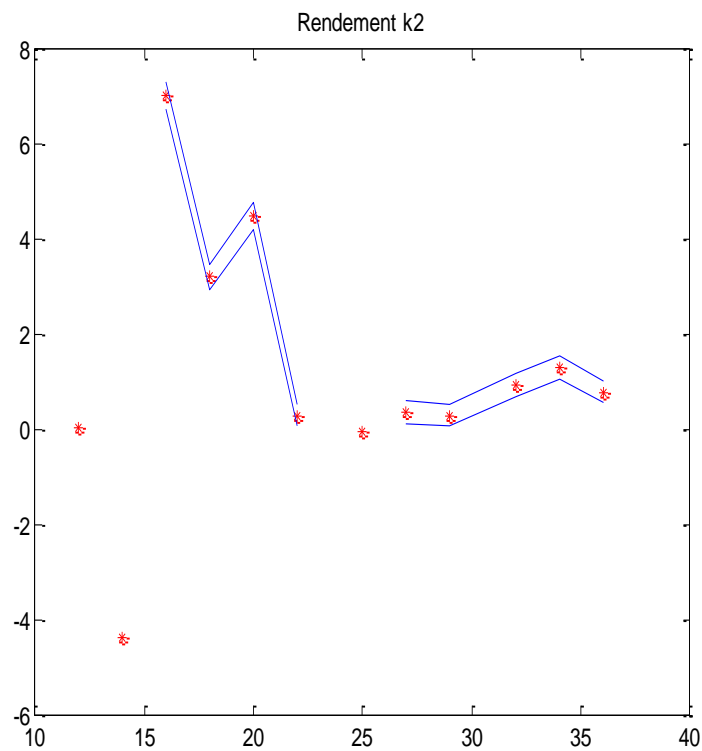


FIGURE 4-52 : Rendement K₂

Les DAG relatifs aux expressions de μ_{max1} et μ_{max2} :

$$\begin{cases} \mu_{max1} = \frac{\mu_1 (S_1 + kS_1)}{S_1} \\ \mu_{max2} = \frac{\mu_2 (S_2 + kS_2)}{S_2} \end{cases} \quad (4 - 7)$$

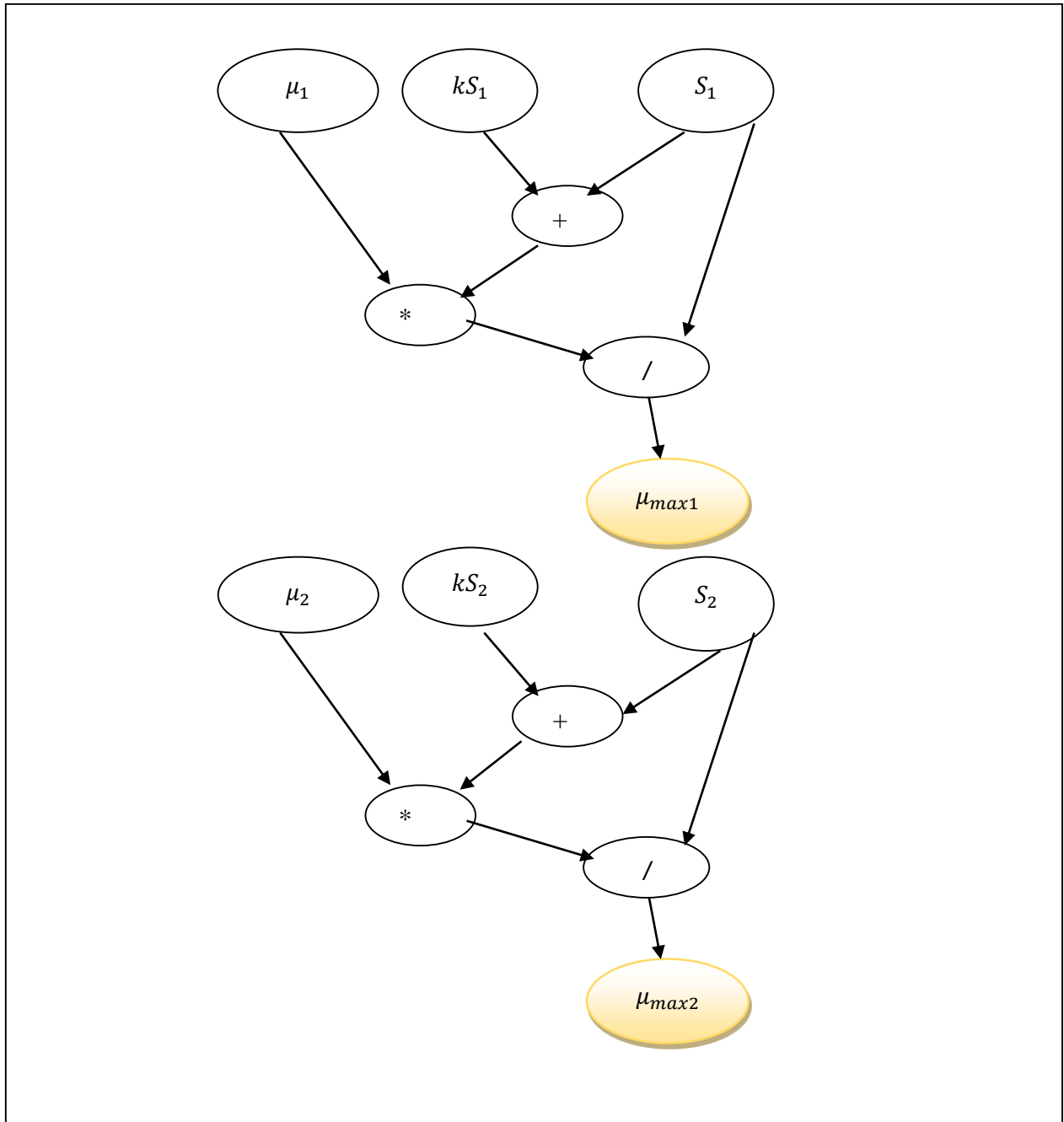


FIGURE 4-6 : Les DAG relatifs aux expressions de μ_{max1} et μ_{max2}

Tous les intervalles des b_i sont choisis $[-\infty, +\infty]$, μ_{max1} et μ_{max2} sont initialisés avec $[0, +\infty]$

➤ Propagation de contraintes

$$\begin{aligned} b_1 &= S_1 + kS_1 \\ b_2 &= \mu_{max1} * S_1 \\ \mu_1 &= (b_2/b_1) \cap \mu_1 \\ b_3 &= S_2 + kS_2 \\ b_4 &= \mu_{max2} * S_2 \\ \mu_2 &= (b_4/b_3) \cap \mu_2 \end{aligned}$$

➤ Rétropropagation de contraintes

$$\begin{aligned} b_4 &= (\mu_2 * b_3) \cap b_4 \\ b_3 &= (b_4/\mu_2) \cap b_3 \\ \mu_{max2} &= (b_4/S_2) \cap \mu_{max2} \\ S_2 &= (b_4/\mu_{max2}) \cap S_2 \\ S_2 &= (b_3 - kS_2) \cap S_2 \\ kS_2 &= (b_3 - S_2) \cap kS_2 \\ b_2 &= (\mu_1 * b_1) \cap b_2 \\ b_1 &= (b_2/\mu_1) \cap b_1 \\ \mu_{max1} &= (b_2/S_1) \cap \mu_{max1} \\ S_1 &= (b_2/\mu_{max1}) \cap S_1 \\ kS_1 &= (b_1 - S_1) \cap kS_1 \\ S_1 &= (b_1 - kS_1) \cap S_1 \end{aligned}$$

Tous les intervalles des b_i sont choisis $[-\infty, +\infty]$, μ_{max1} et μ_{max2} sont initialisés avec $[0, +\infty]$

Les intervalles obtenus par simulation sur Matlab, utilisant la toolbox Intlab

$$\begin{cases} \mu_{max1} = [0.1735, 1.3] \\ \mu_{max2} = [0.2430, 0.3893] \end{cases}$$

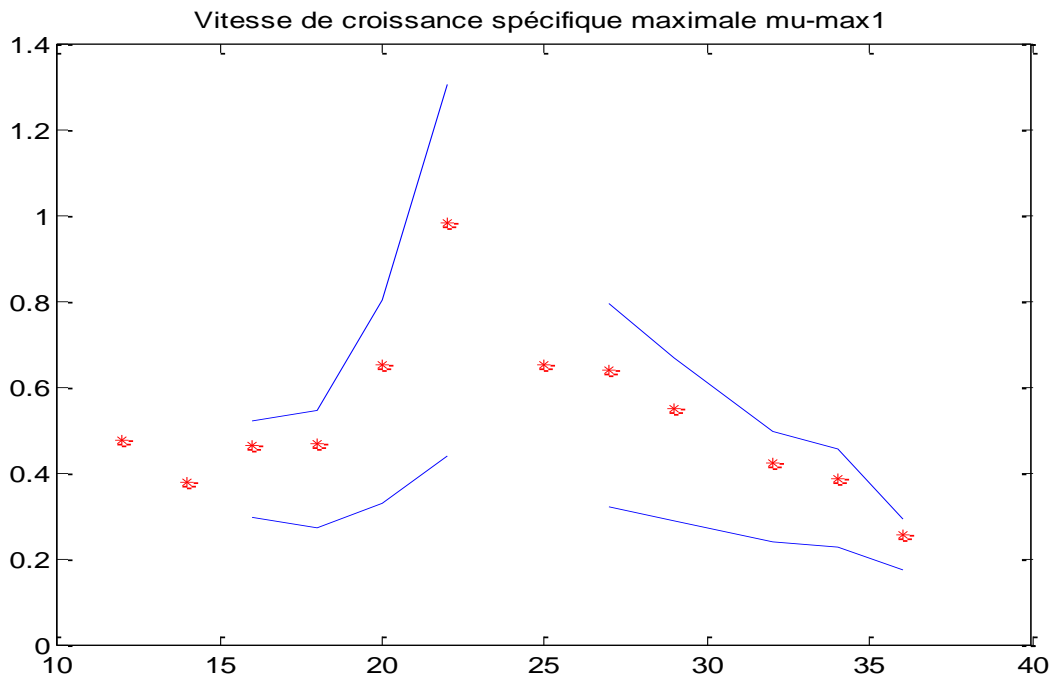


FIGURE 4-71 : Vitesse de croissance maximale μ_{max1}

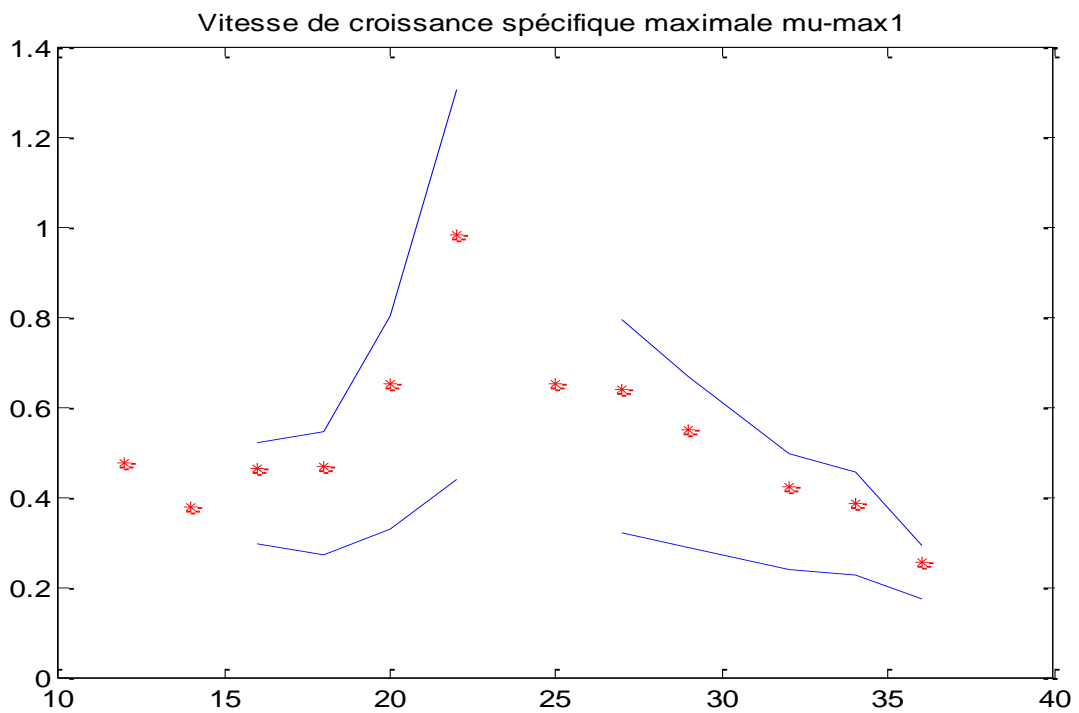


FIGURE 4-72 : Vitesse de croissance maximale μ_{max2}

4.5 Conclusion

Dans ce chapitre nous avons utilisé l'approche algébrique différentielle afin d'étudier l'observabilité et l'identifiabilité d'un modèle de bioréacteur.

Nous avons ensuite décrit un bioréacteur utilisé dans le traitement des eaux usées comme un system dynamique avec des entrées (débit, concentration ...) et des sorties (concentration, des composant de la réaction...) ainsi le rendement K_1 et K_2 , qui sont observables et bien identifiables par rapport aux valeurs, le coefficient de dégradation D et la concentration d'entrée S_{in}

Après une simulation de modèle de réaction avec des paramètres inspirés de la littérature et des mesures sur le chémostat sur un période définie.

Nous décrivons les différents composants d'un bioréacteur et leurs modes de fonctionnement pour une bonne compréhension du principe Eaux usées par boues activées, et représentent le cycle de l'azote et deux étapes Nitrification. Et l'intérêt pour le bioréacteur est en marche de déterminer les taux de croissance maximum $\mu_{max} S_1$ et $\mu_{max} S_2$ pour les deux groupes bactériens X_1 et X_2 , respectivement, puis de suivre l'évolution du rendement (k_1 et k_2), et déterminer la possibilité de son identification et observation.

4.6 Conclusion Générale

Les méthodes d'identification abondent dans la littérature. Ce qui indique l'impossibilité d'établir une relation étroite entre la forme pratique et son modèle mathématique. Le modèle est construit sur la base des données observées, ce qui rend difficile l'étude des systèmes biologiques ou de la biotechnologie lorsque le temps est calculé en jours et lorsque la marge d'erreur est très large. Dans notre étude, nous testons une nouvelle méthode d'identification pour estimer les paramètres du chémostat. Ces méthodes consistent en l'utilisation de techniques ensemblistes basées sur l'arithmétique des intervalles. Le dernier chapitre a été consacré à la description du dispositif expérimental sur lequel notre étude s'est concentrée, c'est un bioréacteur fonctionnant en continu appelé chémostat. Cet appareil conçoit deux étapes de nitrification pour le cycle de l'azote dans le cadre du traitement des eaux usées par boues activées.

Dans le premier chapitre, Nous avons présenté les bases de l'arithmétique des intervalles, rappelé les différentes opérations entre intervalles puis défini les types de récipients pouvant contenir ces ensembles tels que des pavés afin de les manipuler aisément, une notion de fonction d'inclusion est introduite. Dans le deuxième chapitre, on définit l'inversion ensembliste qui est la base de l'identification dans le contexte à erreurs bornées. Nous avons aussi défini la notion de pessimisme résultant qui provoque une surestimation d'un pavé solution. La solution apportée à ce problème de pessimisme est l'introduction de contracteurs et dans le troisième chapitre nous avons présenté en détails la contraction par projection qu'on a utilisée dans ce travail.

Nous avons choisi de déterminer les taux de croissance maximum de $\mu_{\max 1}$ (S1) et $\mu_{\max 2}$ (S2) pour les deux groupes bactériens X_1 et X_2 , respectivement, puis le rendement k_1 et k_2

BIBLIOGRAPHIE

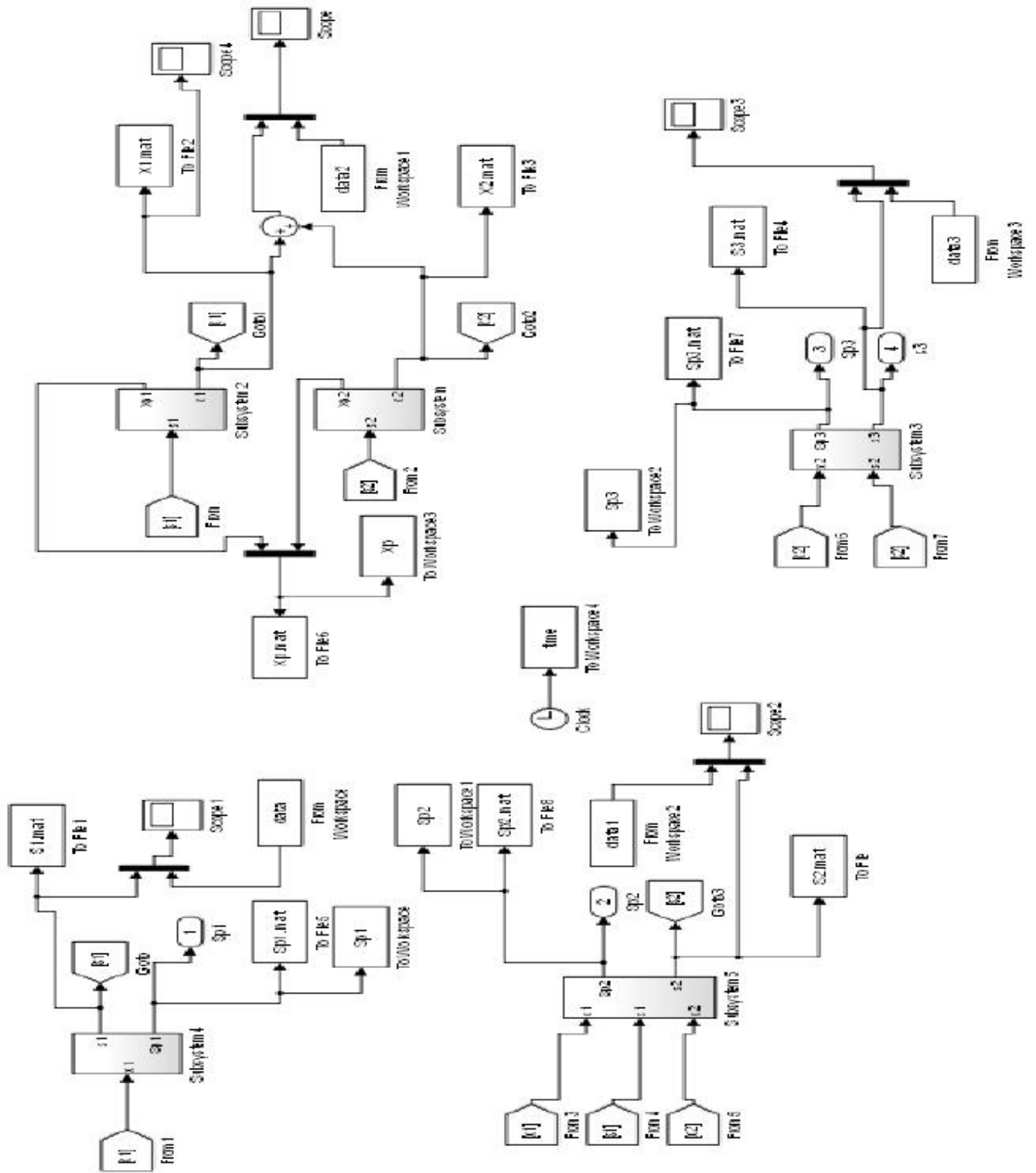
- [1] R Moore. Interval arithmetic. Prentice-Hall, Englewood Cliffs (NJ), USA, 1966.
- [2] A. Neumaier. Interval Methods for Systems of Equations. Cambridge Middle East Library. Cambridge University Press, 1990.
- [3] Eldon Hansen and G William Walster. Global optimization using interval analysis: revised and expanded, volume 264. CRC Press,
- [4] Isabelle Braems, Fabienne Berthier, Luc Jaulin, Michel Kieffer, and Eric Walter. Guaranteed estimation of electrochemical parameters by set inversion using interval analysis. *Journal of Electro analytical Chemistry*, 495(1): 1–9, 2000.
- [5] Luc Jaulin, Michel Kieffer, Olivier Didrit, and Eric Walter. *Applied Interval Analysis*. Springer, 2001
- [6] L. Jaulin, M. L. Kieffer, O. Didrit et E. Walter, *Applied interval analysis with Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer verlag, London, 2001.
- [7] I. Braems. “Analyse par intervalles pour l’estimation et la commande robuste”. PhD dissertation, Université Paris Sud, Orsay, France (2002).
- [8] Isabelle Braems. Méthodes ensemblistes garanties pour l’estimation de grandeurs physiques. PhD of, 2002.
- [9] L.Jaulin, Interval contractors and their applications, Université d'Angers, lesson 9-12 February 2004 à l’Université Politècnica de Catalunya (Barcelone), 2004.
- [10] http://www.istia.univangers.fr/~bagueonar/interval_peeler_console_sources.zip
- [11] INTLAB- INTerval LABoratory: toolbox de MATLAB pour la manipulation des intervalles, développée par Siegfried M. Rump, Insitute for Reliable Computing Hamburg University of Technology, Germany
- [12] A.AMO,R.SETMAX,& J.ULLMAN.“ Compileurs Principes, techniques et outils”. Informatique intelligence artificielle. InterEditions (1995).
- [13] X.BASWENA.D, M.DAo,L.JAULIN,ANDW.KHALIL. Méthodes ensemblistes pour l’étalonnage géométrique. *Journal Européen des Systèmes Automatisés* 37(9), 1059—1074 (2003).
- [TITRE DU DOCUMENT]55
- [14] H.Schichl and A.Neumaier, Interval Analysis on Directed Graphs for Global Optimization, *Journal of Global Optimization*, vol.33, pp.541-562, 2005.
- [15] <https://zestedesavoir.com>

- [16] Pommaret, J.F. "Géométrie différentielle algébrique et théorie du contrôle".C.R. Acad.Sci.ParisSer. I 302,1986, pp.547–550.
- [17] Diop S. "From the geometry to the algebra of nonlinear observability" Contemporary Trends in Nonlinear Geometric Control Theory and its Applications, A.Anzaldo-Meneses, B. Bonnard, J. P. Gauthier, and F. Monroy-Perez, Eds. Singapore:World Scientific Publishing (2002) Co., 305–345.https://doi.org/10.1142/9789812778079_0012
- [18] Fliess, M.: "Quelques remarques sur les observateurs non lineaires". In : Proceedings Colloque GRETSI Traitement du Signal et des Images, GRETSI, 1987, pp. 169–172
- [19] Diop, S., Fliess, M.: "On nonlinear observability" In: Commault, C., NormandCyrot, D., Dion, J.M.,Dugard, L., Fliess, M., Titli, A., Cohen, G., Benveniste, A., Landau, I.D. (eds.). Proceedings of the European Control Conference, Hermes, Paris, 1991, pp. 152–157 DOI: 10.1109/CDC.1991.261405
- [20] Chorukova E, Diop S, Simeonov I, "On differential algebraic decision methods for the estimation of anaerobic digestion models". Lecture Notes in Computer Science, Springer, Verlag, 2007, 4545, pp.202 – 216, DOI: 10.1007/978-3-540-73433-8_15
- [21] Diop S, Simeonov I "On the biomass specific growth rates estimation for anaerobic digestion using differential algebraic techniques". Int. J. BIO automation, 13(3), 2009, pp.47-56,<https://doi.org/10.1016/j.ifacol.2017.08.2232>
- [22] Maxime Dumont. Apports de la modélisation des interactions pour ne compréhension fonctionnelle d'un écosystème : application à des bactéries nitrifiantes en chémostat. PhD thesis, Montpellier 2, 2008.
- [23] NaveenBhat 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. [TITRE DU DOCUMENT]56
- [24] Luc Jaulin. Path planning using intervals and graphs. Reliable computing, 7(1) :1–15, 2001.
- [25] 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.
- [26] Eberhard Bock, Hans-PeterKoops, Heinz Harms, and B Ahlers. The biochemistry of nitrifying organisms. Variations in autotrophic life, pages 171–200, 1991.
- [27] Luc Jaulin, Michel Kieffer, Olivier Didrit, and Eric Walter. Applied Interval Analysis. Springer, 2001.

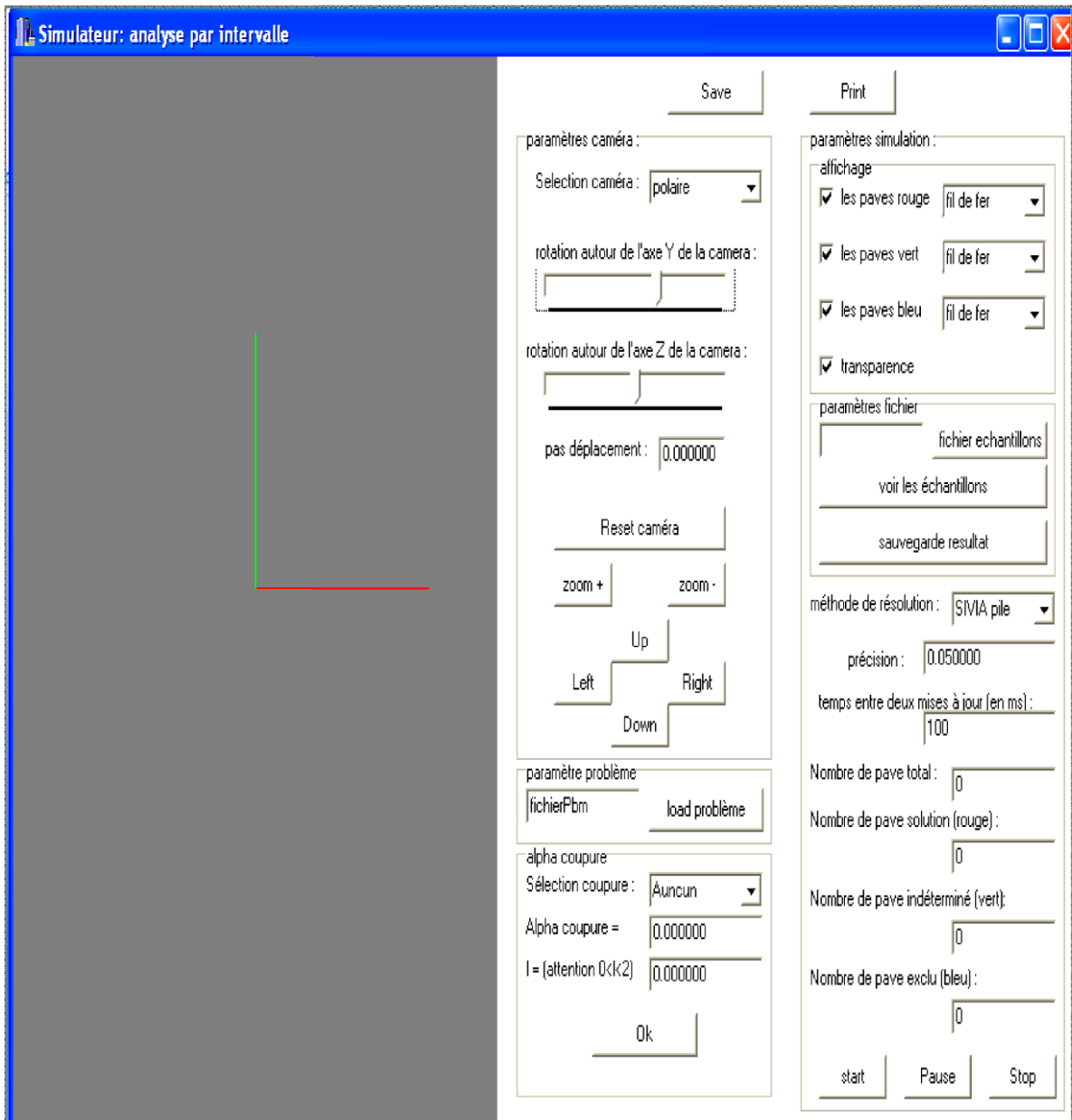
[28] Jean-Guillaume Dumas, Thierry Gautier, Mark Giesbrecht, Pascal Giorgi, Bradford Hovinen, Erich Kaltofen, B David Saunders, Will J Turner, Gilles Villard, et al. Linbox : A generic library for exact linear algebra. In Proceedings of the 2002 International Congress of Mathematical Software, Beijing, China, pages 40–50. World Scientific Pub, 2002

[29] Siegfried M Rump. INTLAB interval laboratory. Springer, 1999.

ANNEXE



« Simulateur : analyse par intervalle » réalisé par Luc Jaulin



INTERVAL PEELER, pour l'utilisation de l'outil « contracteurs », réalisé par L.Jaulin, X. Baguenard et M.Dao.

