MASTER EN SCIENCES MATHÉMATIQUES À FINALITÉ APPROFONDIE

Etude de l'opérateur de Koopman en théorie du contrôle application à la prédiction et la p-dominance

Debauche, Virginie

Award date:
2019

Awarding institution:
Universite de Namur

UNIVERSITE DE NAMUR

Faculté des Sciences

ETUDE DE L'OPERATEUR DE KOOPMAN EN THEORIE DU CONTRÔLE : APPLICATION A LA PREDICTION ET LA p-DOMINANCE

Mémoire présenté pour l'obtention

du grade académique de master à finalité approfondie

Virginie DEBAUCHE

Août 2019

Remerciements

Cette page est ici pour moi l'opportunité de mettre en avant les personnes qui ont rendu possible la remise de ce mémoire, aboutissement des cinq années de labeur qui arrivent ici à leur terme.

Avant toute chose, je voudrais remercier mon promoteur, Monsieur Alexandre Mauroy, pour ses conseils précieux et le temps passé à la relecture et l'orientation dans le travail.

Je voudrais également prendre le temps de remercier ma famille et plus particulièrement ma maman, ma sœur, et également mes amis pour leur relecture constante de ce mémoire. Leur contribution m'aura très certainement permis de rendre le tumulte des études moins laborieux à traverser.

Je voudrais enfin remercier mon compagnon pour son soutien dans les moments émotionnellement compliqués. Quiconque l'a déjà vécu, les études ne sont pas seulement compliquées intellectuellement mais bien souvent psychologiquement et être bien entourée est un plus non négligeable. Ce fût mon cas et j'espère qu'ils percevront, par ces mots, toute la gratitude que j'éprouve envers eux.

Et à tous ceux que j'ai oubliés, merci.

Résumé

A l'heure actuelle, il est difficile de travailler avec des systèmes dynamiques non-linéaires. L'opérateur de Koopman offre une première solution globale puisqu'il permet d'obtenir à partir de tout système dynamique (ouvert ou non) non-linéaire de dimension finie, un système dynamique linéaire de dimension infinie. Cet opérateur peut être approximé en dimension finie grâce à la méthode de décomposition en modes dynamiques, et cette méthode permet également de construire des prédicteurs linéaires. Ceux-ci peuvent être utilisés notamment pour déterminer un contrôle via une commande prédictive dont l'objectif est dans un premier temps d'approximer linéairement la trajectoire et dans un second temps, d'utiliser cette prédiction pour déterminer un signal de contrôle via une méthode linéaire. Par ailleurs, la manipulation de systèmes dynamiques non-linéaires se fait généralement par approche différentielle, autrement dit par linéarisation du système. C'est là-dessus que se base la théorie de la p-dominance qui permet d'étudier le comportement assymptotique d'un système dynamique non-linéaire quelconque.

Ce travail a pour but d'étudier différentes méthodes pour généraliser la définition initiale de l'opérateur de Koopman aux systèmes ouverts, ainsi que plusieurs méthodes de construction d'un prédicteur linéaire. L'erreur de cette prédiction est pour la première fois étudiée et approximée. Finalement, le lien entre p-dominance et l'opérateur de Koopman est établi.

Mots-clé : Opérateur de Koopman, systèmes dynamiques non-linéaires, systèmes ouverts, prédicteurs linéaires, contrôle, erreur, linéarisation, p-dominance

Abstract

Nowadays, working with nonlinear dynamical systems is difficult. The Koopman operator provides a global solution since it yields an infinite dimensional linear dynamical system from a finite dimensional nonlinear one. This operator can be approximated in finite dimension through the extended dynamic mode decomposition method, and it allows to build linear predictors. Linear predictors can be used in the context of model predictive control whose purpose is to approximate linearly the trajectory, and use this prediction to build a signal control with a linear method. Furthermore, dynamical systems can also be studied through a differential approach based on the linearization of the system. p-dominance relies on this approach, and allows to study the asymptotic behaviour of any nonlinear dynamical system.

This work aims to provide different methods to extend the original definition of the Koopman operator to open systems, and different methods to build linear predictors. The error of this prediction is also studied and approximated for the first time. Finally, we investigate the connection between p-dominance and the Koopman operator.

Keywords : Koopman operator, nonlinear dynamical systems, open systems, linear predictors, control, error, linearization, p-dominance

Table des matières

Introduction

Depuis toujours, les mathématiques jouent un rôle crucial dans de nombreux aspects de la vie quotidienne. Au supermarché, à la banque, devant la météo, nous rencontrons partout les mathématiques souvent même sans nous en apercevoir. Nous sommes capables de décrire, d'étudier et de modéliser la plupart des phénomènes qui se produisent dans le monde (et même au-delà). Pour ce faire, nous utilisons différents outils mathématiques, dont les systèmes dynamiques. Naturellement, certains d'entre eux sont plus faciles à manipuler commes les systèmes dynamiques linéaires, et ils ont donc fait l'objet des premières recherches. Quel que soit l'objectif parcouru, il existe généralement une multitude de méthodes pour réaliser cet objectif lorsqu'il s'agit d'un système dynamique linéaire. Malheureusement, la tâche est souvent plus complexe lorsqu'on fait face à un système dynamique non-linéaire. Pour remédier à ce problème, deux stratégies peuvent être envisagées. La première consiste à développer de nouvelles méthodes spécifiques aux systèmes dynamiques non-linéaires, tandis que la deuxième consiste à se baser sur le travail déjà effectué pour les systèmes dynamiques linéaires. En effet, plutôt que de construire de nouvelles méthodes, cette seconde option vise à approximer le système non-linéaire initial par un système linéaire auquel nous pouvons appliquer les outils adaptés aux systèmes linéaires. C'est cette approche que nous allons développer dans ce mémoire.

Il existe actuellement plusieurs méthodes pour déduire un système linéaire à partir d'un système non-linéaire. Toutefois, nous n'allons pas reprendre une méthode déjà établie depuis plusieurs années, mais nous allons plutôt nous pencher sur une nouvelle méthode basée sur l'opérateur de Koopman. A partir des références [11] et [18], nous présentons l'opérateur de Koopman ainsi que ses propriétés importantes. Un des désavantages majeurs de cet opérateur réside dans sa dimension puisqu'elle est infinie. Ainsi nous développerons dans la troisième section du premier chapitre une méthode d'approximation finie, à savoir la méthode (étendue) de décomposition en modes dynamiques. Initialement défini pour les systèmes fermés, l'opérateur de Koopman sera généralisé aux systèmes ouverts dans la dernière section du premier chapitre, ainsi que la méthode d'approximation finie. Pour ce faire, nous nous baserons principalement sur les références [9], [13] et [17]. L'opérateur de Koopman va nous permettre de réaliser l'objectif initial, à savoir de décrire des méthodes de prédiction linéaire de systèmes dynamiques non-linéaires. Par ailleurs, nous développerons brièvement quelques méthodes de contrôle dans la troisième section du deuxième chapitre. Le troisième chapitre, quant à lui, aborde l'erreur que nous commettons en approximant linéairement le système dynamique non-linéaire. Dans un premier temps, nous étudierons théoriquement pour la première fois dans la littérature cette erreur et nous tenterons de la borner et dans un deuxième temps, nous étudierons les variations de l'erreur en fonction des valeurs de différents paramètres tels que le pas d'intégration ou encore le choix de la base.

Le dernier chapitre de ce mémoire aborde un autre nouvel outil, à savoir la p-dominance. La méthode la plus instinctive pour obtenir un système linéaire à partir d'un système non-linéaire est la linéarisation. C'est sur ce principe que la p-dominance a été introduite puisqu'elle est initialement définie pour les systèmes linéaires, et ensuite étendue aux systèmes non-linéaires par approche différentielle. La théorie de la p-dominance vise à déterminer la dimension du comportement asymptotique d'un système dynamique. Ainsi, cette theorie permet de contrer un autre problème actuel, à savoir la taille des systèmes étudiés. Nous aborderons un cas particulier de la p-dominance dans la deuxième section de ce chapitre, à savoir la positivité différentielle. Cette théorie est importante puisqu'un lien a déjà été établi avec l'opérateur de Koopman dans la référence [10]. L'approche différentielle sera récurrente dans ce dernier chapitre puisque p-dominance et positivité différentielle seront définies de la sorte. Ainsi, c'est cette méthode que nous utiliserons pour étendre le lien préexistant entre la positivité différentielle et l'opérateur de Koopman aux systèmes p-dominants.

Chapitre 1

L'opérateur de Koopman

Ce mémoire se base essentiellement sur l'opérateur de Koopman. Dans les deux premières sections de ce chapitre, nous définirons cet opérateur et nous en tirerons les principales propriétés qui en font un outil si intéresant. L'opérateur de Koopman présente cependant quelques inconvénients dont sa dimension infinie. Il est toutefois possible de contourner ce problème en l'approximant par une matrice de dimension finie. Cette méthode est développée dans la troisième section. Finalement, la quatrième et dernière section de ce chapitre a pour but de généraliser l'opérateur de Koopman aux systèmes dynamiques ouverts, c'est-à-dire les systèmes avec une entrée. Ce chapitre est essentiellement basé sur les notes des cours de *Méthodes avancées pour les systèmes non linéaires* [11] et *Systèmes complexes commandés* [18].

1.1 Préliminaires théoriques

La première étape dans le processus de compréhension et d'étude d'un phénomène physique évoluant dans le temps (par exemple le mouvement rectiligne uniforme, la stabilisation d'un pendule, la diffusion de chaleur, etc.) est la mise en équation du phénomène. L'objectif est de trouver les équations qui régissent le phénomène, autrement dit les règles que suivent les différents acteurs. Les équations résultantes dérivent alors un système dynamique. Plusieurs types de systèmes sont possibles (décrits par des équations différentielles partielles, systèmes avec délais, systèmes hybrides, etc.) mais seuls deux d'entre eux seront exploités dans ce travail : d'une part des systèmes discrets pour lesquels l'état évolue de manière discrète (typiquement l'état est indicé par un naturel), et d'autre part, des sytèmes continus pour lesquels la dynamique est valable en tout temps positif. Décrivons brièvement chacun de ces deux types de systèmes.

Dans un premier temps, intéressons-nous aux systèmes dynamiques discrets. Ceux-ci sont décrits par une application (en général non linéaire)

$$\begin{array}{rccc} T: & X & \longrightarrow & X \\ & x & \mapsto & T(x)\,, \end{array}$$

où X est un espace donné de dimension finie (en général, X est l'espace $\mathbb{R}^n$ pour une dimension n fixée). Comme illustré dans la FIGURE 1.1, la dynamique est alors donnée par

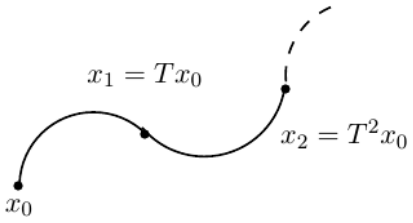


FIGURE 1.1 – Illustration d'un système dynamique discret

$$x_{k+1} \;=\; T\left(x_k\right) \;,\; \forall k \in \mathbb{N}. \tag{1.1}$$

La trajectoire, ou plus exactement l'orbite dans le cas discret, est définie comme la composition successive de l'application T, c'est-à-dire qu'à partir d'une condition initiale x_0, on définit le $k$ième élément comme

$$x_k \;=\; T^k(x_0).$$

Les systèmes dynamiques discrets sont par exemple utilisés pour modéliser l'évolution d'une population. Le modèle le plus connu est celui de Pierre-François Verhulst (1804-1849), plus souvent appelé application logistique.

Exemple 1.1. *Le modèle de l'application logistique suppose que le taux de croissance d'une population est proportionnel à sa taille actuelle p_k, à sa taille maximale P et l'écart entre ces deux valeurs. Ainsi, l'évolution d'une population est donnée par*

$$p_{k+1} \;=\; Kp_k(L - p_k).$$

Il suffit alors de poser $x_k = p_k/L$, et nous obtenons le système discret décrit par

$$x_{k+1} \;=\; \mu\, x_k\,(1 - x_k). \tag{1.2}$$

Un système dynamique continu est également défini à partir d'un champ de vecteurs $F : X \subset \mathbb{R}^n \to X$ à priori non-linéaire de sorte que la dynamique soit donnée par

$$\dot{x} \;=\; F(x). \tag{1.3}$$

Cette équation décrit l'évolution de la dérivée de l'état, contrairement au cas discret où l'équation décrit directement l'évolution de l'état. Dans le cas continu, cette évolution est décrite par le flot, à savoir la fonction

$$\begin{array}{rccc} \varphi : & \mathbb{R} \times X & \to & X \\ & (t, x) & \mapsto & \varphi(t, x), \end{array}$$

où $\varphi(t, x_0)$ désigne la solution au temps t du système dynamique (1.3) issue de la condition initiale x_0 comme illustré à la FIGURE 1.2. Dans le cadre de ce travail, nous considérons

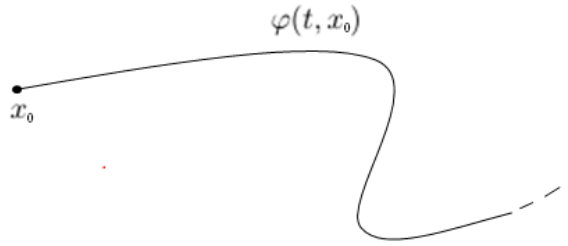


FIGURE 1.2 – Illustration du flot d'un système dynamique continu

que l'ensemble X est $\mathbb{R}^n$ pour une dimension n fixée. Par ailleurs, notons que si la fonction F est Lispschitz [1], alors le système dynamique (1.3) admet une solution unique (sur un certain intervalle de temps).

Exemple 1.2. *L'évolution d'une population peut également être modélisé par un système dynamique continu. Le modèle logistique continu est décrit par*

$$\dot{p} \;=\; K\,p\,(L-p).$$

La solution en fonction du temps $p(t)$ de cette équation différentielle est alors donnée par

$$p(t) \;=\; \frac{L\,p_0\,e^{LKt}}{L-p_0+p_0e^{LKt}},$$

où $p_0 = p(0)$ désigne la population initiale.

L'approche décrite jusqu'à présente est l'approche habituelle, mais il est également possible de définir des systèmes discrets et continus pour des espaces fonctionnels et des opérateurs. En effet, plutôt que de considérer des éléments de $\mathbb{R}^n$, nous pouvons considérer des élements d'un espace fonctionnel $\mathcal{F}$ de Banach. En temps discret, le système est défini grâce à un opérateur

$$\begin{array}{rccc} A: & \mathcal{F} & \to & \mathcal{F} \\ & f & \mapsto & Af. \end{array}$$

La dynamique est alors décrite pas

$$\begin{array}{rcl} f_{k+1} & = & Af_k, \\ & = & A^{\circ(k+1)} f_0, \end{array}$$

pour tout $k \in \mathbb{N}$. De manière similaire, le cas continu peut être décrit au moyen d'un semi-groupe d'opérateurs $\{A^t\}_{t\geq 0}$ de sorte que la fonction initiale f_0 soit envoyée sur $A^t f_0$. Manipuler des fonctions plutôt que des états implique de travailler en dimension infinie, et cela ne facilite rien. Cela pourrait toutefois être utile si un changement de dimension engendrait un avantage majeur : c'est le cas de l'opérateur de Kooopman.

1. Une fonction F est dite Lipschiz s'il existe une constante positive k telle que pour tout $x, y \in X$, $|F(x) - F(x)| \;\leq\; k\,|x-y|$.

1.2 Définition et propriétés de l'opérateur

La manipulation d'objets en dimension infinie n'est pas aisée. Toutefois, un système linéaire est toujours plus facile à manipuler qu'un système non-linéaire. En effet, de nombreux résultats ont déjà été obtenus dans le cas linéaire, et la manipulation de systèmes non-linéaires se fait souvent par approche différentielle, c'est-à-dire via la linéarisation du système. Il semble donc préférable de considérer un système linéaire plutôt que non-linéaire, quite à obtenir un système de dimension infinie. C'est exactement l'action de l'opérateur de Koopman. Ce dernier agit toujours sur des observables, autrement dit des fonctions $f : X \to \mathbb{R}$ (ou $\mathbb{C}$) à valeurs scalaires, et capture leur évolution le long des trajectoires d'un système dynamique associé.

Définition 1.1

Soit un système dynamique discret $x_{k+1} = Tx_k$ avec l'application $T : X \to X$, et $\mathcal{F}$ un espace fonctionnel. L'**opérateur de Koopman** noté U, est défini par

$$\begin{aligned} U : \mathcal{F} &\to \mathcal{U} \\ f &\mapsto Uf := f \circ T, \end{aligned} \tag{1.4}$$

de sorte que, pour tout $x \in X$, $Uf(x) = f(T(x))$.

L'opérateur s'applique donc sur une fonction scalaire, et retourne une nouvelle fonction dont les valeurs seront celles une étape plus loin dans la dynamique. Il est d'ailleurs parfois appelé opérateur de composition. Celui-ci peut également être défini pour un système dynamique continu. Nous obtenons alors un semi-groupe d'opérateurs $\{U^t\}_{t\geq 0}$ tels que

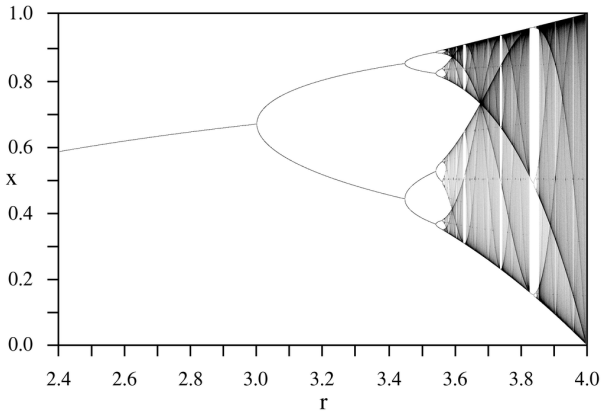
$$U^t f = f \circ \varphi^t, \tag{1.5}$$

où φ^t est une notation pour désigner le flot $\varphi(t, .)$. Dans ce cas, il est possible de définir le générateur infinitésimal de ce semi-groupe, c'est-à-dire l'opérateur non borné noté $L : D(L) \subset \mathcal{F} \to \mathcal{F}$ tel que pour toute fonction $f \in D(L)$, on a

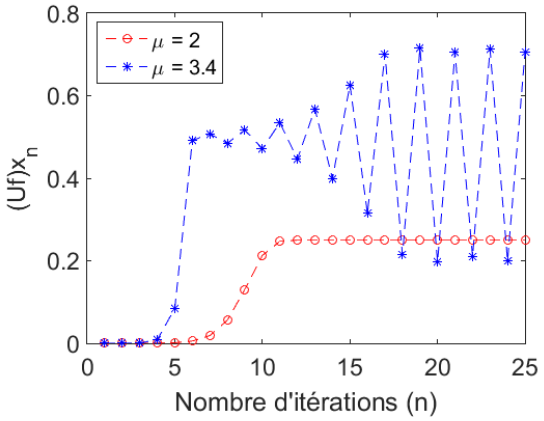
$$Lf = \lim_{t\to 0} \frac{U^t f - f}{t}.$$

Le domaine de L est finalement défini comme les fonctions de $\mathcal{F}$ pour lesquelles la limite précédente existe. Nous pouvons alors reprendre l'exemple de l'application logistique afin de mieux comprendre l'opérateur de Koopman.

Exemple 1.3. *Considérons le système dynamique discret que nous avons introduit précédemment, à savoir l'application logistique (1.2) et nous considérons également une condition initiale $x_0 \in [0, 1]$. Le comportement de ce système dynamique dépend de la valeur du paramètre μ. Considérons deux cas différents : le premier où la suite x_n va converger vers une unique valeur ($\mu = 2$), et le deuxième cas où la suite va finir par osciller entre deux valeurs ($\mu = 3.5$). Regardons alors dans chacun de ces deux cas l'effet de l'opérateur de Koopman. Pour chacun des deux cas, traitons le cas de la fonction $f(x) = x^2$.*



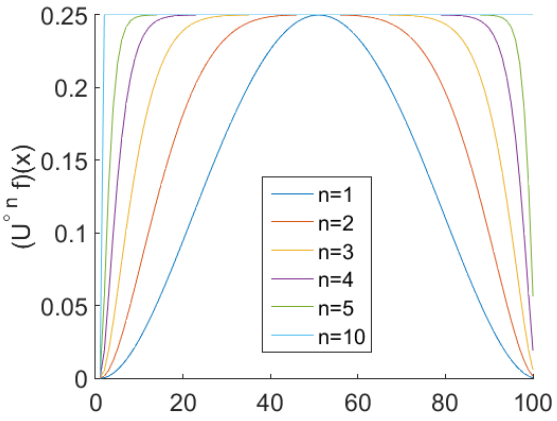
(a) Diagramme de bifurcation de l'application logistique [14]



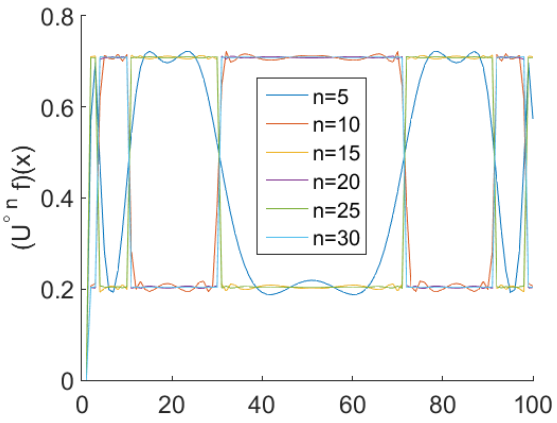
(b) Illustration de l'effet de l'opérateur de Koopman ponctuellement

FIGURE 1.3 – Illustration de l'exemple (1.2)

Dans un premier temps, arrêtons-nous sur la complexité de l'application logistique. Comme évoqué précédemment, cette application change de comportement lorsque le paramètre μ est modifié. Cela est représenté via la FIGURE *1.3a qui représente le diagramme de bifurcation*[2] *de l'application. Autrement dit, ce graphe représente le nombre de points vers lesquels la fonction peut se stabiliser (en fonction de la condition initiale) et leurs valeurs. Dans l'exemple de l'application logistique, le système va osciller entre ces différentes valeurs. Ainsi, sur base du graphe, on s'attend à ce que pour μ qui vaut 2 et 3.4, le système converge vers une valeur et oscille entre deux valeurs respectivement. Pour s'en convaincre, considérons une condition initiale $x_0 = 0.0025$ et appliquons n fois l'opérateur de Koopman. Le résultat obtenu est illustré via la* FIGURE *1.3b. Nous pouvons voir que pour une valeur de $\mu = 2$, nous allons simplement converger vers $f(x^*)$ où x^* est le point vers lequel la suite x_n converge. Dans ce cas-là, x_n tend vers 0.5 et par conséquent,*



(a) $\mu = 2$



(b) $\mu = 3.4$

FIGURE 1.4 – Illustration de l'effet de l'opérateur de Koopman pour différentes valeurs du paramètre μ

2. Une bifurcation intervient lorsqu'un petit changement d'un paramètre physique produit un changement majeur dans l'organisation du système. [15]

$(Uf)x_n$ converge vers 0.25. *De manière globale, la fonction va converger vers une fonctions constante. (Voir* FIGURE *1.4) Cela signifie que quelle que soit la condition initiale considérée, on converge vers le point fixe. Quant au second cas pour lequel le paramètre $\mu = 3.4$, nous observons sur la* FIGURE *1.3b que la fonction oscille entre deux valeurs. En définitive, la fonction ne prend que deux valeurs (la fonction évaluée aux deux points entre lesquels la suite x_n oscille). La fonction capture donc la propriété de périodicité 2 du système. Elle définit également les deux intervalles de conditions initiales qui seront en phase sur l'orbite périodique. L'opérateur de Koopman fait donc évoluer la fonction sur laquelle il s'applique le long des trajectoires du système dynamique, et il permet de capturer certaines propriétés de la dynamique.*

Comme nous l'avons déjà dit, il n'est pas préférable de travailler en dimension infinie. Toutefois, comme l'illustre cet exemple, l'opérateur de Koopman peut parfois simplifier l'étude d'un système. La propriété suivante va nous conforter dans cette idée.

Propriété 1.1

L'opérateur de Koopman est un opérateur linéaire, c'est-à-dire que pour tous $\alpha_1, \alpha_2 \in \mathbb{R}$ et toutes fonctions $f_1, f_2 \in \mathcal{F}$, on

$$U^t\left(\alpha_1 f_1 + \alpha_2 f_2\right) = \alpha_1 U^t f_1 + \alpha_2 U^t f_2.$$

Démonstration. Soient $\alpha_1, \alpha_2 \in \mathbb{R}$, et deux observables f_1 et $f_2 \in \mathcal{F}$. Appliquons tout d'abord la définition de l'opérateur de Koopman. Nous obtenons alors

$$\begin{aligned} U^t\left(\alpha_1 f_1 + \alpha_2 f_2\right) &\underset{(1.5)}{=} \left(\alpha_1 f_1 + \alpha_2 f_2\right) \circ \varphi^t, \\ &= \alpha_1 f_1 \circ \varphi^t + \alpha_2 f_2 \circ \varphi^t, \end{aligned}$$

par linéarité de la composition de fonctions. Alors, si on utilise encore une fois la définition de l'opérateur de Koopman, on retrouve le résultat attendu. Autrement dit,

$$\alpha_1 f_1 \circ \varphi^t + \alpha_2 f_2 \circ \varphi^t \underset{(1.5)}{=} \alpha_1 U^t(f_1) + \alpha_2 U^t(f_2).$$

□

Evidemment, la propriété de linéarité est également satisfaite dans le cas discret, et la démonstration est similaire. Cette propriété est très utile puisque de nombreux résultats ont déjà été développés dans le cs linéaire. De plus, nous pouvons définir les valeurs propres et fonctions propres de l'opérateur de Koopman.

Définition 1.2

Une observable $\phi_\lambda \in \mathcal{F}$ est une **fonction propre** de l'opérateur de Koopman $U^t : \mathcal{F} \to \mathcal{F}$ s'il existe une valeur $\lambda \in \mathbb{C}$ telle que

$$\begin{aligned} U^t \phi_\lambda(x) &= \left(\phi_\lambda \circ \varphi^t\right)(x), \\ &= e^{\lambda t} \phi_\lambda(x), \end{aligned} \tag{1.6}$$

pour tout $x \in X$. Dans ce cas, la valeur λ est appelée la **valeur propre associée**.

Si l'espace fonctionnel $\mathcal{F}$ ne contient que des fonctions continuement différentiables, c'est-à-dire $\mathcal{F} \subset C^1(X)$, alors les fonctions propres de l'opérateur de Koopman sont également des fonctions propres du générateur infinitésimal. En effet, soit $\phi_\lambda(x)$ une fonction propre de l'opérateur de Koopman associée à la valeur propre λ, on a

$$L\phi_\lambda(x) \;=\; \lambda\phi_\lambda(x),$$

autrement dit, ϕ_λ est bien une fonction propre de L.

1.3 Approximation finie de l'opérateur

Actuellement, de nombreux problèmes sont résolus numériquement. Or, l'opérateur de Koopman est de dimension infinie. Il est donc nécessaire d'obtenir une représentation finie de l'opérateur, c'est-à-dire que nous devons approximer l'opérateur dans une dimension finie. Pour ce faire, de nombreuses méthodes peuvent être utilisées. Certaines se basent sur les trajectoires du système dynamique considéré, et d'autres reposent uniquement sur un ensemble de paires de données qui appartiennent à la trajectoire, c'est-à-dire

$$\{x_k\,,\,y_k\}_{k\,=\,1,\ldots,K} \tag{1.7}$$

où K est le nombre de données disponibles. Puisque les données appartiennent à la trajectoire, cela implique que, dans le cas continu, un couple (x_k, y_k) satisfait que $x_k = x(t)$ et $y_k = x(t + \delta t)$. Autrement dit, x_k et y_k sont issus de la même trajectoire et y_k a été mesuré après un laps de temps δt après x_k. Dans le cas discret, on supposera simplement que y_k suit x_k, c'est-à-dire $y_k \;=\; x_{k+1}$.

Parmis les méthodes basées sur un ensemble de paires de données se trouve la méthode de décomposition en modes dynamiques, généralement notée DMD. Celle-ci consiste à établir une relation linéaire entre les données $\{y_k\}$ et $\{x_k\}$, bien que les données soient initialement régies par une dynamique non-linéaire. L'objectif est donc de trouver une matrice $\tilde{U} \in \mathbb{R}^{n\times n}$ où n est la dimension de l'espace X et donc la dimension de chacun des x_k et y_k, de sorte que

$$y_k \;=\; x_k\tilde{U}.$$

La matrice $\tilde{U}$ est donc la représentation finie de l'opérateur de Koopman. Cependant, puisque la relation reliant x_k à y_k est initialement non-linéaire, il est fort probable que cette approximation soit très mauvaise. C'est la raison pour laquelle la méthode de décomposition en modes dynamiques étenduée, appelée EDMD, est généralement préférée. Le principe est relativement simple puisqu'il suppose de "lifter" les états dans un espace de dimension supérieure $N \gg n$. Cela permet de créer de nouvelles composantes, qui seront en réalité des fonctions non-linéaires des données. Formellement, cela consiste à définir des fonctions de lift ψ.

Définition 1.3

Une **fonction de lift** ψ est une fonction définie par

$$\begin{aligned} \psi : \mathbb{R}^n &\to \mathbb{R}^N \\ x &\mapsto \psi(x) \underset{\text{déf}}{=} (\psi_1(x), \ldots, \psi_N(x)), \end{aligned}$$

où $\psi_i : \mathbb{R}^n \to \mathbb{R}$ pour $i = 1, \ldots, N$ sont généralement non-linéaires, et $N \gg n$.

En utilisant ces fonctions de lift, nous pouvons définir un nouvel ensemble de données à partir des données précédentes, à savoir

$$\{X_k\,,\, Y_k\}_{k=1,\ldots,K} \underset{\text{déf}}{=} \{\psi(x_k)\,,\, \psi(y_k)\}_{k=1,\ldots,K}\,.$$

Puisque les données X_k et Y_k contiennent des combinaisons non-linéaires des données initiales, il semble plus probable qu'il existe une relation linéaire entre les données Y et les données X. Généralement, les n premières composantes de la fonction de lift sont les n composantes de l'état, c'est-à-dire

$$\psi_i(x) = x_i,\ \forall i = 1, \ldots, n.$$

Quant aux $N-n$ autres composantes, ce sont des fonctions non-linéaires des composantes. Par exemple, on pourrait avoir que $\psi_{N+1}(x) = x_1 x_2$. Nous cherchons donc la matrice $\tilde{U} \in \mathbb{R}^{N\times N}$, représentative de l'opérateur de Koopman (ou de semi-groupe d'opérateurs dans le cas continu), qui satisfait la relation

$$[\psi_1(x)\,,\,\ldots\,,\,\psi_N(y)] = [\psi_1(x)\,,\,\ldots\,,\,\psi_N(x)]\,\tilde{U}.$$

Pour que ce système linéaire admette une solution, il faut nécessairement qu'il y ait plus de données (K est le nombre de lignes) que la dimension du système lifté (N est le nombre de colonnes). On suppose donc que $N \gg K$. Pour résoudre ce type de systèmes, différentes méthodes existent mais la plus commune reste toutefois la méthode des moindres carrés. En effet, l'action de l'opérateur de Koopman, c'est-à-dire $(Uf)(x_k) = f(x_{k+1})$, peut être réécrite comme

$$\psi(x_{k+1}) = \psi(x_k)\tilde{U} + r(x_k),$$

où $r(x_k)$ est le résidu. La résolution par les moindres carrés a donc pour objectif de déterminer la matrice $\tilde{U}$ telle que la fonction objectif définie par

$$\begin{aligned} J &= \frac{1}{2}\sum_{k=1}^{K-1} |r(x_k)|^2\,, \\ &= \frac{1}{2}\sum_{k=1}^{K-1} \left|\psi(x_{k+1}) - \psi(x_k)\,\tilde{U}\right|^2. \end{aligned}$$

soit minimisée. Alors, la solution de ce problème est donnée par

$$\tilde{U} \;=\; \begin{pmatrix} \psi_1(x_1) & \dots & \psi_N(x_1) \\ \vdots & \ddots & \vdots \\ \psi_1(x_K) & \dots & \psi_N(x_K) \end{pmatrix}^{\dagger} \begin{pmatrix} \psi_1(y_1) & \dots & \psi_N(y_1) \\ \vdots & \ddots & \vdots \\ \psi_1(y_K) & \dots & \psi_N(y_K) \end{pmatrix},$$

où † désigne la matrice pseudo-inverse de Moore-Penrose. Pour rappel, cette matrice est définie de la manière suivante.

Définition 1.4

Soit A une matrice à coefficients réels ou complexes avec n lignes et p colonnes. La **matrice pseudo-inverse de Moore-Penrose** de A notée $A^\dagger$ est l'unique matrice avec p lignes et n colonnes obtenue grâce à

$$A^\dagger \;=\; \lim_{\delta\to 0} \left(A^*A \,+\, \delta I\right)^{-1} A^* \;=\; \lim_{\delta\to 0} A^* \left(AA^* \,+\, \delta I\right)^{-1}.$$

où la matrice A^* dénote la matrice transposée conjuguée de la matrice A. Si la matrice A est de rang plein, alors $A^\dagger$ peut être obtenue via une simple formule algébrique. En particulier, si la matrice A admet des colonnes linéairement indépendantes, alors $A^\dagger$ est obtenu par

$$A^\dagger \;=\; (A^*A)^{-1}A^*.$$

Si A admet des lignes linéairement indépendantes, alors $A^\dagger$ est obtenue par

$$A^\dagger \;=\; A^*(AA^*)^{-1}.$$

Notons que $\tilde{U}$ est la représentation de l'opérateur dans la base définie par les fonctions de lift. En particulier, chaque colonne donne les coordonnées dans la base de l'image d'une fonction de base par l'opérateur.

Utilisons cette méthode d'approximation finie de l'opérateur avec l'exemple de l'oscillateur de Van der Pol.

Exemple 1.4. *Considérons l'oscillateur de Van der Pol dont la dynamique est définie par le système suivant*

$$\begin{cases} \dot{x}_1 &=& 2\,x_2, \\ \dot{x}_2 &=& -0.8\,x_1 + 2\,x_2 - 10\,{x_1}^2 x_2. \end{cases} \tag{1.8}$$

Ce système est fréquemment utilisé pour illustrer ou tester de nouvelles méthodes, comme dans l'article [9] de Korda et Mezic. C'est la raison pour laquelle nous choisissons ici ce système pour illustrer les méthodes de décomposition en modes dynamiques.

Commençons dans un premier temps par la méthode d'approximation finie la plus simple, c'est-à-dire sans fonctions de lift. Puisque la méthode se base sur un ensemble de

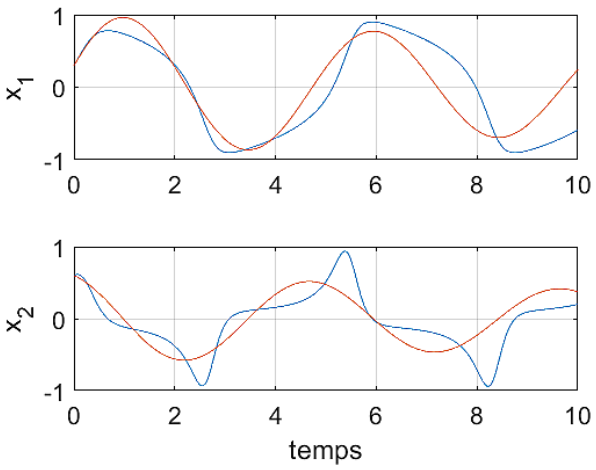
données, la première étape consiste à créer ces données. Pour ce faire, nous pouvons par exemple considérer une condition initiale et intégrer le système numériquement à partir de ce point. Notons toutefois que, de manière générale, les données utilisées pourla méthode (E)DMD ne doivent pas nécessairement être issues de la même condition initiale. Remarquons également que dans la pratique, cette étape n'est pas nécessaire et surtout généralement impossible puisqu'a priori, on ne connait pas la dynamique qui régit le phénomène que l'on étudie. Dans notre cas, considérons le point $x_0 = (0.3\,,\,0.6)$ *comme condition initiale de la trajectoire intégrée jusqu'au temps* 10 *par pas d'intégration de* 0.01. *L'approximation finie de l'opérateur de Koopman par la méthode DMD est donnée par*

$$\tilde{U}^t = \begin{pmatrix} 0.999919923615117 & 0.019992003649274 \\ -0.008004229327996 & 0.999040389611660 \end{pmatrix}$$

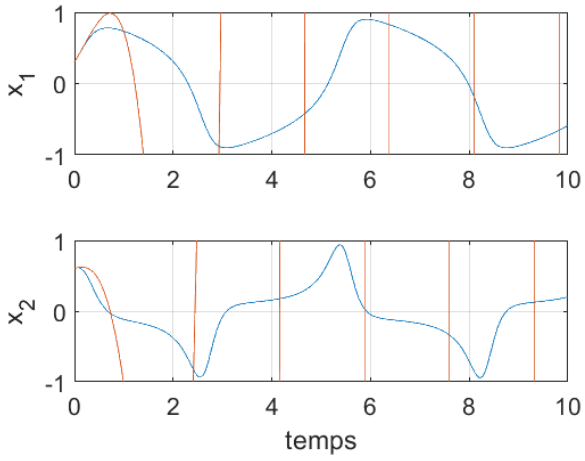
Cette approximation linéaire n'est pas si mauvaise, comme on le voit à la Figure 1.5a Toutefois, la méthode étendue devrait logiquement donner de meilleurs résultats. Dans un premier temps, il est nécessaire de définir les fonctions de lift. Par exemple, considérons les fonctions de x_1 *et* x_2 *d'ordre* 3 *maximum, c'est-à-dire* $\psi_0(x_1, x_2) = 1$ *et*

$$\begin{array}{lll} \psi_1(x_1,x_2) = x_1 & \psi_2(x_1,x_2) = x_2 & \psi_1(x_1,x_2) = {x_1}^2 \\ \psi_4(x_1,x_2) = x_1x_2 & \psi_5(x_1,x_2) = {x_2}^2 & \psi_6(x_1,x_2) = {x_1}^3 \\ \psi_7(x_1,x_2) = {x_1}^2x_2 & \psi_8(x_1,x_2) = x_1{x_2}^2 & \psi_9(x_1,x_2) = {x_2}^3 \end{array}$$

Le nouvel état lifté se trouve donc dans un espace de dimension $N = 10$. *Nous nous attendons donc à trouver une matrice de dimension* 10×10. *Nous obtenons*



(a) décomposition en modes dynamiques (DMD)



(b) décomposition étendue en modes dynamiques (EDMD)

FIGURE 1.5 – Comparaison de la trajectoire obtenue par intégration (bleue) et par l'opérateur de Koopman (rouge) via les méthodes de

1.000	−0.000	−0.000	−0.000	−0.011	−0.009	−0.068	0.001	0.003	0.004
−0.000	0.999	−0.008	0.000	0.000	−0.001	0.693	0.007	−0.001	0.003
−0.000	0.020	1.020	0.000	0.013	0.018	0.068	0.009	−0.017	−0.017
0.000	0.000	0.000	1.000	0.007	0.005	0.051	−0.001	−0.005	−0.005
−0.000	0.000	0.000	0.001	0.049	1.042	0.152	−0.003	−0.007	−0.008
0.000	−0.000	−0.000	0.039	0.952	−0.076	−0.168	0.004	0.009	0.010
−0.000	0.000	0.000	−0.000	−0.012	−0.021	0.066	−0.006	0.033	0.548
0.000	−0.001	−0.098	−0.000	−0.022	−0.039	0.834	0.934	−0.004	0.024
−0.000	−0.000	−0.003	0.001	0.027	0.064	−1.569	0.081	0.935	−0.192
0.000	0.000	0.000	−0.000	−0.012	−0.021	0.066	−0.006	0.033	0.548

Les trajectoires de x_1 et x_2 issues de la méthode étendue avec la matrice précédente sont illustrées à la Figure 1.5b.

1.4 Généralisation de l'opérateur à un système ouvert

Jusqu'à présent, nous avons uniquement manipulé l'opérateur de Koopman avec des systèmes fermés, c'est-à-dire des systèmes qui ne dépendent pas de paramètres extérieurs au système en tant que tel. Cependant, il peut être parfois intéressant ou utile de pouvoir agir sur un système et lui donner la trajectoire que l'on souhaite, le rendre stable etc. Cela est faisable en considérant une entrée dans le système, en particulier un contrôle. Evidemment, toute entrée n'est pas nécessairement un contrôle : pensons notamment à des systèmes dépendant de conditions thermiques ou climatiques sur lesquelles nous n'avons aucune prise. Le contrôle de systèmes linéaires a beaucoup été étudié et de nombreuses méthodes ont été élaborées. Les choses se compliquent lorsqu'on manipule des systèmes non-linéaires. L'utilisation de l'opérateur de Koopman est donc propice à la simplication du contrôle non-linéaire puisque nous pourrons finalement utiliser les méthodes initialement développées pour les systèmes linéaires afin de contrôler les systèmes non-linéaires.

Dans un premier temps, il est nécessaire de redéfinir l'opérateur de Koopman pour les systèmes ouverts. Plusieurs méthodes [9], [13] et [17] ont récemment été élaborées. Citons par exemple la méthode de l'état augmenté [9] que nous développerons dans la suite de cette section, et pour laquelle nous généraliserons la méthode étendue de décomposition en modes dynamiques dans la deuxième partie de cette section. L'avantage majeur de cette méthode réside dans le fait que la définition de l'opérateur de Koopman ne change pas, mais c'est bien la définition de l'état qui est modifiée. La deuxième méthode [13] dont nous allons parler diffère largement de la première puisque l'opérateur de Koopman s'applique maintenant à des fonctions dépendant de deux variables : l'état et l'entrée. Finalement nous décrirons brièvement la méthode élaborée dans la référence [17].

1.4.1 Définition de l'opérateur

La première méthode [9] de généralisation de la définition de l'opérateur de Koopman aux systèmes ouverts est à priori la plus intuitive : considérer l'entrée comme un état et ne rien changer à la définition de l'opérateur. En pratique, nous pouvons définir un nouvel état, dit état augmenté, dans lequel l'entrée est incluse, et ensuite appliquer la définition initiale de l'opérateur de Koopman à savoir l'expression (1.4). Malheureusement, la dimension de l'état devient infinie mais certaines méthodes de contrôle sont adaptées à un tel état. Reprenons en détail la mise en place de l'état augmenté, comme elle est décrite dans la référence [9].

Considérons tout d'abord un système dynamique contrôlé, c'est-à-dire une application (non-linéaire) définie par

$$\begin{array}{rccc} T: & X \times \mathcal{U} & \to & X \\ & (x,u) & \mapsto & T(x,u) \end{array} \tag{1.9}$$

dans le cas discret, où x est l'état initial du système et $u \in \mathcal{U}$ est l'entrée du système. Généralement, on considère que l'état appartient à $X = \mathbb{R}^n$, et donc on suppose également que l'entrée u appartient à $\mathbb{R}^m$, autrement dit que le système admet m entrées. La dynamique associée à ce système est alors donnée par

$$\begin{aligned} X_{k+1} &= \tilde{T}(X_k), \\ &= \begin{pmatrix} T(x_k, u_k(0)) \\ \mathcal{S}u_k \end{pmatrix}, \end{aligned}$$

où $\mathcal{S}$ est l'opérateur de lift vers la droite, autrement dit $(\mathcal{S}u)(i) = u(i+1)$. Définissons alors l'état augmenté comme l'état

$$X = \begin{pmatrix} x \\ \mathbf{u} \end{pmatrix} \tag{1.10}$$

appartenant à l'espace étendu, autrement dit l'espace de l'état initial $\mathbb{R}^n$ multiplié par l'espace $l(\mathcal{U})$ de toutes les suites possibles d'entrées notées $\mathbf{u} = (u_0, u_1, \ldots)$ dans $\mathcal{U}$. Ceci est illustré à la FIGURE 1.6. Finalement, il suffit d'appliquer la définition de l'opérateur de Koopman à ce nouvel état.

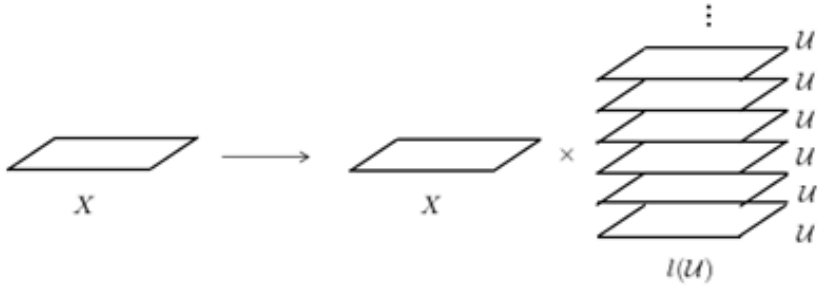


FIGURE 1.6 – Illustration de la méthode de l'état augmenté

Bien que très intuitive, la méthode de l'état augmenté n'est pas très pratique puisqu'il faut manipuler un état en dimension infinie. C'est la raison pour laquelle d'autres méthodes ont été élaborées par la suite, dont celles décrites dans les références [13] et [17]. Décrivons-les brièvement.

Dans un premier temps, la référence [13] traite de la généralisation de la théorie spectrale de l'opérateur de Koopman dans le cadre des systèmes avec entrées et sorties. Une généralisation de la définition de l'opérateur est donc nécessairement introduite, et celle-ci diffère de celle proposée par la référence [9]. Considérons un système dynamique discret avec une entrée défini par l'application (1.9), autrement dit

$$x_{k+1} \;=\; T(x_k\,,\,u_k). \tag{1.11}$$

De plus, nous pouvons considérer un ensemble de fonctions scalaires noté $\mathcal{H}$ de sorte que ces fonctions dépendent soit uniquement de l'état, c'est-à-dire par exemple $f(x\,,\,u) = x_1$, soit uniquement de l'entrée comme $f(x\,,\,u) = u_1$, ou finalement elles dépendent à la fois de l'état et de l'entrée comme par exemple $f(x\,,\,u) = x_1u_1$. L'opérateur de Koopman estalors défini comme l'opérateur $U : \mathcal{H} \to \mathcal{H}$ tel que pour une fonction $g \in \mathcal{H}$ quelconque, on a

$$(Ug)\,(x_k\,,\,u_k) \;=\; g\,(T(x_k\,,\,u_k)\,,\,u_{k+1})\,. \tag{1.12}$$

Cette définition est toutefois très générale. Ils proposent alors dans la référence [13] d'adapter et de spécifier cette définition selon le type d'entrée que l'on manipule. Cette méthode a pour but de bien représenter et dissocier d'une part l'évolution du système dynamique et d'autre part, l'impact du choix du contrôle sur l'évolution du système.

- Supposons que le système soit en boucle fermée comme à la FIGURE 1.7. Dans ce cas, l'entrée u_k dépend de l'état x_k, autrement dit nous pouvons écrire $u_k = h(x_k)$. Alors, l'opérateur de Koopman peut être définie par

$$(Ug)(x_k\,,\,u_k) \;=\; g\,(T(x_k\,,\,u_k)\,,\,h(T(x_k\,,\,u_k)))\,.$$

  Grâce à la relation (1.11), on en déduit alors que

$$(Ug)(x_k\,,\,u_k) \;=\; g\,(x_{k+1}\,,\,h(x_{k+1}))\,.$$

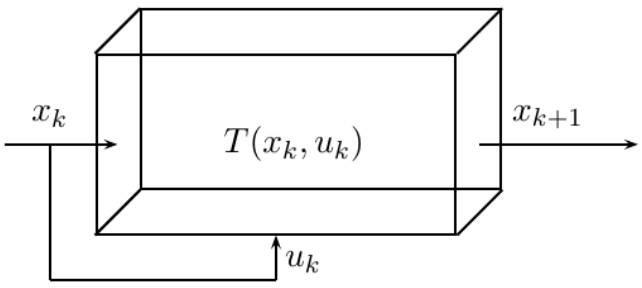


FIGURE 1.7 – Illustration d'un système dynamique discret en boucle fermée

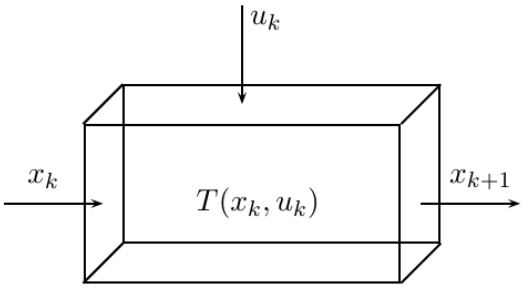


FIGURE 1.8 – Illustration d'un système dynamique discret en boucle ouverte

- Supposons maintenant qu'à l'inverse, le système est en boucle ouverte (voir Figure 1.8). Plusieurs cas peuvent être envisagés : supposons dans un premier temps que l'entrée soit issue d'une force constance c. Nous pouvons alors définir une famille d'opérateurs de Koopman pour chaque entrée c. Dans un second temps, supposons que l'entrée ait sa propre dynamique, autrement dit que l'entrée soit régie par une relation de la forme

$$u_{k+1} \;=\; f_u(u_k).$$

L'action de l'opérateur de Koopman peut alors s'écrire comme

$$(Ug)(x_k\,,\,u_k) \;=\; g\left(T(x_k\,,\,u_k)\,,\,f_u(u_k)\right).$$

Finalement, la méthode de généralisation développée dans la référence [17] part du principe que l'opérateur de Koopman est jusqu'à présent, très utile pour travailler à partir d'un jeu de données. Toutefois, les méthodes d'approximation finie de l'opérateur ne sont initialement établies que pour des données dont la relation sous-jacente ne dépend que de l'état. Elles ne sont alors pas adaptées aux systèmes dynamiques contrôlés par exemple. Il est donc nécessaire de généraliser l'opérateur de Koopman aux systèmes avec entrées dans le but de pouvoir utiliser les méthodes d'approximation finie pour ces mêmes systèmes. Ceci est donc la motivation de la généralisation de la référence [17]. En réalité, cette méthode se base sur les modèles linéaires à paramètre variant (LPV) et celle-ci peut être utilisée puisque l'hypothèse de linéarité, nécessaire pour cette méthode, est bien vérifiée par l'opérateur de Koopman. Toutefois, le système obtenu peut dépendre de manière linéaire du paramètre. A la base, cette méthode traite différemment les paramètres et les entrées.

1.4.2 Généralisation de la méthode EDMD

Comme la dernière méthode de redéfinition de l'opérateur de Koopman aux systèmes ouverts, il pourrait être très utile de généraliser la méthode d'approximation finies de l'opérateur, c'est-à-dire la méthode (E)DMD aux systèmes ouverts. Nous pourrions ainsi approximer en dimension finie un système non-linéaire contrôlé. Dans cette section, nous supposons que nous optons pour la première méthode de généralisation à un système ouvert, autrement dit la méthode de l'état augmenté et nous nous basons sur la référence [9]. Le système que nous traitons est donc de la forme

$$X_{i+1} \;=\; \tilde{T}(X_i).$$

Procédons exactement de la même manière que pour la méthode initiale de décomposition en modes dynamiques. Pour ce faire, supposons que nous avons un ensemble de paires de données

$$(X_i\,,\,Y_i),$$

pour $i \;=\; 1,\ldots,K$ et de sorte que les données satisfont pour chacun des i que

$$\begin{aligned} Y_i \;&=\; \tilde{T}(X_i), \\ &\underset{\text{not}}{=}\; X_{i+1}, \end{aligned}$$

comme nous l'avons fait pour la méthode initiale. L'objectif de la méthode est de trouver la matrice V qui minimise les résidus, autrement dit l'expression suivante

$$\sum_{i\,=\,1}^{K} \parallel \psi(Y_i) - VX_i \parallel_2^2,$$

où $\psi(X) \;=\; \big(\psi_1(X),\ldots,\psi_{N_\psi}(X)\big)^t$ est le vecteur de lift, et de sorte que pour tout i, la fonction $\psi_i : \mathbb{R}^n \times l(\mathcal{U}) \to \mathbb{R}$. Toutefois, puisque l'état augmenté est de dimension infinie, l'expression $\psi(X)$ pourrait être compliquée à calculer, à moins que les fonctions ψ_i soient idéalement choisies. En pratique, nous allons imposer que les fonctions de lift soient de la forme

$$\psi_i(X) \;=\; \rho(x) + \mathcal{L}_i(u), \tag{1.13}$$

où la fonction $\mathcal{L}_i :\; l(\mathcal{U}) \to \mathbb{R}$ est linéaire. De plus, nous pouvons supposer sans perdre de généralité que la dimension de l'espace de lift est donnée par $N_\psi \;=\; N + m$ et que la fonction de lift est de la forme

$$\psi(x,u) \;=\; (\rho_1(x)\,,\,\ldots\,,\,\rho_N(x)\,,\,u_1(0)\,,\,\ldots\,,\,u_m(0))^t\,.$$

Ici, nous ne sommes pas encore interessés par les suites de contrôle, nous pouvons donc négligler les m dernières composantes de chaque élément de la somme considérée. Ainsi, si nous gardons uniquement les N premières lignes de V et que l'on décompose cette matrice en deux, à savoir $A \in \mathbb{R}^{Ntimes N}$ et $B \in \mathbb{R}^{N\times m}$ qui s'appliquent respectivement sur $\rho(x)$ et $u_i(0)$, alors nous pouvons réécrire la somme des résidus comme

$$\sum_{i\,=\,1}^{K} \parallel \rho(y_i) - A\rho(x_i) - Bu_i(0) \parallel_2^2\,. \tag{1.14}$$

Chapitre 2

Prédiction de systèmes non-linéaires

A l'heure actuelle, peu de méthodes de contrôle sont conçues pour les systèmes non-linéaires tandis que de nombreuses méthodes existent pour les systèmes linéaires. Toutefois, nous savons que, grâce à l'opérateur de Koopman, tout système dynamique non-linéaire de dimension finie est équivalent à un système dynamique linéaire de dimension infinie. Dès lors, si nous parvenons à l'approximer en dimension finie, nous pouvons ainsi régler le problème et utiliser uniquement les méthodes linéaires. L'objectif de cette section est donc de développer deux méthodes de prédiction linéaire de systèmes non-linéaires sur base de l'opérateur de Koopman, et de donner quelques exemples de méthodes de contrôle. En particulier, les méthodes développées par la suite sont celles des références [9] et [13] ainsi que les méthodes de contrôle.

2.1 Méthode 1 : construction d'un prédicteur linéaire

La première méthode que nous allons développer ici est la méthode décrite dans la référence [9] de 2016. Celle-ci s'incrit dans un processus en deux étapes, dit commande prédictive, pour contrôler un système dynamique non-linéaire dont la dynamique est nécessairement inconnue. La première étape consiste en la construction d'un prédicteur linéaire ou bilinéaire du système dynamique initial tandis que la deuxième étape utilise une commande prédictive à partir de l'approximation (bi)linéaire. Dans cette section, nous allons nous concenter sur la méthode de prédiction linéaire et le contrôle sera brièvement repris dans la troisième section de ce chapitre.

Considérons un système dynamique contrôlé discret défini par

$$x_{k+1} \;=\; T(x_k\,,\,u_k).$$

L'objectif ici est de trouver un prédicteur de ce système en ne connaissant que la condition initiale x_0 et la suite d'entrées $\{u_0\,,\,u_1\,,\,\ldots\}$. En particulier, nous souhaitons obtenir un prédicteur linéaire. Ainsi, la forme que l'on recherche est un système dynamique linéaire contrôlé de la forme

$$\left\{\begin{array}{rcl} z_{k+1} &=& Az_k + Bu_k, \\ \tilde{x}_k &=& Cz_k. \end{array}\right. \tag{2.1}$$

où $z \in \mathbb{R}^N$ est l'état lifté introduit dans le chapitre précédent, $\tilde{x}$ est la prédiction de l'état initial x et les matrices $B \in \mathbb{R}^{N\times m}$ et $C \in \mathbb{R}^{N\times N}$. Notons que l'approximation considérée suggère que les fonctions de lift ne dépendent pas de l'entrée u. Si on dénote par x_0 la condition initiale, alors la condition initiale associée au prédicteur est définie par

$$\begin{aligned} z_0 &= \rho(x_0), \\ &= \begin{pmatrix} \rho_1(x_0) \\ \vdots \\ \rho_N(x_0) \end{pmatrix}, \end{aligned}$$

où les fonctions ρ_i sont les fonctions de lift préalablement définies. Remarquons que le choix d'un prédicteur linéaire est assez réducteur, et qu'un prédicteur plus complexe pourrait être envisagé. Par exemple, un système dynamique bilinéaire de la forme

$$\begin{cases} z_{k+1} &= Az_k + (Bz_k)u_k, \\ \tilde{x}_k &= Cz_k. \end{cases}$$

pourrait être préféré au cas linéaire. Que ce soit le modèle linéaire ou bilinéaire, chacun de ces prédicteurs est uniquement défini à partir des matrices A, B et C. L'objectif est donc de trouver les matrices adéquates. Pour ce faire, nous allons utiliser la généralisation de la méthode EDMD aux états augmentés que nous avons développée dans le chapitre précédent. Il suffit de minimiser l'expression (1.14) sur A et B pour obtenir ces deux matrices à partir de la condition initiale z_0. Cependant, nous souhaitons pouvoir utiliser ces méthodes numériquement. Il est donc nécessaire de mettre au point un algorithme numérique associé à la méthode décrite cidessus. Reprenons l'algorithme proposé dans la référence [9].

Supposons que nous possédons trois ensembles de données définis par

$$X = (x_1, \ldots, x_K), \quad Y = (y_1, \ldots, y_K), \quad \text{et} \quad U = (u_1, \ldots, u_K),$$

de sorte qu'ils respectent la dynamique du système, autrement dit pour tout $1 \leq i \leq K$, les données satisfont que $y_i = T(x_i, u_i)$. Puisque nous utilisons la méthode étendue de la décomposition en modes dynamiques, nous devons considérer les états liftés. Pour ce faire, dénotons-les par X_{lift} et Y_{lift} tels que

$$X_{\text{lift}} = (\rho(x_1), \ldots, \rho(x_K)) \quad \text{et} \quad Y_{\text{lift}} = (\rho(y_1), \ldots, \rho(y_K)),$$

où $\rho(x) = (\rho_1(x), \ldots, \rho_N(x))$, avec ρ_i les fonctions de lift précédemment définies. Comme nous l'avons expliqué au chapitre précédent, nous recherchons les matrices A et B qui satisfont au mieux le prédicteur (2.1), en particulier ces matrices doivent être solution du problème d'optimisation (1.14). Autrement dit, si on exprime le problème avec les matrices de lift, nous devons minimiser l'expression suivante

$$\min_{A,B} \| Y_{\text{lift}} - AX_{\text{lift}} - BU \|_2 .$$

En d'autres mots, nous cherchons le meilleur prédicteur au sens des moindres carrés. La solution à ce problème est donnée par

$$(A\,,\,B) \;=\; Y_{\text{lift}}\,(X_{\text{lift}}\,,\,U)^{\dagger}\,.$$

Finalement, il reste encore à déterminer la matrice $C \in \mathbb{R}^{n\times N}$ du modèle linéaire (2.1) de prédicteur. De manière similaire à la recherche des matrices A et B, nous cherchons la matrice C définie comme la meilleure estimation linéaire au sens des moindres carrés de X, c'est-à-dire C est obtenue en résolvant le problème suivant

$$\min_{C} \parallel X_{\text{lift}} - CX \parallel_2 .$$

On choisit de définir la matrice C comme la meilleure projection de x sur l'espace généré par les fonctions ρ_i (au sens des moindres carrés). Ainsi, C est obtenu en minimisant l'expression

$$\sum_{i=1}^{K} \parallel x_i - C\rho(x_i) \parallel_2^2 .$$

Analytiquement, la matrice C est donnée par $XX_{\text{lift}}^{\dagger}$.

Illustrons cette méthode avec un exemple déjà introduit préalablement, à savoir l'oscillateur de Van der Pol. Supposons toutefois qu'il existe une entrée supplémentaire et tentons d'approximer linéairement ce nouveau système.

Exemple 2.1. *Soit l'oscillateur de Van der Pol, et supposons qu'il soit forcé. Cela signifie que la dynamique du système est donnée par*

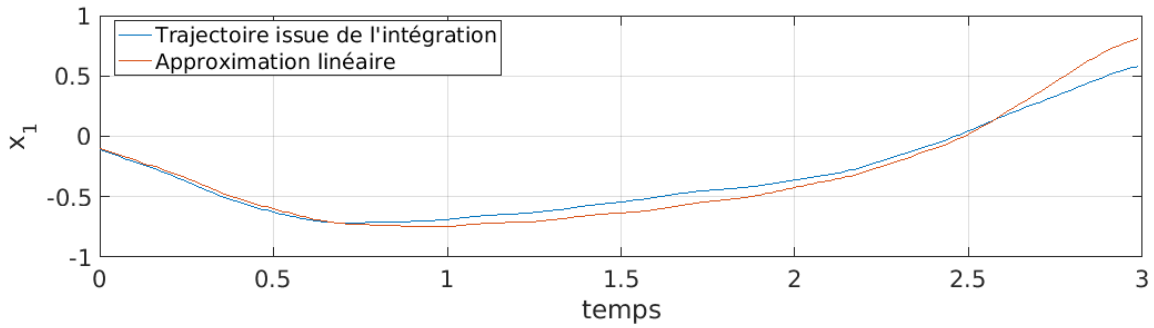
$$\begin{cases} \dot{x}_1 &= 2\,x_2, \\ \dot{x}_2 &= -0.8\,x_1 + 2\,x_2 - 10\,x_1{}^2 x_2 + u. \end{cases} \tag{2.2}$$

Ce système est étudié dans la référence [9] où les auteurs y supposent que l'entrée est un signal binaire pseudo-aléatoire[1]*. De plus, les fonctions de lift sont choisies de la manière suivante : les deux premières sont les états, autrement dit* $\rho_1(x) = x_1$ *et* $\rho_2(x) = x_2$*. Les cent autres fonctions de lift sont des fonctions de base radiale dont les centres sont déterminés aléatoirement dans la boite unité* $[-1\,,\,1]^2$*, autrement dit les fonctions de lift s'expriment de la forme suivante*

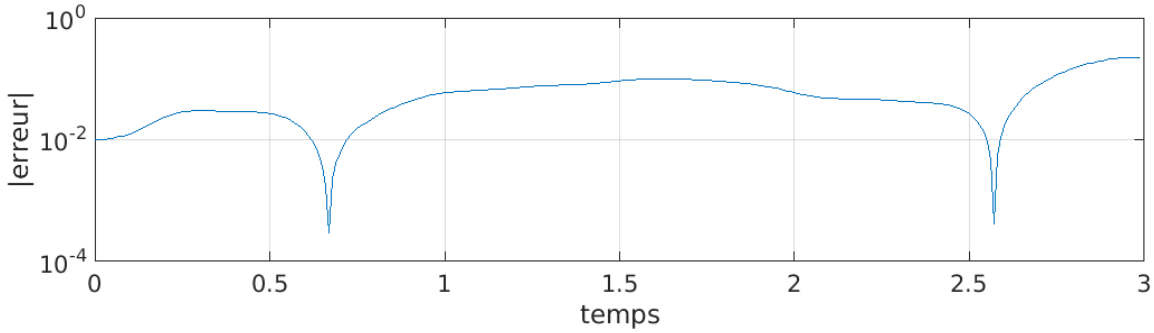
$$\rho_i(x) \;=\; r^2 \log(r),$$

où r *désigne la norme 2 de la différence entre* x *et le centre. D'où, l'espace lifté est de dimension* 102*. Finalement, nous utilisons un pas d'intégration de* 0.01s *pour générer les données, et nous intégrons durant* 10 *secondes. Ainsi, nous possédons* $K = 1000$ *paires de données. Dans l'exemple repris aux* FIGURES *2.1 et 2.2 pour les composanes* x_1 *er* x_2 *respectivement, nous avons généré les données à partir de cent trajectoires différentes. Nous avons alors pu construire les matrices* $A \in \mathbb{R}^{102\times 102}$*,* $B \in \mathbb{R}^{102\times 1}$ *et* $C \in \mathbb{R}^{2\times 102}$ *qui définissent le prédicteur linéaire. Au vu de leur taille, il est évidemment qu'il serait difficile et inutile de les présenter ici. Toutefois, nous pouvons tout de même observer*

1. Dans Matlab, la commande $unifrnd(a, b)$ envoie un tableau de nombres aléatoires à partir d'une distribution uniforme continue dont les bornes supérieure et inférieure sont spécifiées par a et b.

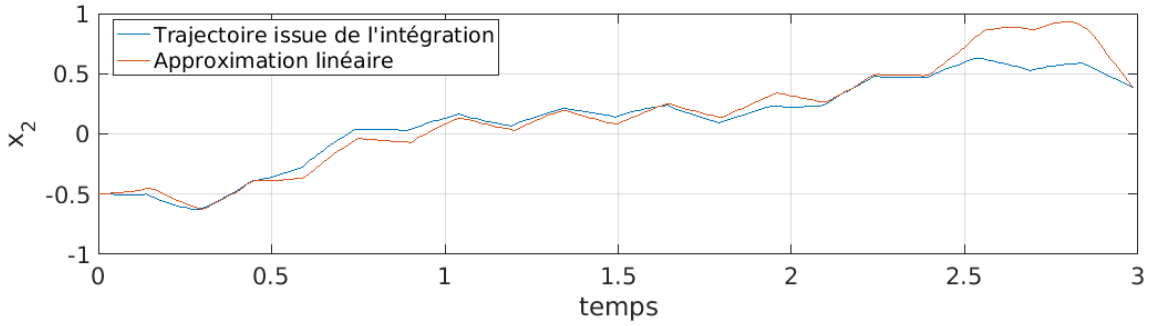


(a) Trajectoire issue de l'intégration en bleu, et approximation linéaire en rouge

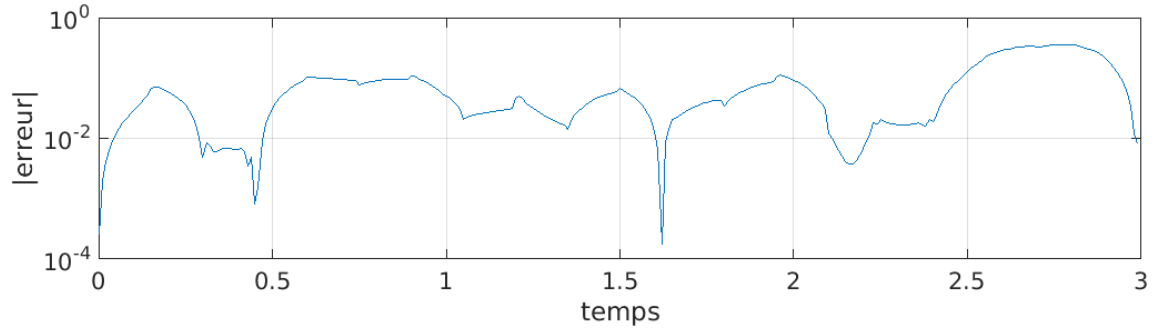


(b) Valeur absolue de l'erreur entre la trajectoire et son approximation linéaire

FIGURE 2.1 – Application de la première méthode de prédicteur linéaire issue de la référence [9] au système forcé de Van der Pol (2.2) : analyse de la première composante



(a) Trajectoire issue de l'intégration en bleu, et approximation linéaire en rouge



(b) Valeur absolue de l'erreur entre la trajectoire et son approximation linéaire

FIGURE 2.2 – Application de la première méthode de prédicteur linéaire issue de la référence [9] au système forcé de Van der Pol (2.2) : analyse de la deuxième composante

les résultats que nous obenons. En effet, pour tester l'efficacité du prédicteur, nous avons considéré la condition initiale $(-0.1\,,\,-0.5)$ *et nous avons prédit jusqu'au temps* 3. *Dans un premier temps, la trajectoire réelle a été obtenue par intégration (ode45 sur Matlab) du système dynamique, tandis que l'approximation a été faite via le prédicteur linéaire. On peut voir sur les* FIGURES *2.1b et 2.2b que l'erreur commise se situe globalement entre* 10^{-4} *et* 10^{-1} *sauf sur la fin où pour chacune des deux composantes, l'erreur augmente. Il est important de noter que nous avons fixé les paramètres à certaines valeurs, comme le pas d'intégration, le temps d'intégration, le nombre de fonction de lift, etc. En réalité, ceux-ci pourraient être modifiés pour obtenir une meilleure approximation. C'est exactement l'objet du troisième chapitre puisqu'il est important de pouvoir étudier et connaître la précision de l'approximation.*

2.2 Méthode 2 : linéarisation

La deuxième méthode dont nous allons parler se base sur la linéarisation et est développée dans la référence [13]. Nous avons vu préalablement que nous pouvons trouver une approximation finie de l'opérateur de Koopman et que nous avons pu généraliser cette définition aux systèmes ouverts. Dans le cas d'un système sans entrée, nous savons que nous avons approximativement que

$$x_{k+1} \;\approx\; \tilde{U}^t \psi(x_k)^t.$$

Si nous supposons que les fonctions de lift sont différentiables, nous pouvons utiliser la linéarisation pour obtenir une relation linéaire entre l'état courant et le prochain état, autrement dit nous obtenons

$$\begin{aligned} x_{k+1} &\approx \tilde{U}^t \frac{\partial \psi}{\partial x} x_k, \\ &\approx A(x_k)\, x_k. \end{aligned}$$

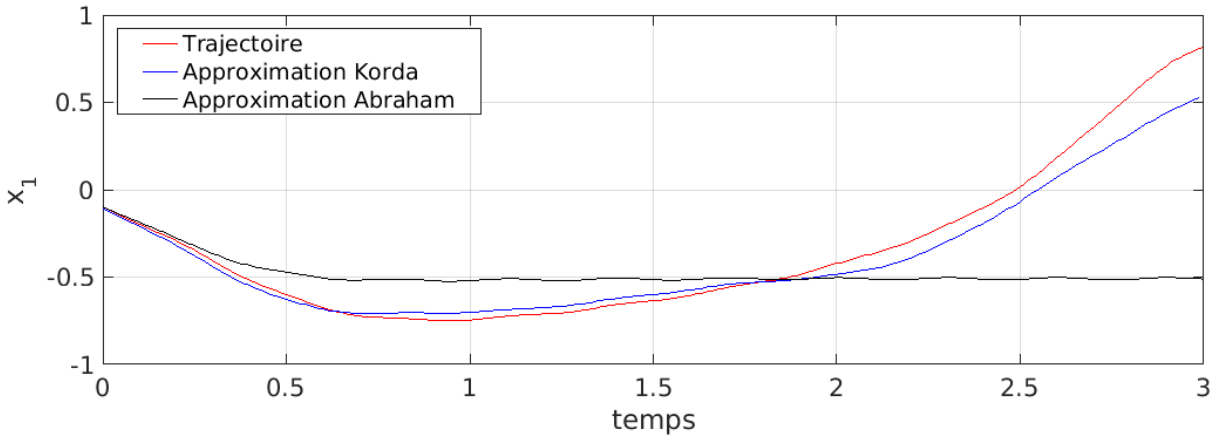
Nous pouvons utiliser exactement le même principe en supposant que les fonctions de lift dépendent maintenant de l'état x_k ainsi que de l'entrée u_k. Autrement dit, nous pouvons supposer qu'elles s'expriment via la forme suivante

$$\psi(x,u) \;=\; \left(x^t\,,\,u^t\,,\,\psi_1(x,u)\,,\,\ldots\,,\,\psi_N(x,u)\right).$$

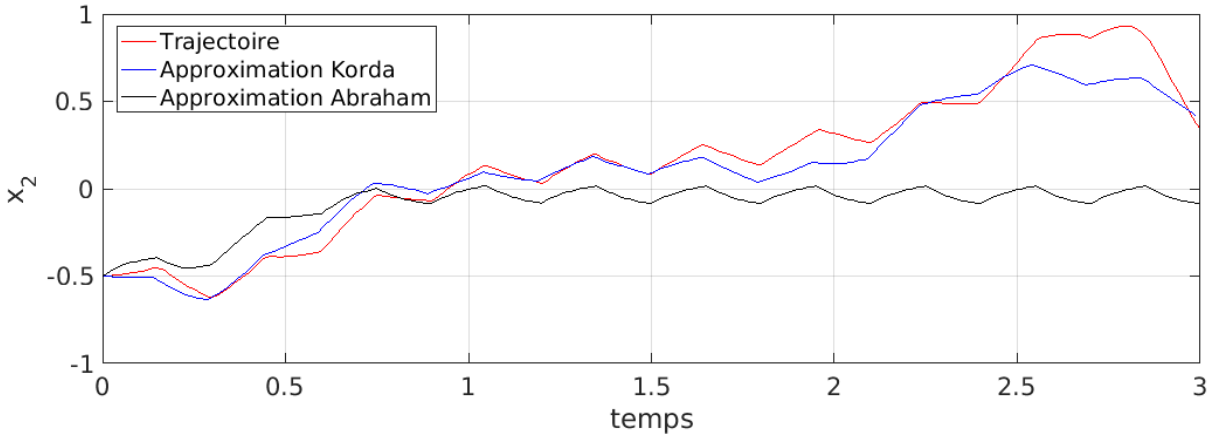
Dès lors, si les fonctions de lift sont différentiables, nous pouvons linéariser et finalement obtenir l'expression suivante

$$\begin{aligned} x_{k+1} &\approx \tilde{U}^t \frac{\partial \psi}{\partial x} x_k + \tilde{U}^t \frac{\partial \psi}{\partial u} u_k, \\ &\approx A(x_k\,,\,u_k) x_k + B(x_k\,,\,u_k) u_k. \end{aligned}$$

Reprenons l'exemple de l'oscillateur de Van der Pol forcé pour illuster cette seconde méthode.



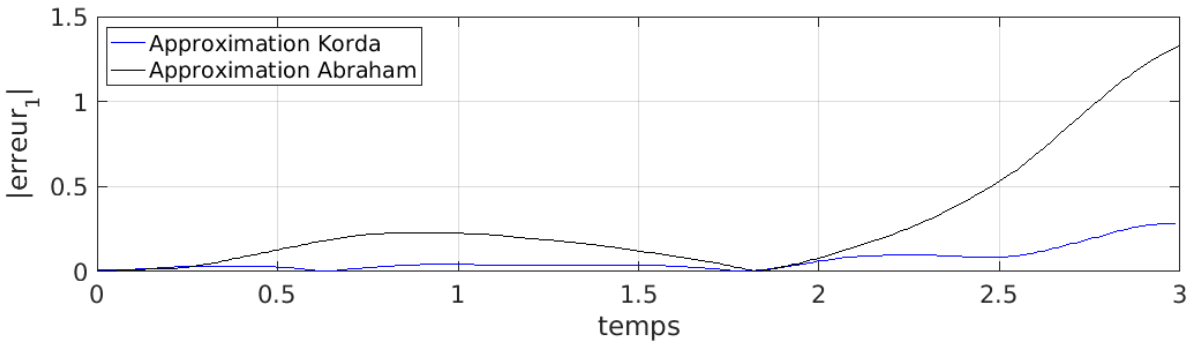
(a) Trajectoire issue de l'intégration en bleu, et approximation linéaire en rouge pour x_1



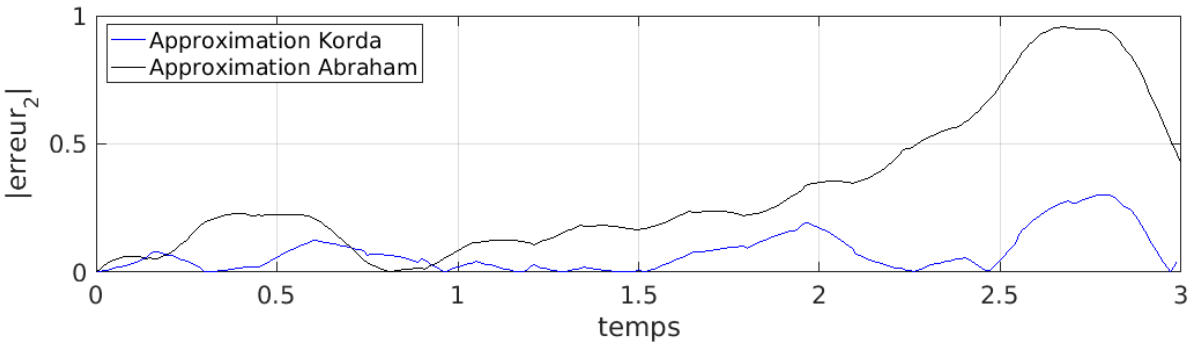
(b) Trajectoire issue de l'intégration en bleu, et approximation linéaire en rouge pour x_2

FIGURE 2.3 – Application de la deuxième méthode de prédicteur linéaire issue de la référence [13] au système forcé de Van der Pol (2.2) : analyse des deux composantes

Exemple 2.2. *Supposons que nous travaillons encore une fois avec l'oscillateur de Van der Pol forcé, autrement dit le système décrit par la dynamique (2.2). La méthode que nous devons de décrire nécessite les dérivées partielles de chacune des fonctions de lift tandis que la méthode précédente n'en avait pas besoin. Cette linéarisation nous laisse penser que ce modèle risque peut-être d'être moins bon que le précédent. L'exemple de la* FIGURE *2.3 a pour condition initiale la même que pour l'exemple précédent, à savoir* $(-0.1\,,\,-0.5)$ *et les prédicteurs ont été construits de la même manière. Alors, on peut voir clairement que l'approximation de la référence [9, Korda et al.] (en bleu), c'est-à-dire la première méthode est globalement meilleure que la deuxième méthode, c'est-à-dire l'approximation de la référence [13, Abraham et al.] (en noir). Ceci est encore plus voyant si nous regardons l'erreur commise pour chacune des méthodes à la* FIGURE *2.4. En effet, que ce soit pour la première composante ou pour la deuxième, l'erreur est globalement plus petite pour la première méthode que pour la deuxième. En réalité, il semble que la seconde méthode est de moins en moins bonne lorsque le temps augmente, autrement dit l'horizon de prédiction doit être plus petit que pour la première méthode.*



(a) Valeur absolue de l'erreur entre la trajectoire et son approximation linéaire pour x_1



(b) Valeur absolue de l'erreur entre la trajectoire et son approximation linéaire pour x_2

FIGURE 2.4 – Application de la deuxième méthode de prédicteur linéaire issue de la référence [13] au système forcé de Van der Pol (2.2) : analyse de l'erreur

2.3 Application au contrôle

Grâce aux deux méthodes développées dans les deux sections précédentes, nous obtenons finalement une approximation linéaire (ou bilinéaire) du système que l'on souhaite étudier. Pour rappel, la première méthode construit un prédicteur linéaire ou bilinéaire tandis que la deuxième méthode a recourt à la linéarisation des fonctions de lift. Chacune des références sur lesquelles nous nous basons propose une méthode pour déterminer un signal de contrôle. En effet, puisque nous obtenons des systèmes linéaires, il est beaucoup plus simple de les contrôler car de nombreuses méthodes ont déjà été developpées. Notons toutefois que le contrôle n'est pas l'objet de ce mémoire puisque nous nous sommes concentrés sur la prédiction, mais que nous allons brièvement décrire dans cette section les contrôles proposés dans les références [9] et [13].

Pour la première méthode, l'auteur de l'article opte pour une commande prédictive, permettant ainsi d'obtenir un contrôle par feedback. Découverte par le français J. Richalet en 1978, cette méthode s'applique à des systèmes relativement complexes et consiste à prédire et anticiper le comportement futur du système. Le modèle proposé doit résoudre pour chaque k le problème d'optimisation suivant

$$\begin{aligned} \underset{u_i, z_i}{\text{minimiser}} \quad & z_{N_p}^t \, P \, z_{N_p} + \sum_{i=1}^{N_p-1} z_i{}^t \, Q \, z_i + u_i{}^t \, R \, u_i + q^t z_i + r^t u_i \\ \text{sous contraites} \quad & z_{i+1} = A z_i + B u_i, \;\; i \, = \, 0 \, , \, \ldots \, , \, N_p - 1 \\ & E z_i + F u_i \, \leq \, b, \qquad i \, = \, 0 \, , \, \ldots \, , \, N_p - 1 \\ & z_0 \, = \, \rho(x_k) \end{aligned}$$

où N_p est l'horizon de prédiction et les matrices $P \in \mathbb{R}^{N \times N}$, $Q \in \mathbb{R}^{N \times N}$ et $R \in \mathbb{R}^{m \times m}$ sont des matrices semi-définies positives dites matrices de coût. Ainsi, on peut résoudre le problème en minimisant par exemple l'énergie. De plus, les matrices $E \in \mathbb{R}^{n_c \times N}$ et $F \in \mathbb{R}^{n_c \times m}$ ainsi que le vecteur $b \in \mathbb{R}^{n_c}$ permettent de définir des contraintes sur l'état z_i et sur l'entrée u_i. Naturellement, n_c n'est rien d'autre que le nombre de contraintes imposées. Finalement, ce problème d'optimisation est paramétrisé par x_k, c'est-à-dire l'état courant du système. Après résolution du problème, on obtient un contrôle par feedback noté $u_0^\star(x_k)$. Qu'il soit linéaire ou non dans l'espace lifté, le contrôle ne l'est généralement pas dans l'espace d'origine. En pratique, si nous connaissons un contrôle dans l'espace lifté dénoté par $K_{\text{lift}} : \mathbb{R}^N \to \mathbb{R}^m$, alors nous pouvons déterminer un contrôle dans l'espace initial via la relation suivante

$$K(x) \, = \, K_{\text{lift}}(\rho(x)).$$

La deuxième méthode quant à elle opte pour une méthode relativement similaire puisqu'elle définit tout d'abord une fonction objectif dépendant de l'état et de l'entrée qu'on souhaite minimiser. Cette fonction s'exprime sous une forme habituelle, c'est-à-dire

$$J \, = \, \sum_{k=0}^{N} \frac{1}{2} \, (x_k - \tilde{x}_k)^t \, P \, (x_k - \tilde{x}_k) + \frac{1}{2} \, u_k{}^t R u_k, \tag{2.3}$$

où $P \in \mathbb{R}^{n \times n}$ et $R \in \mathbb{R}^{m \times m}$ sont des matrices définies positives de poids sur l'état et le contrôle, tandis que $\tilde{x}_k$ est la trajectoire de référence au temps k. Ainsi, la matrice de poids P mesure l'importance de l'état, tandis que la matrice de poids R mesure l'impact du contrôle. Généralement, ces matrices sont diagonales et au plus les éléments diagonaux sont élevés, au plus l'objet auquel la matrice s'applique (l'état ou le contrôle) devra être minimisé. En pratique, si on souhaite obtenir un contrôle relativement faible, la matrice R devrait par exemple, contenir des éléments diagonaux positifs et relativement grands. Ceci est relativement intéressant s'il existe des contraintes sur l'entrée, comme par exemple une limite de puissance d'un moteur. Notons que cette fonction objectif est relativement similaire à celle de la méthode précédente puisqu'elle semble être un cas de particulier de celle-ci.

La référence [13] propose de trouver la solution de ce problème d'optimisation en fonction de la nature du système que nous étudions. En effet, la méthode de résolution est différente si le système est en boucle ouverte ou en boucle fermée :

- Si le système est en boucle ouverte, il est possible de calculer préalablement l'ensemble des trajectoires et entrées qui minimisent la fonction objectif (2.3) La référence [13] préconise d'utiliser la méthode développée dans la référence [7].

- Si le système est en boucle fermée, alors la référence [13] propose d'utiliser une méthode similaire à la méthode précédente puisque c'est une commande prédictive, autrement dit les auteurs commencent tout d'abord par simuler une trajectoire jusqu'à un horizon donné, et ensuite ils déterminent le contrôle à partir de cette simulation. En particulier, ils suggèrent d'utiliser une version discrète d'un *Sequential Action Control*, dit SAC. Les détails de cette méthode sont repris à la page 3 de la référence [13] ainsi que dans la référence [2].

Chapitre 3

Etude de l'erreur de la prédiction

Dans la section précédente, nous avons décrit deux méthodes de prédiction finie et linéaire d'un système dynamique non-linéaire en dimension finie grâce à l'opérateur de Koopman. Toutefois, nous ne connaissons pas la précision de ces prédicteurs. Il est donc important d'étudier ces prédicteurs pour savoir quand les utiliser, quels paramètres choisir pour avoir la meilleure prédiction possible, sur quel horizon la prédiction est satisfaisante, etc. Ces questions sont très importantes puisque, comme nous l'avons remarqué dans le chapitre précédent, ces prédicteurs peuvent être utilisés pour contrôler le système non-linéaire initial. Une erreur trop grande pourrait donc avoir des conséquences néfastes sur le contrôle.

Dans la première section de ce chapitre, nous allons établir une borne théorique de l'erreur que nous commettons en approximant en dimension finie l'opérateur de Koopman. Pour ce faire, nous nous basons principalement sur les résultats obtenus dans la référence [3]. Ensuite, nous étudierons l'erreur commise selon différentes valeurs de paramètres dans la deuxième section de ce chapitre. Une brève conclusion terminera finalement ce chapitre.

3.1 Résultats théoriques

Dans cette section, nous tentons de borner l'erreur commise en utilisant l'opérateur de Koopman, c'est-à-dire en approximant linéairement un système initialement non-linéaire via cet opérateur. Pour ce faire, nous nous basons principapelement sur [3, Theorem 2.4]. Le résultat de ce théorème est repris ci-dessous.

Théorème 3.1

Soit une fonction $f \in C^k[a,b]$ et soit p_n, la projection orthogonale L_2 de f dans P_n, dit $L_2(a,b)$-projection de f. Si $n \geq k-1$, alors l'inégalité suivante est satisfaite :

$$\| f - p_n \|_{L^\infty(a,b)} \leq (2+n)\, a_n \left(\frac{b-a}{2}\right)^k \| f^{(k)} \|_{L^\infty(a,b)},$$

où $a_n = \dfrac{(\pi/2)^k}{(n+1)n\ldots(n-k+2)}$ e $k \in \mathbb{N}_0$.

Notons que dans ce mémoire, nous abordons uniquement le cas à une dimension. Toutefois, les résultats obtenus devraient pouvoir être généralises à n dimensions. Dans notre cas, nous nous intéressons à l'erreur commise due à l'approximation finie de l'opérateur de Koopman. Autrement dit, l'erreur que nous devons étudier est défine par

$$\| Uf - PUf \| .$$

Or, si nous voulons utiliser le Théorème 3.1, nous devons nécessairement considérer la dérivée $k$ième de la fonction, à savoir Uf dans notre cas. Puisque la valeur $k = 0$ ne peut pas être utilisée, nous opterons pour la valeur $k = 1$. Ainsi, la norme infinie de l'erreur commise peut être bornée par

$$\|Uf - PUf\|_{L^\infty(a,b)} \leq \frac{(n+2)}{2(n+1)} \frac{\pi(b-a)}{2} \left\| \frac{d}{dx} (U \circ f) \right\|_{L^\infty(a,b)} . \tag{3.1}$$

Or, la norme de la dérivée de la compositon de l'opérateur de Koopman noté U avec la fonction f peut se réécrire. En effet, nous avons

$$\begin{aligned} \frac{d}{dx} (U \circ f) &\underset{(1.5)}{=} \frac{d}{dx} (f \circ \varphi^t) , \\ &= \left(\frac{df}{dx} \circ \varphi^t \right) \left(\frac{d\varphi^t}{dx} \right) , \\ &\underset{(1.5)}{=} \left(U \circ \frac{df}{dx} \right) \frac{d\varphi^t}{dx}, \end{aligned}$$

où φ^t désigne le flot issue de la trajectoire. On peut alors exprimer la norme de cette expression comme étant le produit de la norme de la dérivée de la fonction f et de la dérivée du flot. Autrement dit,

$$\left\| \frac{d}{dx} (U \circ f) \right\| = \left\| \frac{df}{dx} \right\| \left\| \frac{d\varphi^t}{dx} \right\| .$$

Ainsi, l'inégalité (3.1) peut être adaptée, et nous obtenons ainsi

$$\|Uf - PUf\|_{L^\infty(a,b)} \leq \frac{(n+2)}{2(n+1)} \frac{\pi(b-a)}{2} \left\| \frac{df}{dx} \right\| \left\| \frac{d\varphi^t}{dx} \right\|_{L^\infty(a,b)} . \tag{3.2}$$

Cette inégalité permet également d'étudier la propagation de l'erreur. Cela est très utile car lorsque nous approximons une trajectoire non-linéaire, nous appliquons en réalité plusieurs fois à la suite l'approximation finie de l'opérateur de Koopman. Ainsi, il est utile d'étudier l'erreur définie par

$$\|U^n f - (PU)^n f\|_{L_\infty(a,b)} . \tag{3.3}$$

L'objectif est alors de trouver une formule de récurrence pour pouvoir utiliser l'inégalité (3.2). Ainsi, nous pouvons écrire

$$\begin{aligned}\left\|U^n f - (PU)^n f\right\| &= \left\|U^n f - (PU)^n f - U(PU)^{n-1} f + U(PU)^{n-1} f\right\|, \\ &\leq \left\|U^n f - U(PU)^{n-1} f\right\| + \underbrace{\left\|U(PU)^{n-1} f - (PU)(PU)^{n-1} f\right\|}_{(\star)},\end{aligned}$$

Ainsi, nous pouvons appliquer la même décomposition au premier terme du membre de gauche de l'inégalité. Le deuxième terme $(\star)$ quant à lui peut être considéré comme étant une constante. Ainsi, nous obtiendrons une somme de constantes additionnées à la norme (3.2). Par ailleurs, chacune de ces constantes peuvent être réécrites à partir du Théorème 3.1. En effet, si nous dénotons $U(PU)^{n-1} f$ comme étant une fonction g, le terme $(\star)$ n'est rien d'autre que l'erreur entre g et sa projection orthgonale Pg. Ainsi, nous obtenons

$$\|g - Pg\| \leq \frac{(2+n)}{2(n+1)} \frac{\pi(b-a)}{2} \|g'\|$$

Remarque : lorsque nous utilisons l'opérateur de Koopman, nous ne pouvons pas calculer une norme L_2 continue mais uniquement une norme discrète. En effet, celles-ci sont définies respectivement par

$$\| f \|_{L_2(a,b)} \underset{\text{déf}}{=} \sqrt{\int_a^b f^2(x)dx}, \tag{3.4}$$

et

$$\| f \|_{L_2^K(a,b)} \underset{\text{déf}}{=} \sqrt{\sum_{k=1}^{K} f^2(x_k)}, \tag{3.5}$$

où les x_k sont les uniques points auxquels nous connaissons la valeur de la fonction. Une inégalité peut facilement être énoncée.

3.2 Conséquences du choix des paramètres

Jusqu'à présent, l'erreur due aux prédictions linéaires que nous avons présentées dans le deuxième chapitre, n'a été que très peu étudiée. Indépendemment d'une borne théorique comme nous l'avons construit dans la première section, nous pouvons étudier les variations de l'erreur en fonction des différentes valeurs que prennent les paramètres. Dans cette section, nous allons uniquement nous concentrer sur la méthode de prédiction de Korda et al. [9] puisque nous avons déjà pu remarquer au chapitre précédent que celle-ci est meilleure que la seconde méthode [13]. Ainsi, la méthode de Korda et al. va être étudiée selon différents paramètres, à savoir la valeur du pas d'intégration pour générer les données tout en gardant le même nombre total de données, le pas d'intégration sur un temps d'intégration fixé (modifiant donc le nombre de données), le nombre de fonctions de lift c'est-à-dire la dimension de l'espace de lift, ainsi que leur nature et le nombre de trajectoires considérées. Nous pouvons étudier les résultats de deux manières différentes. Nous évaluerons dans un premier temps quelles valeurs de paramètres permettent d'obtenir une bonne approximation globale, c'est-à-dire sur un intervalle de temps relativement long, et dans un second temps, nous considérerons les valeurs de paramètres qui permettent d'obtenir une excellente approximation sur un laps de temps relativement court. En effet, selon l'objectif poursuivi, on pourrait préférer une approximation moyenne mais dont l'erreur n'explose pas et reste relativement constante, ou une très bonne approximation mais limitée dans le temps.

Avant tout, rappellons comment nous procédons. Le système que nous tentons de prédire linéairement est le système dynamique forcé de Van der Pol, c'est-à-dire la dynamique décrite par l'expression (2.2) pour laquelle l'entrée est un signal binaire pseudo-aléatoire. La méthode pour laquelle nous avons opté nécessite de faire des choix. Le plus évident est naturellement le choix des fonctions de lift. Dans le cadre de cette analyse, deux possibilités s'offrent à nous : soit des fonctions polynômiales, soit des fonctions de base radiales comme dans l'Exemple 2.1. Dans ce cas-là, il est nécessaire de définir les centres de ces fonctions, obtenus en pratique pseudo-aléatoirement. En pratique, nous utiliserons les fonctions de base radiale, et nous évoquerons brièvement dans l'une des sections les fonctions polynômiales. Le nombre de fonctions de lift est d'ailleurs un des paramètres que nous allons étudier. En outre, la première étape consiste à générer des données car nous ne disposons pas de données réelles. Puisqu'elles ne doivent pas nécessairement appartenir à la même trajectoire, nous pouvons intégrer à partir de plusieurs conditions initiales. Ainsi, l'un des paramètres que nous allons considérer est le nombre de trajectoires utilisées pour générer l'ensemble des données. Le pas d'intégration est également un des paramètres étudiés. Evidemment, ce pas est celui utilisé d'une part pour générer les données, et d'autre part pour étudier l'erreur commise. Ensuite, nous pouvons également modifier le temps d'intégration, ce qui revient à modifier le nombre de données. Ainsi, les deux cas seront étudiés, à savoir lorsque le pas est modifié tandis que la durée d'intégration ne l'est pas, et lorsque le pas est modifié tout en gardant le nombre de données constant. Une fois que les matrices A, B et C définissant le prédicteur linéaire seront obtenues, nous calculerons une trajectoire par intégration, et son approximation linéaire issue de la condition initiale $(-0.1\,,\,-0.5)$. De plus, l'entrée considérée est une onde carrée de magnitude 1 et de période 0.3s. L'erreur commise due à la prédiction sera alors calculée. Il est important de signaler que, pour chaque valeur de paramètre, le processus que nous venons de décrire est effectué vingt fois. La moyenne et l'écart-type seront finalement évalués.

3.2.1 Pas d'intégration

Commençons tout d'abord par étudier l'erreur lorsque le pas d'intégration est modifié, tout en gardant le même nombre de données. Différentes valeurs peuvent être considérées. Naturellement, nous considérons des valeurs de pas plausibles, à savoir 0.1, 0.05, 0.01, 0.005 et 0.001. Ainsi, les résultats obtenus sont illustrés aux FIGURES [1] 3.1 et 3.2. Ce qu'on y voit est relativement intuitif. En effet, il semblerait que lorsque le pas diminue, l'erreur commise diminue également en moyenne. On voit notamment que, pour la première composante x_1, l'erreur est largement supérieure à un dixième lorsque le pas vaut 0.1 alors qu'elle dépasse rarement cette valeur lorsque le pas est inférieur ou égal à un centième. Toutefois, il est important de noter qu'au plus le pas diminue, au plus l'erreur varie. Cela devrait transparaître dans les valeurs de l'écart-type. Reprenons-les ainsi que les valeurs de la moyenne dans la TABLE 3.1. On y voit que le pas d'intégration le plus grand, à savoir 0.1, admet une moyenne relativement grande mais un écart-type très faible. Un pas de 0.05 permet également d'obtenir un tel écart-type mais il permet d'obtenir une moyenne largement inférieure. Ainsi, une telle valeur de pas pourrait être utilisée si de légères variations de l'erreur sont requises, quelles que soient les données. Par ailleurs, nous pouvons observer que le plus petit pas ne génère pas la plus petite erreur en moyenne. C'est en effet la valeur de pas 0.01 qui minimise la moyenne de l'erreur.

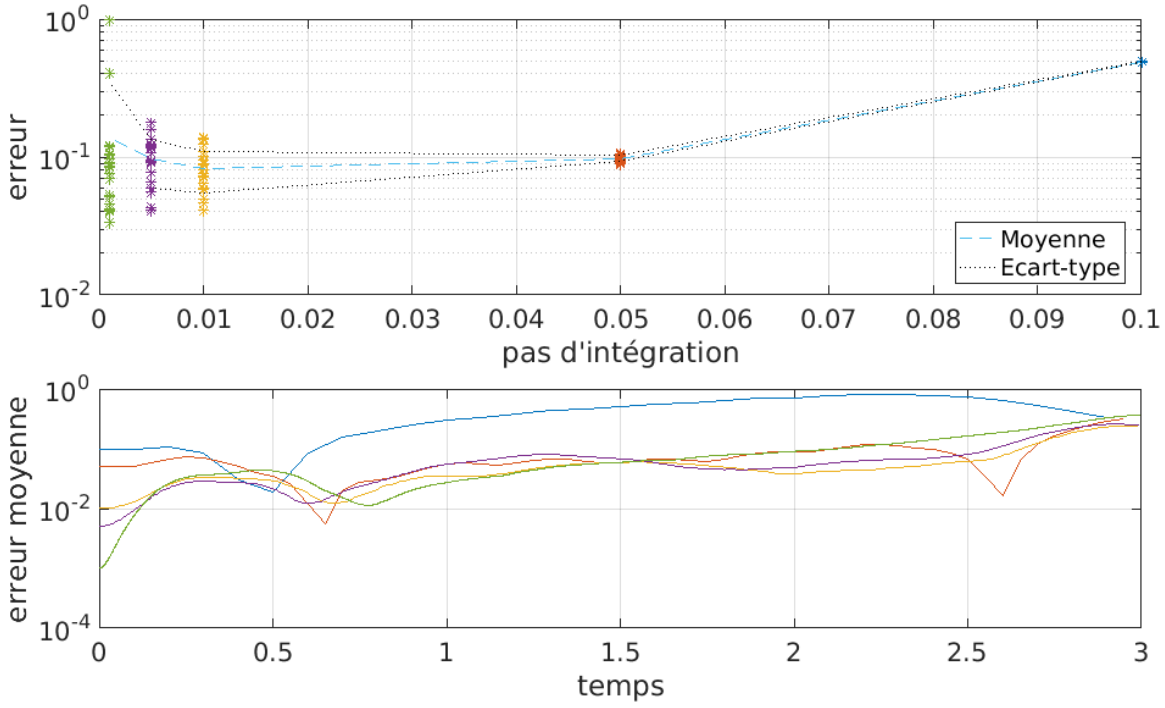


FIGURE 3.1 – Analyse de l'erreur commise lorsque le pas d'intégration varie pour la première composante x_1

1. Les couleurs des courbes dans le deuxième graphe se rapportent aux couleurs des étoiles dans le premier graphe. Ainsi, la courbe bleue par exemple se rapporte au pas 0.1. Cette remarque est valable pour chacune des figures suivantes.

Pas d'intégration	**0.001**	**0.005**	**0.01**	**0.05**	**0.1**
Moyenne	0.1369	0.0965	0.0828	0.0977	0.4890
Ecart-type	0.2141	0.0365	0.0277	0.0047	0.0044

TABLE 3.1 – Moyenne et écart-type de l'erreur commise sur la première composante x_1 du système forcé de Van der Pol (2.2) lorsque le pas d'intégration varie

Notons que les valeurs reprises dans la TABLE 3.1 reflètent l'erreur globale. Ainsi, il est impossible d'y voir qu'au plus la valeur du pas diminue, au plus l'approximation linéaire est bonne dans les premières secondes. En effet, nous pouvons voir à la FIGURE 3.1 que l'erreur moyenne est minimale pour les cinq premiers dixièmes de secondes lorsque le pas est minimal. L'erreur augmente toutefois avec le temps, et finit par atteindre une valeur similaire aux autres pas. D'ailleurs, nous pouvons observer que quelle que soit la valeur du pas, l'erreur moyenne a tendance à augmenter avec le temps sauf pour la valeur 0.01 qui diminue légèrement. En conclusion, le choix de la valeur du pas dépend de l'objectif poursuivi. Une petite valeur de pas serait préférable pour une prédiction très limitée dans le temps, tandis qu'une valeur moyenne serait préférée pour une erreur relativement constante et peu variable selon les données initiales. Finalement, un pas de l'ordre de 0.05 sera préféré si la variation doit être minimisée, tout en obtenant une erreur correcte.

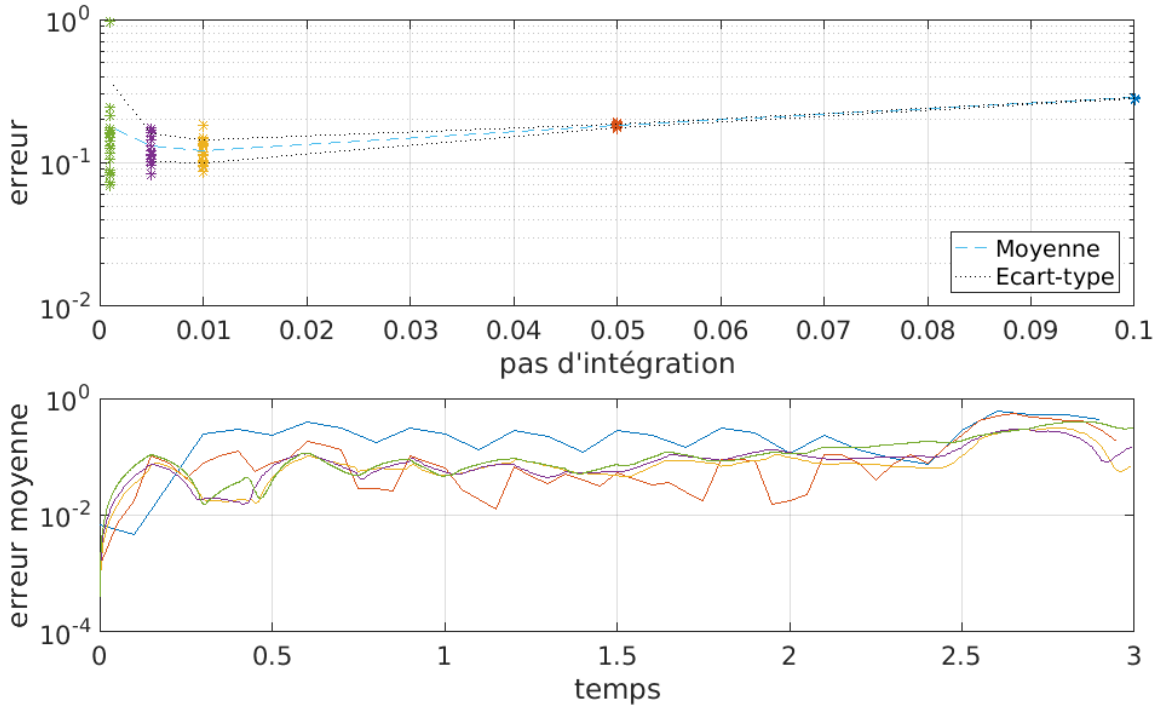


FIGURE 3.2 – Analyse de l'erreur commise lorsque le pas d'intégration varie pour la deuxième composante x_2

Pas d'intégration	**0.001**	**0.005**	**0.01**	**0.05**	**0.1**
Moyenne	0.1803	0.1303	0.1226	0.1819	0.2831
Ecart-type	0.1946	0.0277	0.0234	0.0049	0.0023

TABLE 3.2 – Moyenne et écart-type de l'erreur commise sur la deuxième composante x_2 du système forcé de Van der Pol (2.2) lorsque le pas d'intégration varie

La même analyse peut être faite pour la deuxième composante. Notons tout de même que le système de Van der Pol est en réalité une équation différentielle d'une dimension du second ordre. Ainsi, la trajectoire est décrite par la première composante tandis que la deuxième composante décrit la vitesse. Au vu des valeurs reprises dans laTABLE 3.2, il semble que la tendance soit similaire à celle décrite pour la première composante. L'erreur moyenne en fonction du temps reprise à la FIGURE 3.2 est cependant légèrement différente. Bien que le plus grand pas génère toujours la plus mauvaise approximation, l'erreur moyenne obtenue pour le pas 0.05 est à plusieurs reprises la plus petite. Ainsi, il semblerait que ce pas soit un excellent choix puisqu'il permet d'obtenir peu de variance, une erreur moyenne relativement proche (à six centièmes près) de l'erreur moyenne minimale et une des meilleures approximations sur le long terme. Avant de modifier un autre paramètre, nous pouvons noter que nous avons considéré ici 100 fonctions de lift de base radiales (en plus de l'identité), ainsi la dimension de l'espace lifté vaut 102, et que pour chacune des 100 trajectoires, nous générons 1000 données. Nous gardons ces valeurs de paramètre pour le deuxième changement de paramètre puisque nous allons encore modifier le pas d'intégration mais en intégrant à chaque fois jusqu'au temps 10. Ainsi, on plus le pas est petit, au plus le nombre de données augmente.

3.2.2 Nombre de données

En plus du pas d'intégration, nous pouvons faire varier le nombre de données. Autrement dit, la durée d'intégration reste constante à 10. Etudions dans un premier temps l'erreur sur la première composante x_1. La TABLE 3.3 reprend la moyenne et l'écart-type de l'erreur pour chaque pas d'intégration. Ces valeurs sont illustrées à la FIGURE 3.3 où on retrouve dans un premier temps l'evolution de l'erreur moyenne en fonction du pas et l'erreur moyenne en chaque point en fonction du temps.

Pas d'intégration	**0.001**	**0.005**	**0.01**	**0.05**	**0.1**
Moyenne	0.1145	0.1013	0.0812	0.1014	0.4876
Ecart-type	0.0699	0.0292	0.0254	0.0100	0.0142

TABLE 3.3 – Moyenne et écart-type de l'erreur commise sur la première composante x_1 du système forcé de Van der Pol (2.2) lorsque le pas d'intégration varie mais pas la durée

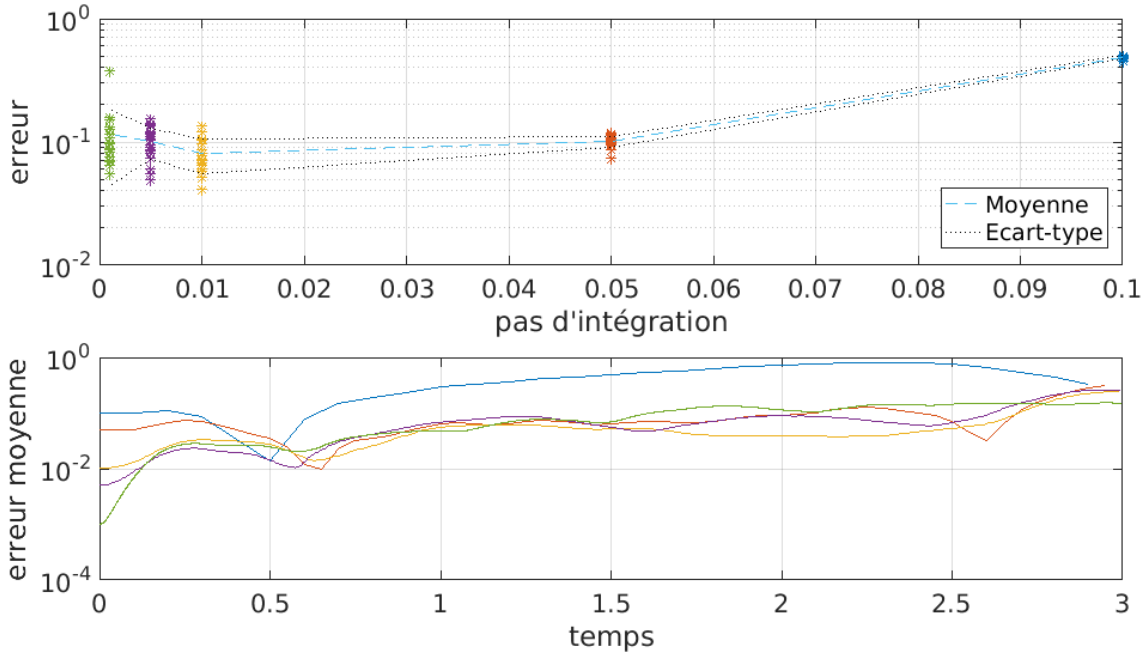


FIGURE 3.3 – Analyse de l'erreur commise lorsque le pas d'intégration varie tandis que la durée reste constante, pour la première composante x_1

La FIGURE 3.3 suggère encore une fois que, lorsque le pas diminue, l'erreur diminue contrairement à la variance qui augmente. En effet, cela se reflète également dans les nombres à la TABLE 3.3. Quelques nuances doivent toutefois être soulignées. Les deux valeurs de pas les plus grandes admettent un écart-type relativement faible, contrairement à la plus petite valeur, c'est-à-dire 0.001 qui admet un écart-type six fois plus grand. La valeur de pas 0.05 est cependant largement préférée à la valeur 0.1 puisque l'erreur moyenne est presque divisée par cinq. Notons encore une fois que ce sont les valeurs de pas les plus petites qui génèrent les meilleurs approximations pour un court laps de temps tandis que le pas le plus grand, c'est-à-dire 0.01 admet globalement une erreur bien plus grande que les autres pas. En effet, l'erreur ne parvient pas à descendre sous le centième et peine même à rester en dessous du dixième. Ainsi, une conclusion similaire au cas précédent peut être déduite.

Pas d'intégration	**0.001**	**0.005**	**0.01**	**0.05**	**0.1**
Nombre de données	10000	2000	1000	200	100
Moyenne	0.2138	0.1511	0.1374	0.1853	0.2831
Ecart-type	0.1431	0.0530	0.0355	0.0085	0.0075

TABLE 3.4 – Moyenne et écart-type de l'erreur commise sur la deuxième composante x_2 du système forcé de Van der Pol (2.2) lorsque le pas d'intégration varie mais pas la durée

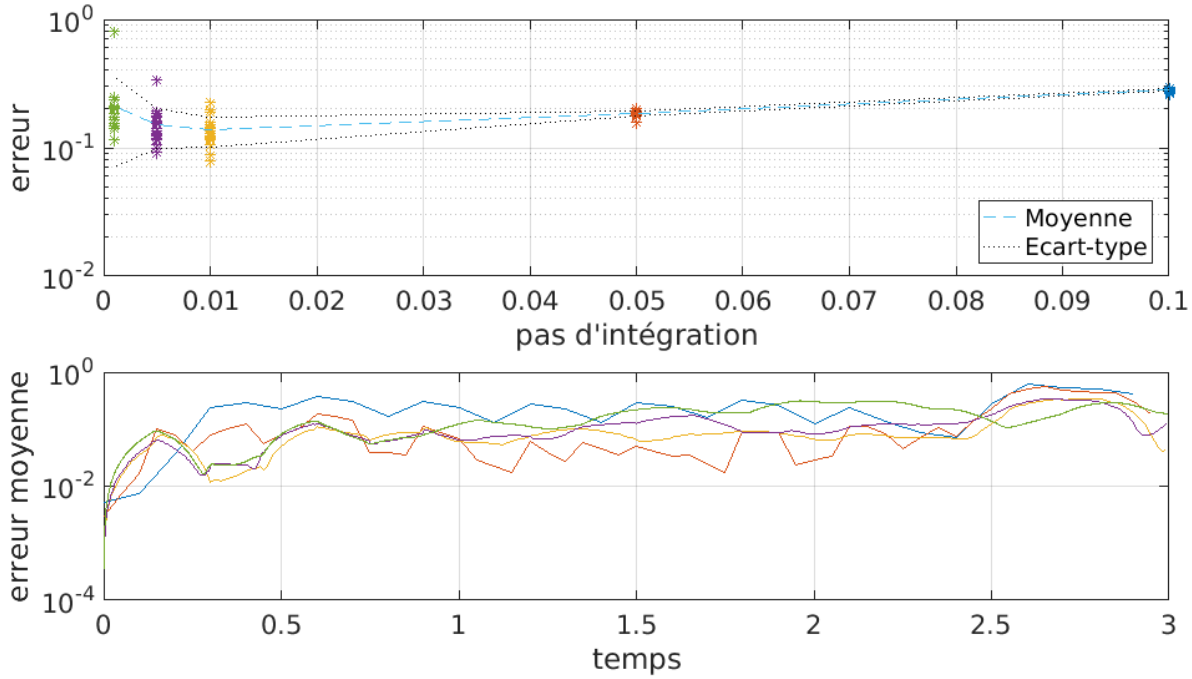


FIGURE 3.4 – Analyse de l'erreur commise lorsque le pas d'intégration varie tandis que la durée reste constante, pour la deuxième composante x_2

Quant à la deuxième composante, c'est-à-dire la vitesse, les différentes semblent moins marquées. En effet, d'après les valeurs reprises à la TABLE 3.4, la moyenne de l'erreur est jamais inférieure à un dixième et jamais supérieure à trois dixième. L'écart-type cependant varie beaucoup plus puisqu'il ne fait qu'augmenter au fur et à mesure que le pas diminie. Cela semble relativement intuitif puisqu'au plus il y a de données, au plus l'approximation est précise. Ainsi, une modification des données engendre plus de modifications dans la construction du prédicteur linéaire. Notons également que la même tendance peut être aperçue à la FIGURE 3.4 qu'à la FIGURE 3.2. En effet, la valeur de pas 0.05 semble générer une erreur plus faible que les autres à plusieurs reprises au cours du temps.

3.2.3 Nombre de trajectoires

Après avoir observé l'impact d'une modification du pas d'intégration en modifiant la durée de l'intégration ou pas, nous pouvons maintenant nous concentrer sur les conséquences dues à la modification du nombre de trajectoires. Rappelons que les données sont générées à partir d'un ensemble de trajectoires issues de conditions initiales définies pseudo-aléatoirement. Ainsi, le nombre de trajectoires impacte nécessairement le nombre total de données. Notons également que pour étudier l'influence du nombre de trajectoires, nous avons fixé la durée de l'intégration à 10, le pas d'intégration à 0.01 et nous avons opté pour 100 fonctions de lift de base radiales en plus de la fonction identité. Comme nous l'avons fait pour les deux sections précédentes, les résultats pour la première composante sont repris à la FIGURE 3.5 et à la TABLE 3.5 tandis que les résultats pour la deuxième composante sont repris à la FIGURE 3.6 et à la TABLE 3.6. Commençons dans un premier temps par analyser l'erreur commise sur la première composante.

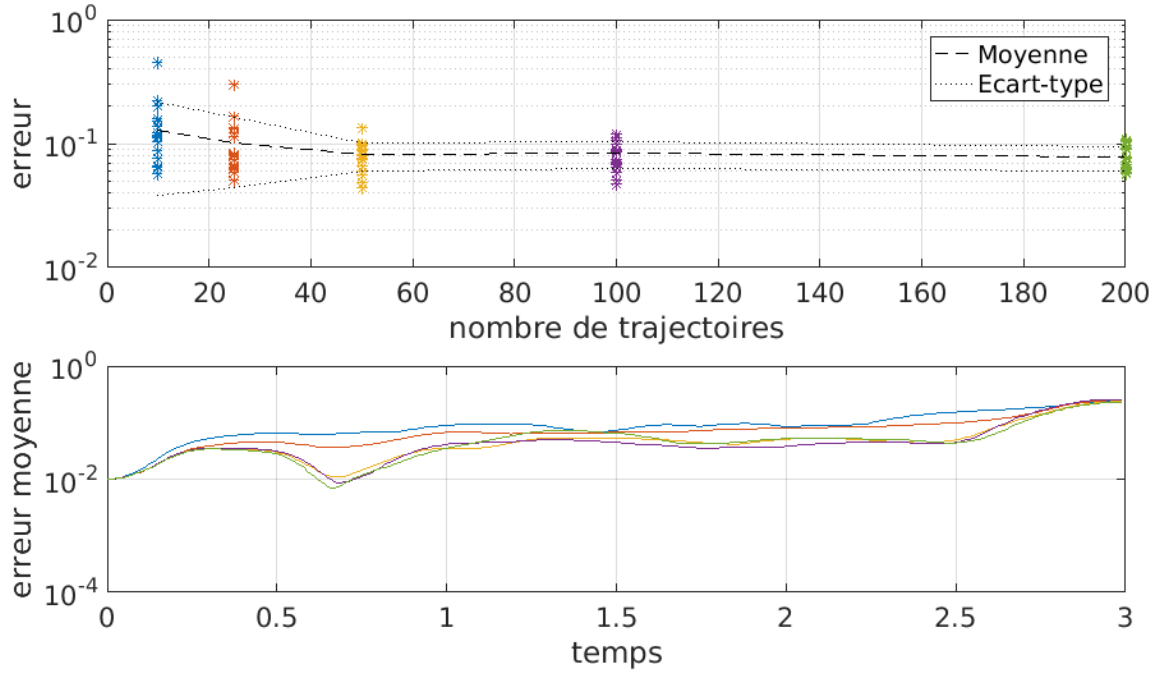


FIGURE 3.5 – Analyse de l'erreur commise lorsque le nombre de trajectoires varie, pour la première composante x_1

Si nous observons la FIGURE 3.5, il est facile de déduire qu'à première vue, au plus nous considérons de trajectoires différentes, au plus l'approximation est précise (en moyenne). Ceci semble intuitif puisque la méthode tente de suivre au mieux le comportement dicté par les données. Ainsi, au plus les données reflètent la dynamique à partir de conditions initiales différentes, au plus la méthode sera plus encline à reproduire une trajectoire. Nous pouvons également remarquer que l'écart-type semble diminuer lorsque le nombre de trajectoires augmentent. Cette tendance est d'ailleurs visible à la TABLE 3.5. Par ailleurs, si nous observons l'évolution de l'erreur moyenne en fonction du temps, il semble que l'erreur est relativement semblable lorsque nous diposons de plus de cinquante trajectoires. De plus, l'erreur augmente avec le temps, comme pour les précédents exemples. Cela n'est pas étonnant puisqu'au plus le temps augmente, au plus l'erreur est susceptible de s'accumuler, voire de diverger. En conclusion, opter pour un nombre de trajectoires entre cinquante et deux cents permet d'obtenir une erreur moyenne et un écart-type de l'ordre de 10^{-2}. Ainsi nous avons opté pour cent trajectoires dans les autres exemples.

Nbr. de trajectoires	**10**	**25**	**50**	**100**	**200**
Moyenne	0.1275	0.1023	0.0808	0.0829	0.0774
Ecart-type	0.0892	0.0587	0.0208	0.0201	0.0173

TABLE 3.5 – Moyenne et écart-type de l'erreur commise sur la première composante x_1 du système forcé de Van der Pol (2.2) lorsque le nombre de trajectoires varie

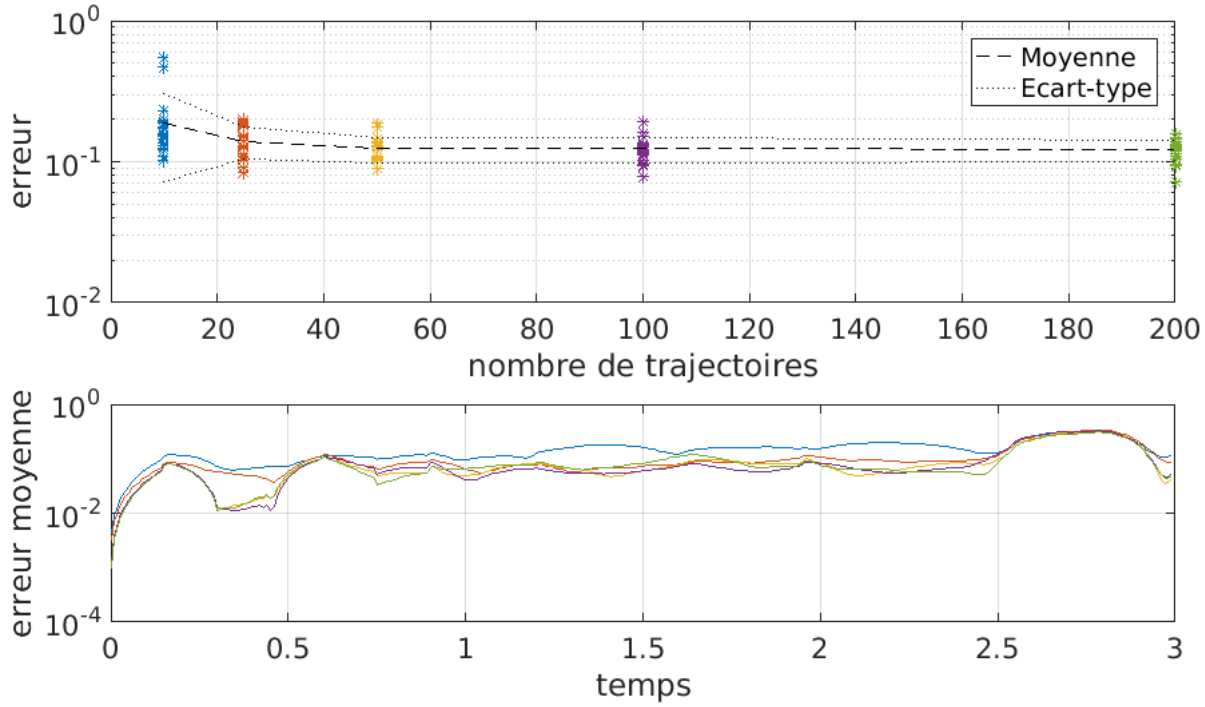


FIGURE 3.6 – Analyse de l'erreur commise lorsque le nombre de trajectoires varie, pour la deuxième composante x_2

L'analyse de l'erreur moyenne commise sur la seconde composante, c'est-à-dire x_2, nous conforte dans notre choix. En effet, que ce soit à la FIGURE 3.5 ou à la TABLE 3.6, la même tendance est visible : la moyenne de l'erreur et l'écart-type diminue au fur et à mesure que le nombre de trajectoires augmente. Toutefois, nous pouvons remarquer que lorsque nous avons déjà beaucoup de fonctions, en rajouter davantage n'augmente pas autant la précision. En pratique, nous pouvons voir qu'avec cent fonctions, l'erreur moyenne vaut 0.1231 tandis qu'elle vaut 0.1213 à savoir environ deux millièmes de moins avec cent fonctions supplémentaires. Il ne semble donc pas nécessaire de considérer plus de cent trajectoires.

Nbr. de trajectoires	**10**	**25**	**50**	**100**	**200**
Moyenne	0.1881	0.1402	0.1241	0.1231	0.1213
Ecart-type	0.1157	0.0362	0.0254	0.0254	0.0203

TABLE 3.6 – Moyenne et écart-type de l'erreur commise sur la deuxième composante x_2 du système forcé de Van der Pol (2.2) lorsque le nombre de trajectoires varie

3.2.4 Nombre de fonctions de lift

Grâce aux analyse précédentes, nous savons dorénavant qu'une centaine de trajectoires est suffisante pour obtenir une erreur de l'ordre de 10^{-2}. De plus, le pas d'intégration 0.01 permet d'obtenir à la fois une bonne approximation et une variance relativement faible. Ainsi, dans l'analyse de l'impact du nombre de fonction de lift, nous fixerons à 100 le nombre de trajectoires, à 0.01 le pas d'intégration et à 10 sa durée, et finalement nous opterons pour les fonctions de base radiales. Dans ces conditions, nous pourrons ainsi analyser l'erreur en fonction du nombre de fonctions de lift. Intuitivement, au plus il y en a, au plus l'approximation est bonne. Vérifions donc cette supposition par l'exemple de Van der Pol, et attardons-nous dans un premier temps sur la première composante, c'est-à-dire x_1.

La FIGURE 3.7 semble indiquer la tendance suivante : au plus nous considérons de fonctions de lift, au plus l'erreur diminue, mais au-delà de cent fonctions, l'erreur a tendance à stagner. C'est exactement ce que les chiffres reflètent à la TABLE 3.7. En effet, lorsque nous considérons au minimum une cinquantaine de fonctions, l'erreur moyenne reste environs à 0.09, avec des variations de l'ordre du centième. Il n'y a donc pas d'intérêt a priori à considérer 102, 202 ou encore 302 fonctions. Il est toutefois important de souligner que l'écart-type diminue également et admet sa valeur minimale avec 102 fonctions. Ces observations suggèrent donc de considérer une centaine de fonctions de lift. Par ailleurs, un grand nombre de fonctions de lift permet d'obtenir une meilleure approximation en fonction du temps. En effet, si nous observons l'évolution de l'erreur moyenne en fonction du temps à la FIGURE 3.7, nous y voyons que l'erreur reste relativement constante (sauf vers la fin) lorsque nous considérons 102, 202 ou 302 fonctions de lift. En conclusion, le choix d'une centaine de fonctions de lift semble être le choix optimal

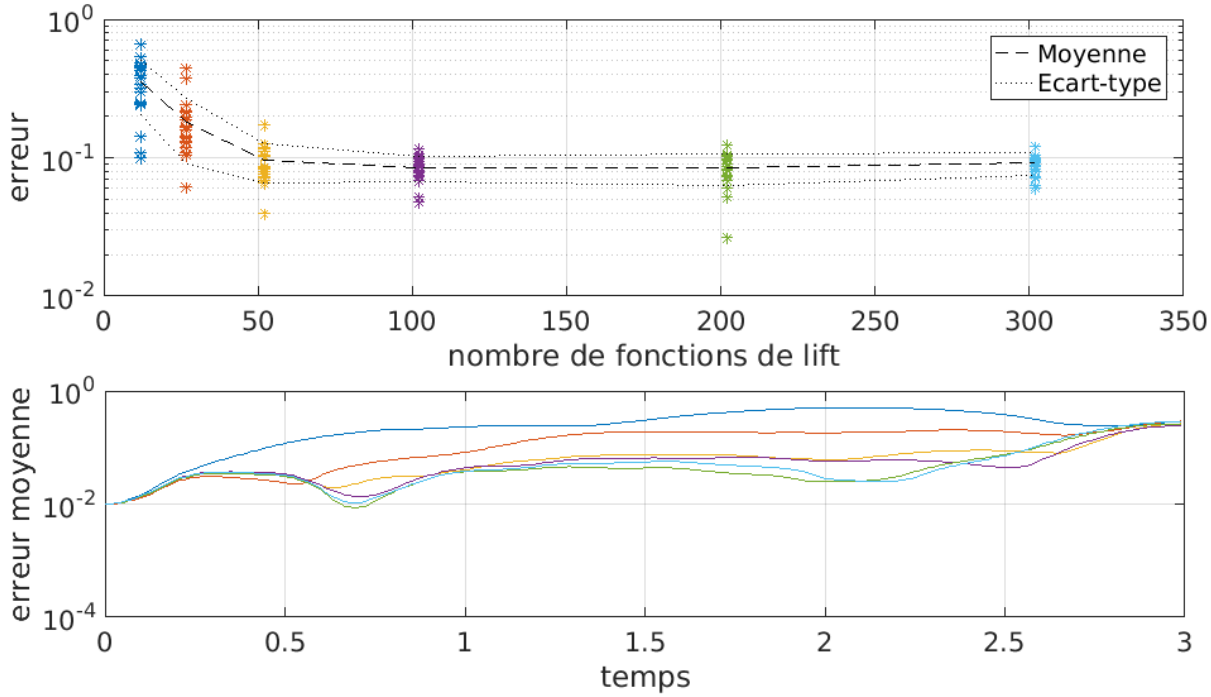


FIGURE 3.7 – Analyse de l'erreur commise lorsque le nombre de fonctions de lift, pour la première composante x_1

Nbr. de fonctions	**12**	**27**	**52**	**102**	**202**	**302**
Moyenne	0.3532	0.1801	0.0953	0.0842	0.0848	0.0919
Ecart-type	0.1467	0.0892	0.0295	0.0169	0.0216	0.0173

TABLE 3.7 – Moyenne et écart-type de l'erreur commise sur la première composante x_1 du système forcé de Van der Pol (2.2) lorsque le nombre de fonctions de lift varie

puisqu'il minimise l'écart-type moyen, il permet d'obtenir une erreur moyenne de l'ordre de 10^{-2} et l'erreur moyenne en fonction du temps ne semble pas exploser.

Terminons cette analyse par la deuxième composante du système dynamique. Il est facile de s'apercevoir que la FIGURE 3.8 et la TABLE 3.8 reflètent les mêmes tendances que celles de la première composante. En effet, l'erreur moyenne et l'écart-type diminue lorsque le nombre de fonctions augmente (sauf pour la moyenne avec 302 fonctions). Notons toutefois que le passage de 102 à 202 fonctions est plus utile pour la deuxième composante puisque l'erreur moyenne est diminuée de deux centièmes, et l'évolution de l'erreur moyenne en fonction du temps est meilleure, comme illustré à la Figure 3.8.

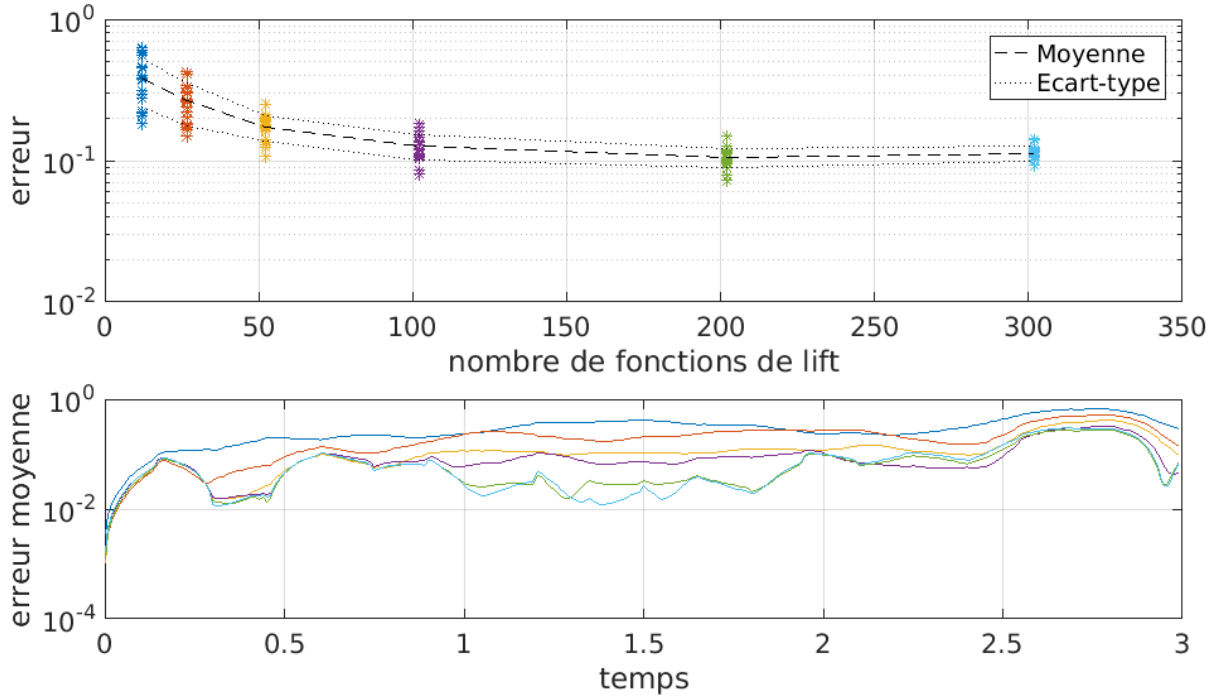


FIGURE 3.8 – Analyse de l'erreur commise lorsque le nombre de fonctions de lift, pour la deuxième composante x_2

Nbr. de fonctions	**12**	**27**	**52**	**102**	**202**	**302**
Moyenne	0.3812	0.2636	0.1718	0.1275	0.1057	0.1127
Ecart-type	0.1405	0.0874	0.0351	0.0263	0.0163	0.0133

TABLE 3.8 – Moyenne et écart-type de l'erreur commise sur la deuxième composante x_2 du système forcé de Van der Pol (2.2) lorsque le nombre de fonctions de lift varie

3.2.5 Choix des fonctions de lift

Abordons brièvement dans cette section le choix des fonctions de lift. Jusqu'ici nous avons uniquement considéré les fonctions de lift suggérées dans la référence [9]. Toutefois, rien ne nous oblige à priori à choisir ces fonctions. C'est la raison pour laquelle nous avons tenté d'utiliser de simples fonctions polynômiales (voir Annexe A.2 pour le code Matlab utilisé). Comme précédemment, nous avons itéré vingt fois la méthode, et le résultat est illustré à la FIGURE 3.9. L'objectif ici n'est pas d'analyser l'erreur commise en fonction d'un paramètre, mais plutôt de se convaincre qu'aucune des deux types de fonction de base n'est mieux que l'autre. En effet, lorsqu'on regarde l'erreur générée, nous n'obtenons pas de meilleurs résultats que ceux obtenus grâce aux fonctions de base radiale. En effet, l'erreur est toujours de l'ordre du dixième tandis que nous avons obtenu précédemment une erreur moyenne de l'ordre du centième à plusieurs reprises. En conclusion, il est largement préférable d'opter pour les fonctions de base radiale. Notons toutefois que d'autres

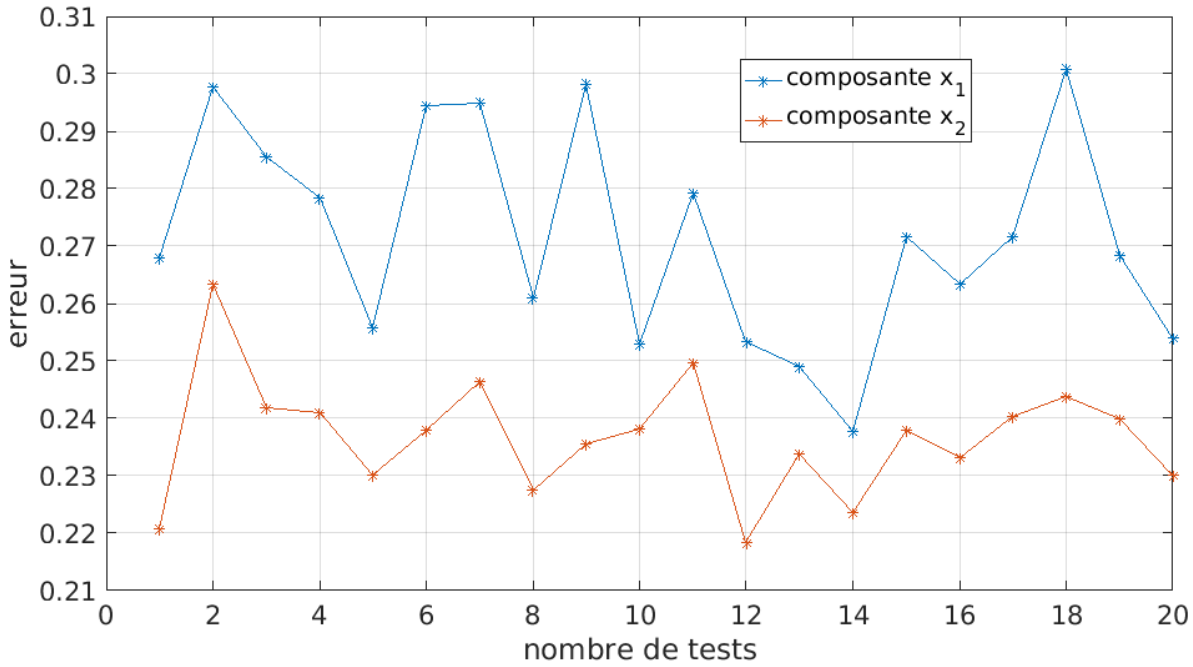


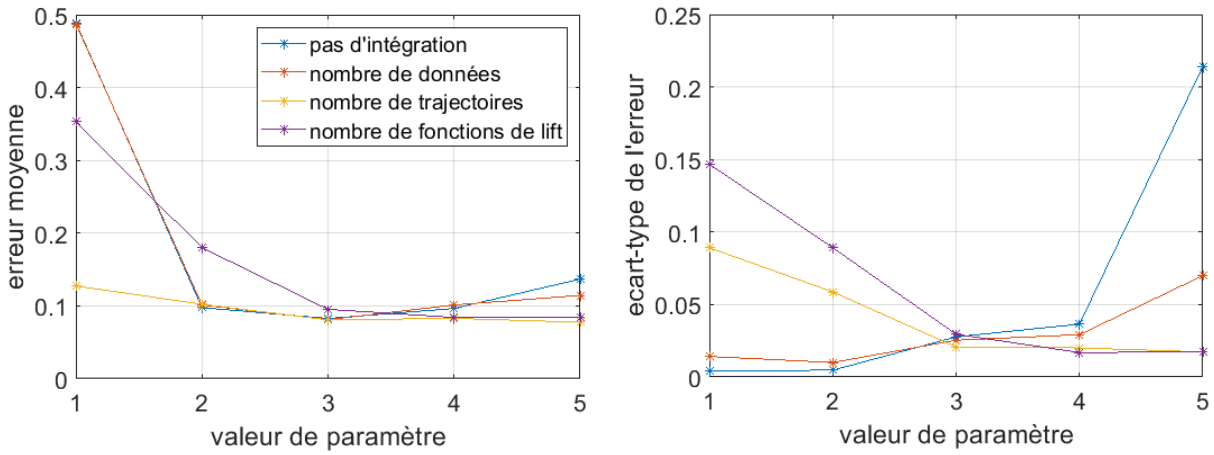
FIGURE 3.9 – Erreur moyenne commise sur les deux composantes du systme forcé de Van Pol lorsque les fonctions de lift sont des polynômes

fonctions de base auraient pu être envisagées, mais il semblerait que les fonctions de base radiable soient déjà un des meilleurs choix possibles. Il est également intéressant de remarquer que l'erreur sur la deuxième composante est inférieure à l'erreur sur la première composante, alors que nous observions généralement le comportement inverse avec les fonctions de base radiale.

3.3 Conclusion

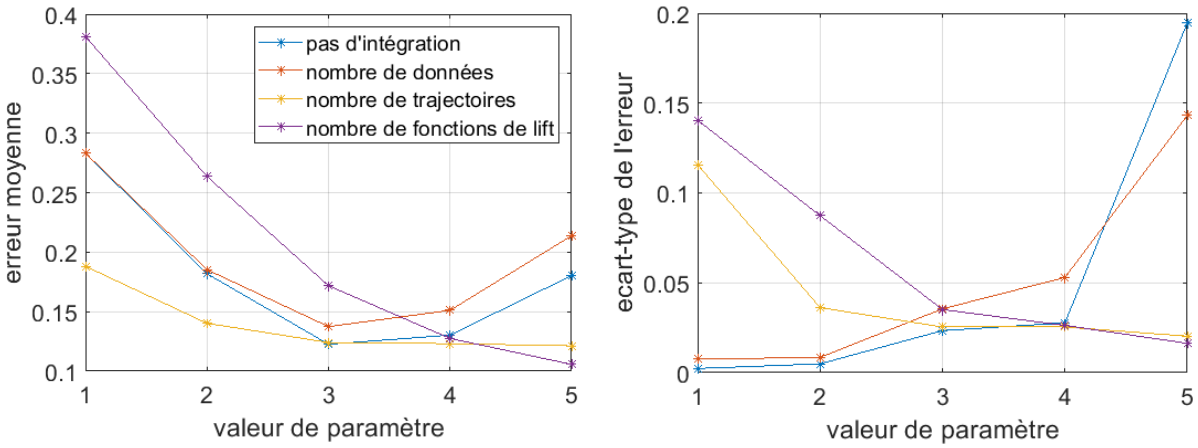
Après avoir étudié individuellement les conséquences dûes à un changement de valeur paramètre, il semble utile de regrouper ces différentes analyses. En effet, nous avons pu comparer l'erreur pour deux valeurs de paramètres, mais nous ne savons pas encore à l'heure actuelle si une modification de permètre est plus utile qu'une autre. Par exemple, est-il préférable de diminuer le pas d'intégration ou d'augmenter le nombre de fonctions de lift ? L'objectif de cette section est donc de regrouper les analyses précédentes et de conclure à propos de l'influence es paramètres sur l'erreur moyenne engendrée par l'approximation linéaire proposée par Korda et al. dans la référence [9]. Afin de répondre à la question soulevée précédemment, nous allons principalement nous baser sur les analyses précédentes ainsi que sur deux figures, à savoir les FIGURES 3.10 et 3.11. Celles-ci reprennent l'évolution de l'erreur moyenne et de l'écart-type en fonction des valeurs de paramètres. Ainsi, la première valeur de pas d'intégration est 0.1, la deuxième est 0.05 et ainsi de suite. Quant aux nombre de fonctions de lift ou au nombre de trajectoires, les valeurs sont classées de manière croissante. Notons que nous ignorons les résultats obtenus lorsque nous considérons 302 fonctions de lifts afin d'obtenir cinq valeurs différentes pour chaque paramètre. De plus, n'oublions pas de signaler que le choix des paramètres dépend de l'objectif poursuivi.

Quelle que soit la composante, il semble que l'un des paramètres dont on doit absolument tenir compte est le nombre de fonctions de lift. En effet, lorsque nous en considérons peu, la moyenne de l'erreur ainsi que l'écart-type explosent. A contrario, le nombre de tra-



(a) la moyenne (b) l'écart-type

FIGURE 3.10 – Comparaison de l'influence d'un changement de paramètre sur l'erreur commise sur première composante via deux mesures statistiques



(a) la moyenne (b) l'écart-type

FIGURE 3.11 – Comparaison de l'influence d'un changement de paramètre sur l'erreur commise sur la deuxième composante via deux mesures statistiques

jectoires semble être bien moins crucial puisque l'erreur diminue uniquement de quelques centièmes, et reste même raltivement constante à partir d'une cinquantaine de trajectoires. Le pas d'intégration joue également un rôle crucial : un pas de 0.1 ou de 0.001 est définitivement à banir. En effet, le pas de 0.1 fait exploser l'erreur sur les deux composantes, tandis que le pas de 0.001 augmente considérablement l'écart-type.

Chapitre 4

p-dominance et l'opérateur de Koopman

Depuis le début, nous travaillons sur des systèmes dynamiques et nous tentons, via l'opérateur de Koopman, de nous affranchir de la difficulté du caractère non-linéaire. En effet, dès que la dimension d'un système non-linéaire est supérieure à deux, il est généralement difficile de l'analyser, et notamment de le prédire, comme nous avons pu le voir dans les chapitres précédents. L'opérateur de Koopman nous a permis de nous ramener à un système linéaire, et ainsi d'utiliser les méthodes initialement adaptées à ces systèmes. En particulier, une structure linéaire permet d'introduire les notions de fonction et valeur propres, et c'est donc cette approche, dite approche spectrale qui nous intéresse dans ce chapitre. Bien que l'opérateur de Koopman permette d'obtenir un système linéaire, il est possible d'extraire localement de tout système non-linéaire un système linéaire via la linéarisation et donc la matrice Jacobienne. Alors que la première approche est globale, la linéarisation est plutôt locale. C'est sur ce principe que se base l'approche différentielle, une méthode consistant à transposer localement des concepts initialement définis pour des systèmes linéaires, à des sytèmes non-linéaires via la linéarisation du système. Cette approche a permis par exemple de définir les concepts de positivité différentielle et de p-dominance qui font l'objet de ce chapitre. En effet, ces deux notions traitent du comportement asymptotique d'un système non-linéaire et peuvent être étroitement liés à l'opérateur de Koopman. Le lien a été déjà été établi pour la positivité différentielle [10] et nous tenterons de faire de même dans le cas de la p-dominance. Ce chapitre se base également sur les articles [4], [5] de Fulvio Forni et Rodolphe Sepulchre dans lesquels les concepts de p-dominance et de positivité différentielle sont introduits.

Dans la première section de ce chapitre, nous allons présenter la théorie de la p-dominance. Un cas particulier de la p-dominance est la positivité différentielle qui est équivalence à la 1-dominance sous certaines conditions. Ainsi cette théorie sera développée dans la deuxième section. Dans ces deux premières sections, l'approche sera identique : la théorie sera développée dans un premier temps pour un système linéaire, et cette theorie sera élargie dans un deuxième temps aux systèmes non-linéaires en adaptant la définition initiale au système prolongé, c'est-à-dire à la matrice Jacobienne. Finalement, la troisième et dernière section de ce chapitre tente de faire le lien entre la théorie de la p-dominance et l'opérateur de Koopman. Nous nous baserons sur le lien déjà établi entre positivité différentielle et l'opérateur dans la référence [10].

4.1 Théorie de la p-dominance

Généralement, les systèmes de grande dimension, qu'ils soient linéaires ou pas, sont difficiles à analyser bien qu'ils soient relativement courants. L'objectif est donc de réduire la dimension en se concentrant sur les directions principales du système. Cette première section présente un nouvel outil, à savoir la théorie de la p-dominance, permettant de connaitre la dimension du comportement dominant, toujours inférieure (ou égale) à la dimension initiale du système. Ce nouvel outil va être introduit selon la démarche décrite précédemment, c'est-à-dire l'approche différentielle. Autrement dit, la définition de ce nouvel outil va être définie dans un premier temps pour les systèmes linéaires, et elle sera par la suite définie pour les systèmes non-linéaires via la matrice Jacobienne. Commencons donc tout d'abord par les systèmes linéaires, et présentons les résultats les plus intéressants liés à ce concept.

Comme annoncé précédemment, la p-dominance permet de se concentrer sur les directions dominantes du système. Or, dans le cadre des systèmes linéaires, ces directions sont décrites par les vecteurs propres associés aux plus grandes valeurs propres de la matrice qui définit le système. Il est donc logique d'utiliser l'approche spectrale pour définir la notion de système p-dominant.

Définition 4.1

Un système linéaire $\dot{x} = Ax$ avec $A \in \mathbb{R}^{n\times n}$ est **p-dominant** avec le taux $\lambda \geq 0$ s'il existe une matrice symétrique P d'inertie $(p, 0, n-p)$ tel que

$$A^tP + PA + 2\lambda P + \varepsilon I \leq 0, \tag{4.1}$$

pour $\varepsilon \geq 0$. La propriété est stricte si $\varepsilon > 0$.

Rappelons dans un premier temps qu'une matrice d'inertie $(p, 0, n-p)$ admet p valeurs propres négatives et $n-p$ valeurs propres positives. Par la suite, une telle matrice sera dite d'inertie p. Par ailleurs, la notion de p-dominance est assez simple à comprendre puisque le taux λ correspond au shift que l'on effectue sur le spectre de la matrice A de sorte à obtenir p valeurs propres strictement positives. En effet, nous savons que pour obtenir les valeurs propres λ d'une matrice, nous résolvons l'équation suivante

$$\det(A - \Lambda I) \;=\; 0.$$

Nous obtenons alors le spectre de la matrice A qui peut être représenté dans le plan complexe où l'abscisse décrit la partie réelle tandis que l'ordonnée décrit la partie complexe de chaque valeur propre. Il est possible d'effectuer un shift sur ce spectre, c'est-à-dire de déplacer chacune des valeurs propres de la même distance autant vers la gauche que vers la droite. Ceci est fait en additionant une matrice diagonale à la matrice initiale, dont les éléments diagonaux sont identiques et valent la longueur du shift. Autrement dit, le shift est généré en aditionnant une matrice de la forme λI à la matrice initiale. En effet si nous dénotons par Λ une valeur propre de la matrice A, alors il existe nécessairement une valeur propre notée μ de la matrice $A + \lambda I$ donnée par $\mu \;=\; \lambda + \Lambda$. En effet,

$$\begin{aligned} \det(A-\Lambda I) &= 0, \\ \Rightarrow \quad \det(A+\lambda I-\mu I) &= 0, \\ \Leftrightarrow \quad \det\left(A-[\mu-\lambda]I\right) &= 0, \\ \Leftrightarrow \quad \det(A-\Lambda I) &= 0, \end{aligned}$$

où $\Lambda = \mu - \lambda$. Ainsi, une valeur λ négative génère un shift vers la gauche tandis qu'une valeur positive génère un shift vers la droite. Notons que dans le cadre de la p-dominance, seuls les shifts vers la droite sont autorisés puisque le paramètre λ doit être positif. La notion de p-dominance consiste donc à trouver un shift λ adéquat vers la droite de sorte à obtenir p valeurs propres positives donc dominantes, et $n-p$ valeurs propres négatives donc négligeables. Ainsi, il est possible qu'un système soit à la fois p_1-dominant et p_2 dominant pour deux shifts différents. En réalité, si un système est p-dominant, alors il est nécessairement p'-dominant pour tout p' supérieur à p. Toutefois, l'important ici est de déterminer le plus petit p possible.

La notion de p-dominance peut être reformulée via la fonction quadratique $V(x) = x^t\,P\,x$ puisque si nous calculons la dérivée de cette fonction le long des trajectoires, nous obtenons

$$\begin{aligned} \dot{V}(x) &= \dot{x}^t\,P\,x + x^t\,P\,\dot{x}, \\ &= x^t A^t P x + x^t P A x, \\ &= x^t\left(A^t P + P A\right)x. \end{aligned}$$

Sous l'hypothèse de p-dominance, on en déduit que

$$\dot{V}(x) \;\leq\; -2\lambda V(x) - \varepsilon\,|x|^2.$$

Ceci implique donc que la fonction quadratique décroit le long des trajectoires, et que si nous définissons les cones K^- et K^+ par

$$K^- \;=\; \{x \in \mathbb{R}^n \mid V(x) \leq 0\} \qquad \text{et} \qquad K^+ \;=\; \{x \in \mathbb{R}^n \mid V(x) \geq 0\}$$

alors il est facile de voir que K^- est invariant vers l'avant tandis que K^+ est invariant vers l'arrière, c'est-à-dire

$$e^{At}K^- \subseteq K^- \quad \text{et} \quad e^{-At}K^+ \subseteq K^+$$

pour tout temps $t \geq 0$.

Quelques caractérisations de la p-dominance ont été établies par F. Forni et R. Sepulchre dans la version précédente de leur article [5] de 2017. Celles-ci vont nous permettre de mieux comprendre cette notion.

Proposition 4.1

Pour $\varepsilon > 0$, l'inéquation linéaire matricielle (4.1) est équivalente à chaque des conditions suivantes :

(1) La matrice $A + \lambda I$ admet p valeurs propres à partie réelle strictement positive et $n - p$ valeurs propres à partie réelle strictement négative,

(2) Il existe un scission invariante de l'espace $\mathbb{R}^n = \mathcal{H} \oplus \mathcal{V}$ telle que $A\mathcal{H} \subset \mathcal{H}$ et $A\mathcal{V} \subset \mathcal{V}$, où $\mathcal{H}$ est de dimension p et $\mathcal{V}$ est de dimension $n-p$. De plus, il existe des constantes $0 \leq \underline{c} \leq 1 \leq \overline{c}$ et $\overline{\lambda} > \lambda > \underline{\lambda}$ telles que

$$\forall x \in \mathcal{H} \ : \ \left|e^{At}x\right| \ \geq \ \underline{c}\, e^{-\underline{\lambda}t}\,|x| \quad t \geq 0,$$
$$\forall x \in \mathcal{V} \ : \ \left|e^{At}x\right| \ \leq \ \overline{c}\, e^{-\overline{\lambda}t}\,|x| \quad t \geq 0.$$

Cela signifie que, lorsqu'un système linéaire $\dot{x} = Ax$ est strictement p-dominant, on observe p modes dominants et $n-p$ modes transitoires. Cette observation nous permet de faire le lien avec des propriétés que nous connaissons déjà. Par exemple, si un système est strictement 0-dominant, cela signifie que la fonction V est une fonction de Lyapunov et par extension, V est toujours une fonction de Lyapounov mais uniquement pour la restriction du flot à l'espace $\mathcal{V}$. Notons également que l'inégalité (4.1) est en réalité équivalente à la contrainte définie par l'inégalité matricielle suivante

$$\begin{bmatrix} \dot{x} \\ x \end{bmatrix}^t \begin{bmatrix} O & P \\ P & 2\lambda P + \varepsilon I \end{bmatrix} \begin{bmatrix} \dot{x} \\ x \end{bmatrix} \ \leq \ 0. \tag{4.2}$$

Remarque : Dans les chapitres précédents, nous avons notamment étendu la définition de l'opérateur de Koopman aux systèmes avec entrée. Il est également possible d'étendre la notion de p-dominance à de tels systèmes, c'est-à-dire des systèmes de la forme

$$\begin{cases} \dot{x} \ = \ Ax \ + \ Bu, \\ y \ = \ Cx \ + \ Du. \end{cases}$$

On parle dans ce cas-là de p-dissipativité. Alors, la contrainte précédente peut être adaptée aux systems ouverts, et devient

$$\begin{bmatrix} \dot{x} \\ x \end{bmatrix}^t \begin{bmatrix} O & P \\ P & 2\lambda P + \varepsilon I \end{bmatrix} \begin{bmatrix} \dot{x} \\ x \end{bmatrix} \ \leq \ \underbrace{\begin{bmatrix} y \\ u \end{bmatrix}^t \begin{bmatrix} Q & L \\ L^t & R \end{bmatrix} \begin{bmatrix} y \\ u \end{bmatrix}}_{:= s(y,u)}, \tag{4.3}$$

où P est une matrice d'inertie p, le taux $\lambda \geq 0$, $\varepsilon \geq 0$ et L, Q et R sont des matrices aux dimensions adéquates et où le membre de droite, c'est-à-dire l'expression $s(y,u)$, est appelé le taux d'approvisionement. Ainsi, un système dynamique linéaire ouvert est dit p-dissipatif si la relation (4.3) est satisfaite pour tous x et u. Cette inégalité est équivalente à l'inégalité suivante

$$\begin{bmatrix} A^tP + PA + 2\lambda P - C^tQC + \varepsilon I & PB - C^tL - C^tQD \\ B^tP - L^tC - D^tQC & -D^tQD - L^tD - D^tL - R \end{bmatrix} \leq 0.$$

Il apparaît donc que p-dominance et p-dissipativité peuvent être traitées de la même manière.

Afin de mieux saisir la notion de p-dominance, illustrons-la maintenant via un exemple simple ainsi que les différents cônes que nous avons définis.

Exemple 4.1. *Considérons le système dynamique linéaire suivant*

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 \\ -2 & -3 \end{pmatrix}}_{:=A} \begin{pmatrix} x \\ y \end{pmatrix}. \tag{4.4}$$

Les valeurs propres de la matrice A sont $\lambda_1 = 0.4142\ (\geq 0)$ et $\lambda_2 = -2.4142\ (\leq 0)$. Puisque la p-dominance vise à dissocier les directions dominantes des directions provisoires au moyen de la division du spectre de la matrice, il semble logique de consider le taux $\lambda = 0$. En effet, les deux valeurs propres sont déjà séparées puisqu'une d'entre elles est positive, tandis que l'autre est négative. Ceci est clairement visible à la FIGURE *4.1. Ainsi, aucun shift n'est nécessaire, d'où $\lambda = 0$ et on s'attend à obtenir un système 1-dominant. En pratique, la matrice P* [1] *définie par*

$$P = \begin{pmatrix} -3.060 & -1.275 \\ -1.275 & -0.255 \end{pmatrix},$$

est d'inertie 1 puisque ses valeurs propres sont -3.5529 et 0.2379. Alors, si nous calculons l'expression de V, nous obtenons

$$\begin{aligned} V(x,y) &= -0.255y^2 - 2.55xy - 3.06x^2, \\ &= -0.255(y + 8.6056x)(y + 1.3944x). \end{aligned}$$

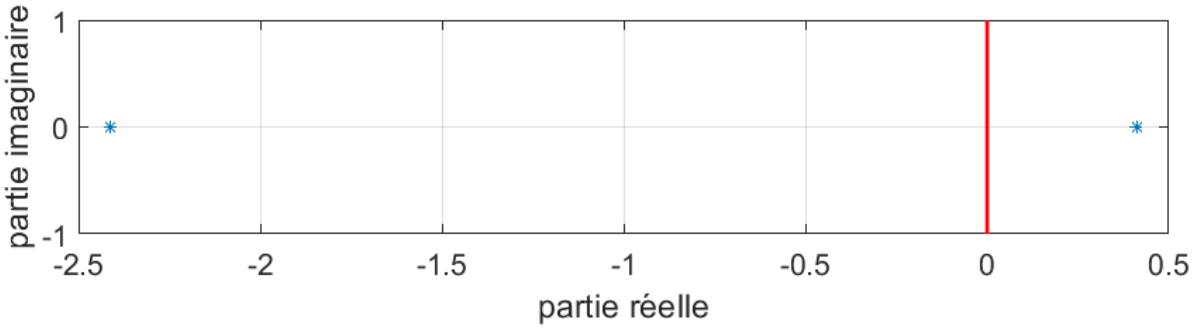


FIGURE 4.1 – Illustration du spectre de la matrice A du système (4.4)

1. Toutes les matrices P de ce travail sont calculées via une méthode algorithmique à base d'inégalités matricielles linéaires, détaillée dans l'article [5] de F. Forni et R. Sepulchre. Nous utilisons YalMip et SeDuMi sur Matlab pour la résolutions des LMIs.

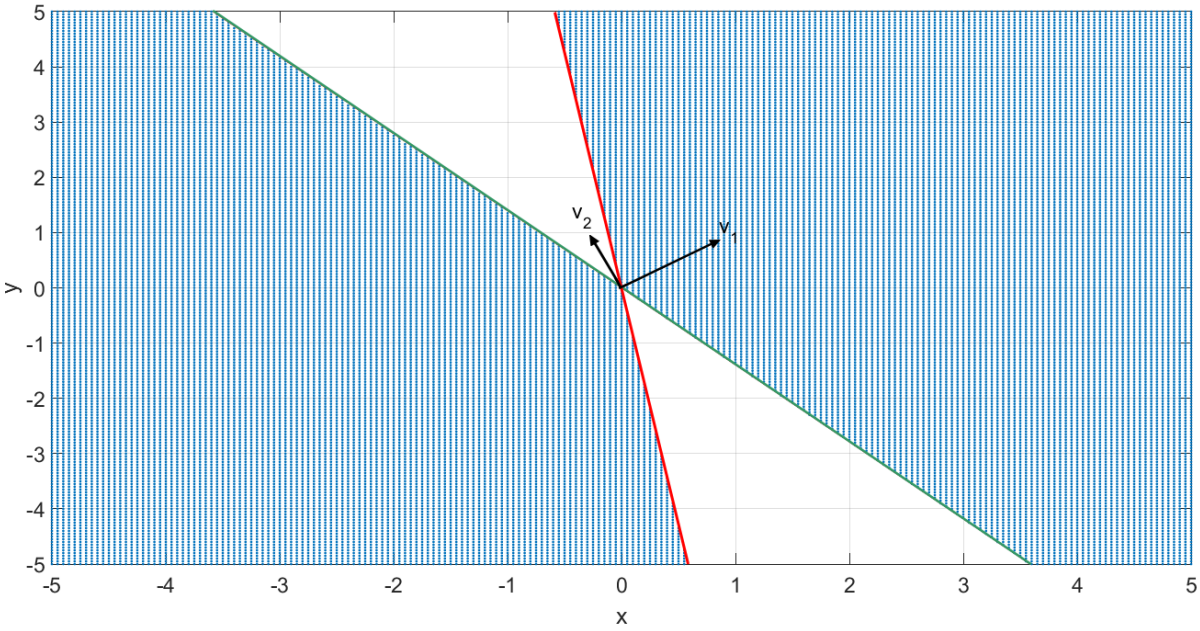


FIGURE 4.2 – Illustration du cône K^- en bleu, des délimitations ($y = -1.3944x$ en rouge et $y = -8.6056x$ en vert) et des vecteurs propres de la matrice du systeme linéaire (4.4)

Ainsi, un vecteur appartient au cône K^- si les expressions $y + 8.6056x$ et $y + 1.3944x$ ont le même signe. Naturellement, le vecteur appartient à K^+ si les deux expressions ont un signe différent. Ceci est visible sur la FIGURE *4.2, où nous avons également affiché les vecteurs propres de la matrice du système linéaire. Nous pouvons y voir que K^- comprend le vecteur propre associé à la plus grande valeur propre, tandis que K^+ contient l'autre vecteur propre, v_2.*

L'observation que nous venons de faire à propos des vecteurs propres n'est pas anodine. En effet, au vu des définitions de la fonction quadratique $V(x)$ et de la p-dominance, nous pouvons démontrer que le cône $\mathcal{K}^-$ contient les p premiers vecteurs propres tandis que le cône $\mathcal{K}^+$ contient les $n - p$ autres vecteurs propres.

Propriété 4.1

Soit $\dot{x} = Ax$ un système dynamique linéaire p-dominant avec le taux $\lambda \geq 0$ et la matrice symétrique P d'inertie p. Supposons que la matrice A admette n valeurs propres λ_i pour $i = 1, \ldots, n$ telles que $Re(\lambda_1) \geq \ldots \geq Re(\lambda_n)$ et n vecteurs propres notés v_i, $i = 1, \ldots, n$. Alors, le cône $\mathcal{K}^-$ contient les p premiers vecteurs propres de A et le cône $\mathcal{K}^+$ contient les $n - p$ vecteurs propres v_i de A pour $i > p$.

Démonstration. Soit v_i, un vecteur propre de A où $1 \leq i \leq p$. Nous devons montrer que v_i appartient au cône $\mathcal{K}^-$, autrement dit que

$$V(v_i) \underset{\text{déf}}{=} {v_i}^t P v_i \leq 0.$$

Or, puisque le système est p-dominant, nous savons que

$$\begin{aligned} \dot{V}(x) &= x^t \left(A^t P + PA\right) x, \\ &\leq -2\lambda x^t P x - \varepsilon \left|x\right|^2, \end{aligned} \tag{4.5}$$

pour tous $x \in \mathbb{R}^n$ et $\varepsilon \geq 0$. D'où, grâce à l'inégalité (4.5), on en déduit que en particulier pour $x = v_i$, on a

$$\begin{aligned} & {v_i}^t A^t P v_i + {v_i}^t P A v_i + 2\lambda x^t P x \leq 0, \\ \Leftrightarrow \quad & \left(\lambda_i + \lambda\right)\left({v_i}^t P v_i\right) \leq 0. \end{aligned}$$

Or, $\lambda_i + \lambda$ pour $i = 1, \ldots, n$ sont les valeurs propres de la matrice $A + \lambda I$. La proposition 4.1 nous assure que cette matrice admet p valeurs propres à partie réelle strictement positive, et $n - p$ autres à partie réelle strictement négative. Ces p valeurs propres ne sont rien d'autre que $\lambda_i + \lambda$ pour $i = 1, \ldots, p$. D'où, puisque $\lambda_i + \lambda$ est strictement positif, cela implique que ${v_i}^t P v_i$ est négatif. À l'inverse, tout vecteur propre v_i tel que $i \geq p + 1$ n'appartient pas à $\mathcal{K}^-$ puisque $\lambda_i + \lambda < 0$, d'ou ${v_i}^t P v_i \geq 0$. □

Ce résultat est cohérent avec l'exemple 4.1 puisqu'à la FIGURE 4.2, on voit clairement que le vecteur propre v_1 est contenu dans le cône tandis que le vecteur propre v_2 ne l'est pas : v_2 est en effet contenu dans l'autre cône $\mathcal{K}^+$. Illustrons cette propriété avec un système à trois dimensions.

Exemple 4.2. *Soit le système dynamique linéaire donné par*

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} 3 & -1 & 1 \\ -8 & -3 & -4 \\ 6 & 2 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}.$$

La matrice de ce système admet trois valeurs propres : $\lambda_1 = 6$, $\lambda_2 = 1$ *et* $\lambda_3 = -3$, *et les trois vecteurs propres à droite sont*

$$v_1 = \begin{pmatrix} -13 \\ 20 \\ -19 \end{pmatrix}, \quad v_2 = \begin{pmatrix} 1 \\ 0 \\ -2 \end{pmatrix} \quad \textit{et} \quad v_3 = \begin{pmatrix} 1 \\ 4 \\ -2 \end{pmatrix}.$$

Puisque deux des valeurs propres sont positives, il semble que le système soit 2*-dominant pour le taux* $\lambda = 0$, *autrement dit sans shift. En effet, la matrice* P *définie par*

$$P = \begin{pmatrix} 2.3184 & 1.6591 & 0.8789 \\ 1.6591 & -0.4060 & -0.4384 \\ 0.8789 & -0.4384 & -0.9124 \end{pmatrix},$$

d'inertie 2 *satisfait l'inégalité (4.1) pour* $\varepsilon = 0.01$. *Nous obtenons que*

$$\begin{aligned} V(v_1) &= {v_1}^t P\, v_1 &&= -1730.03\,(\leq 0), \\ V(v_2) &= {v_2}^t P\, v_2 &&= -4.8469\,(\leq 0), \\ V(v_3) &= {v_3}^t P\, v_3 &&= 8.9446\,(\geq 0). \end{aligned}$$

Cela signifie donc que les p premiers vecteurs propres, autrement dit v_1 et v_2 appartiennent au cône $\mathcal{K}^-$, tandis que le vecteur v_3 n'y appartient pas, mais appartient à l'autre cône $\mathcal{K}^+$.

Intéressons-nous maintenant aux systèmes non-linéaires. Comment peut-on transposer la notion de p-dominance à de tels systèmes ? Comme évoqué précédemment, nous allons opter pour l'approche différentielle. Dans un premier temps, considérons un système non-linéaire quelconque décrit par

$$\dot{x} \;=\; f(x)$$

pour $x \in \mathcal{X}$, où $\mathcal{X}$ est une variété à priori différentiable. Nous pouvons nous inspirer du cas linéaire et de l'inégalité (4.2) pour l'adapter au cas non-linéaire. Ainsi, plutôt que d'imposer cette inégalité aux états, nous allons l'imposer aux déplacement infinitésimaux sur le plan tangent, c'est-à-dire que nous avons

$$\begin{bmatrix} \dot{\delta x} \\ \delta x \end{bmatrix}^t \begin{bmatrix} O & P \\ P & 2\lambda P + \varepsilon I \end{bmatrix} \begin{bmatrix} \dot{\delta x} \\ \delta x \end{bmatrix} \;\leq\; 0, \tag{4.6}$$

où P est une matrice symétrique d'inertie p, $\varepsilon \geq 0$ et δx est un déplacement infinitésimal sur le plan tangent noté $T_x\mathcal{X}$. En réalité, puisque $\mathcal{X}$ est une variété différentiable , il est possible de définir en chaque point un plan tangent à cette variété. Ce plan peut être vu comme une approximation locale et linéaire de la variété, et c'est la raison pour laquelle nous considérons un petit déplacement sur ce plan, à savoir δx. Remarquons que pour pouvoir parler de métrique, nous supposons que la variété $\mathcal{X}$ est Riemanienne pour laquelle $|\delta x|$ dénote la métrique. Nous pouvons maintenant définir formellement la p-dominance pour les systèmes non-linéaires.

Définition 4.2

Un système non-linéaire $\dot{x} = f(x)$ est **p-dominant** avec le taux $\lambda \geq 0$ si il existe une matrice symétrique P d'inertie p telle que l'inégalité (4.6) est satisfaite par les solutions du système prolongé

$$\begin{cases} \dot{x} &= f(x), \\ \dot{\delta x} &= \partial f(x)\delta x, \end{cases} \tag{4.7}$$

où $\partial f(x)$ est la différentielle de f au point x, et $\varepsilon \geq 0$. La propriété est stricte si $\varepsilon > 0$.

Nous pouvons réfléchir exactement de la même manière que pour le cas linéaire. Le paramètre $\lambda \geq 0$ mesure toujours le shift effectué sur le spectre de la matrice Jacobienne

évaluée en chaque point de l'espace. Ainsi, contrairement au cas linéaire où la matrice Jacobienne est indépendante des variables, et donc admet n valeurs propres constantes, dans le cas non-linéaire, la matrice Jacobienne admet différentes valeurs propres pour chaque point. Toutefois, certains recouvrements sont possibles, et il est donc plus compliqué de démontrer la p-dominance de tels systèmes. En effet, puisque la Jacobienne dépend de x, il faut que le shift λ et la matrice P conviennent pour chacun des x. Il est donc important de regarder le spectre généré par la matrice Jacobienne, calculée en chaque point. Ainsi, plutôt que d'obtenir n valeurs propres ponctuelles dans le cas linéaire, nous allons obtenir n groupes de valeurs propres. Ces différents groupes de valeurs propres peuvent alors se reconvrir et éventuellement se confondre, et ainsi rendre la tâche plus complexe.

Comme nous l'avons fait dans le cas linéaire, nous pouvons exprimer la p-dominance via une fonction quadratique, à savoir

$$V(\delta x) \;=\; \delta x^t \, P \, \delta x.$$

Alors, la dérivée de cette fonction le long des trajectoires est négative puisque nous avons

$$\begin{aligned} \dot{V}(\delta x) &= \delta x^t \, \left(\partial f(x)^t P + P \partial f(x)\right) \, \delta x, \\ &\leq -2\lambda V(\delta x) - \varepsilon \, |\delta x|^2. \end{aligned}$$

Plutôt que de définir des cônes, nous allons ici définir des champs de cônes c'est-à-dire des cônes qui diffèrent en chaque point, à savoir $K^-(x)$ et $K^+(x)$ contenus dans le plan tangent et définis par

$$K^+(x) \;=\; \{\delta x \in T_x\mathcal{X} \mid V(\delta x) \geq 0\} \quad \text{et} \quad K^-(x) \;=\; \{\delta x \in T_x\mathcal{X} \mid V(\delta x) \leq 0\}\,.$$

Ces deux champs de cônes sont contractants, l'un vers l'avant et l'autre vers l'arrière puisque

$$\begin{aligned} \forall\, t \,>\, 0: \quad & \partial\psi^{-t}(x)\, \mathcal{K}^+(x) \;\subset\; \mathcal{K}^+(\psi^{-t}(x)), \\ \forall\, t \,>\, 0: \quad & \partial\psi^{t}(x)\, \mathcal{K}^-(x) \;\subset\; \mathcal{K}^-(\psi^{t}(x)). \end{aligned}$$

Dans le cas des systèmes linéaires, nous avons pu établir des caractérisations à la Proposition 4.1. Dans le cas des systèmes non-linéaires, nous pouvons plutôt établir des conditions nécessaires pour être p-dominant. La première condition concerne le spectre de la matrice Jacobienne du système.

Proposition 4.2

Soit $\dot{x} = f(x)$ pour $x \in \mathcal{X}$, un système strictement p-dominant. Alors, il existe un intervalle maximal $(\lambda_{min}, \lambda_{max})$ tel que la matrice $\partial f(x) + \lambda I$ admette p valeurs propres instables et $n-p$ stables pour tout $\lambda_{min} \leq \lambda \leq \lambda_{max}$ et pour tout $x \in \mathcal{X}$.

Ce résultat est semblable à la caractérisation pour le cas linéaire. Ceci implique qu'une analyse spectrale de la matrice Jacobienne est toujours nécessaire afin de déterminer d'une part si il est possible de diviser les valeurs propres en deux groupes de p et $n-p$ éléments. D'autre part, cette analyse permet de déterminer les paramètres p et λ.

La deuxième condition nécessaire concerne la scission invariante de l'espace tangent $T_x\mathcal{X}$ en un point x.

Proposition 4.3

Soient $\mathcal{A} \subseteq \mathcal{X}$ un ensemble compact invariant et $\dot{x} = f(x)$, un système non-linéaire strictement p-dominant avec le taux $\lambda \geq 0$. Alors, pour chaque point $x \in \mathcal{A}$, il existe une scission invariante de $T_x\mathcal{X} = \mathcal{H}_x \oplus \mathcal{V}_x$ telle que

$$\forall\, t \in \mathbb{R}, \quad \partial\psi^t(x)\mathcal{H}_x \;\subseteq\; \mathcal{H}_{\psi^t(x)},$$

$$\forall\, t \in \mathbb{R}, \quad \partial\psi^t(x)\mathcal{V}_x \;\subseteq\; \mathcal{V}_{\psi^t(x)}.$$

De plus, $\mathcal{H}_x$ et $\mathcal{V}_x$ sont de dimension p et $n-p$ respectivement, et il existe des constantes $\underline{c} \leq 1 \leq \overline{c}$ et $\underline{\lambda} < \lambda < \overline{\lambda}$ tels que

$$\forall\, x \in \mathcal{A},\, \forall\, \delta x \in \mathcal{H}_x : \quad |\partial\psi^t(x)\delta x| \;\geq\; \underline{c}\, e^{-\underline{\lambda} t}\, |\delta x|,$$

$$\forall\, x \in \mathcal{A},\, \forall\, \delta x \in \mathcal{V}_x : \quad |\partial\psi^t(x)\delta x| \;\leq\; \overline{c}\, e^{-\overline{\lambda} t}\, |\delta x|.$$

Il existe donc p directions dominantes et $n-p$ directions transitoires. Dans le cas particulier où $\mathcal{X} = \mathbb{R}^n$, nous pouvons aller plus loin et étudier le comportement asymptotique via le théorème suivant.

Théorème 4.1

Soient $\mathcal{X} = \mathbb{R}^n$ et $\dot{x} = f(x)$ un système non-linéaire strictement p-dominant pour le taux $\lambda \geq 0$. Alors, le flot sur n'importe quel ensemble compact ω-limite est topologiquement équivalent à un flot sur un ensemble compact invariant d'un système Lipschitz dans $\mathbb{R}^p$.

En d'autres mots, cela signifie que le comportement asymptotique d'un système strictement p-dominant est assimilé à un système de p dimensions. Ceci permet donc d'expliquer, dans certains cas, l'existence et/ou l'unicité de point fixe, de cycle limite, etc. En effet, un système 0-dominant peut être vu comme un système contractant. Le corollaire suivant se concentre sur les cas les plus favorables, c'est-à-dire quand p vaut 0, 1 ou 2.

Corollaire 4.1

Sous les hypothèses du théorème précédent (4.1), toute solution bornée converge asymptotiquement vers

1. un unique point fixe si $p = 0$,
2. un point fixe si $p = 1$,
3. un simple attracteur si $p = 2$, c'est-à-dire un point fixe, un ensemble de points fixes et des arcs connectants, ou un cycle limite.

Naturellement, lorsque p est grand, le concept de p-dominance n'est pas très utile puisqu'il ne permet qu'une petite réduction de dimension. Pour illustrer cette notion et les résultats que nous avons introduits, reprenons l'exemple de l'article [5] de F. Forni et R. Sepulchre.

Exemple 4.3. *Considérons le système dynamique suivant*

$$\begin{cases} \dot{x}_p &= x_v, \\ \dot{x}_v &= -\alpha(x_p) - cx_v + u, \end{cases} \tag{4.8}$$

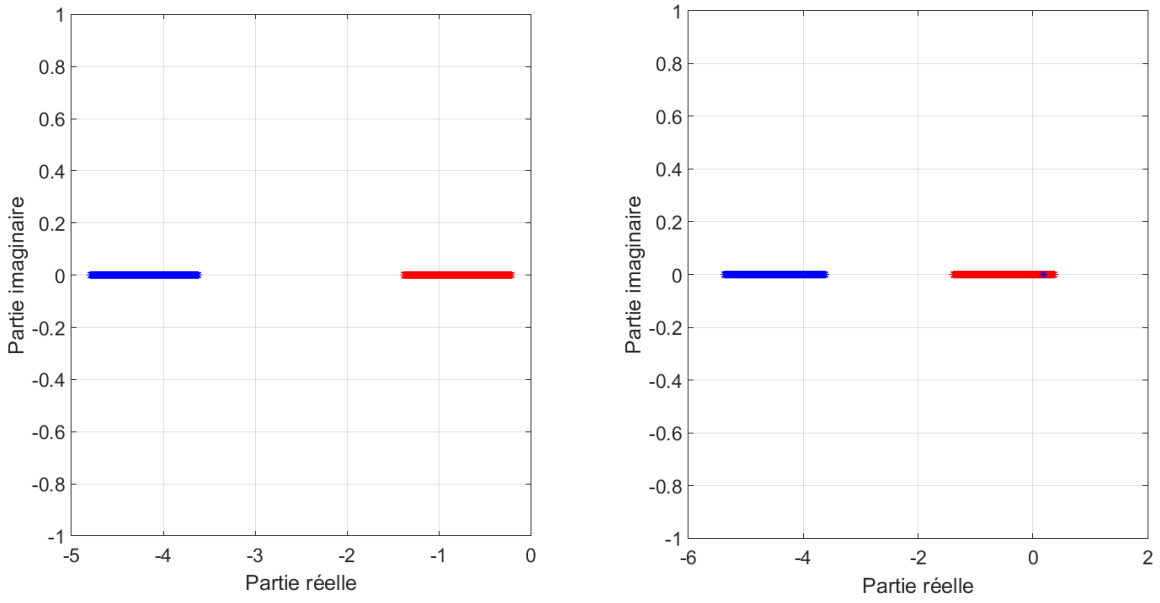
où x_p désigne la position, x_v la vitesse, u l'entrée, c le coefficient d'amortissement et l'expression $\alpha(x_p)$ dérive d'un potentiel noté U. La matrice Jacobienne associée à ce système est donnée par

$$\partial f(x) = \begin{pmatrix} 0 & 1 \\ -\partial\alpha(x_p) & -c \end{pmatrix}.$$

Supposons dans un premier temps que $c = 5$ et que $1 \leq \partial\alpha(x_p) \leq 5$. Nous pouvons alors étudier le spectre de cette matrice pour avoir une idée des paramètres λ et p. Le résultat est présenté à la FIGURE *4.3a où on y voit facilement que les valeurs propres forment deux ensembles bien distincts, sans jamais aucune valeur propre positive. D'où, le paramètre $\lambda = 0$ et $p = 0$ devrait convenir. En effet, la matrice P définie par*

$$P = \begin{pmatrix} 1.1348 & 0.1808 \\ 0.1808 & 0.1361 \end{pmatrix},$$

d'inertie 0 ($\lambda_1 = 0.1044$ et $\lambda_2 = 1.1655$) pour $\lambda = 0$ et $\varepsilon = 0.1$ satisfait bien la définition de p-dominance. Ceci signifie donc que le système admet un unique point fixe. Voyons



(a) sous l'hypothèse que $1 \leq \partial\alpha(x_p) \leq 5$ (b) sous l'hypothèse que $-2 \leq \partial\alpha(x_p) \leq 5$

FIGURE 4.3 – Analyse spectrale du système (4.8)

maintenant ce que devient la p-dominance si nous modifions les hypothèses, en particulier si nous supposons que $-2 \leq \partial\alpha(x_p) \leq 5$. *L'analyse spectrale sous cette hypothèse est illustrée à la* FIGURE *4.3b. On y voit toujours deux groupes distincts de valeurs propres sans recouvrement, toutefois certaines d'entre elles peuvent être positives. Ceci nous empêche d'avoir la* 0*-dominance, mais nous pouvons espérer la* 1*-dominance pour un taux* $\lambda = 2$ *par exemple. En effet, la matrice* P *décrite par*

$$P = \begin{pmatrix} -5.1987 & 3.6260 \\ 3.6260 & 6.1987 \end{pmatrix}$$

d'inertie 1 *(puisque les valeurs propres sont* $\lambda_1 = 7.2545$ *et* $\lambda_2 = -6.2545$*) convient pour* $\varepsilon = 0.01$. *Calculons l'ensemble* K^-. *Pour cela, nous devons dans un premier temps calculer l'expression* $V(\delta x)$. *On trouve*

$$\begin{aligned} V(\delta p, \delta v) &= 6.1987\,\delta v^2 + 2 * 3.6260\,\delta p\,\delta v - 5.1987\,\delta p^2, \\ &= 6.1987\,(\delta v - 0.5017\,\delta p)(\delta v + 1.6716\,\delta p). \end{aligned}$$

Le cône invariant vers l'avant est illustré à la FIGURE *4.4. Supposons maintenant que l'entrée* u *s'exprime comme*

$$u = k_f x_i, \tag{4.9}$$

où l'évolution de x_i *est décrite par l'équation*

$$\dot{x}_i = -10x_i - 10x_v + V. \tag{4.10}$$

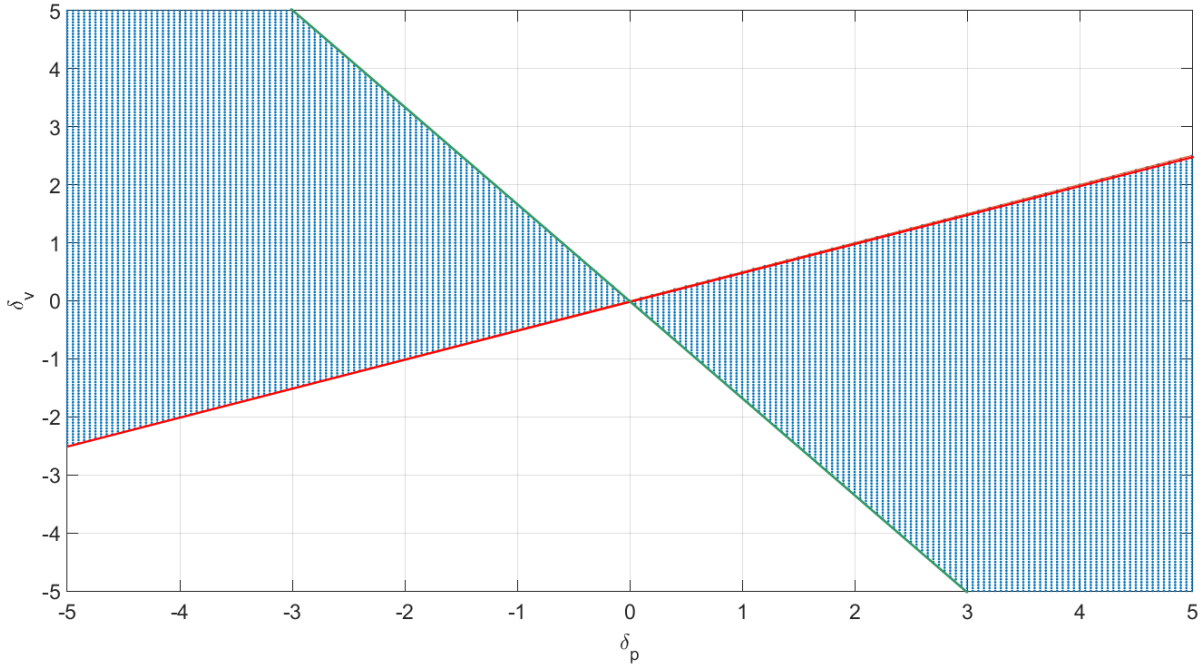
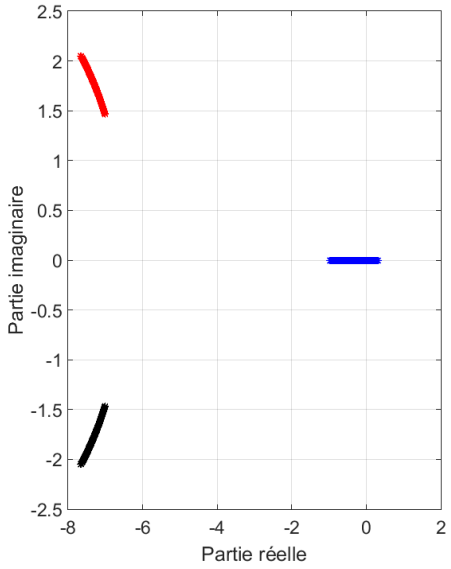
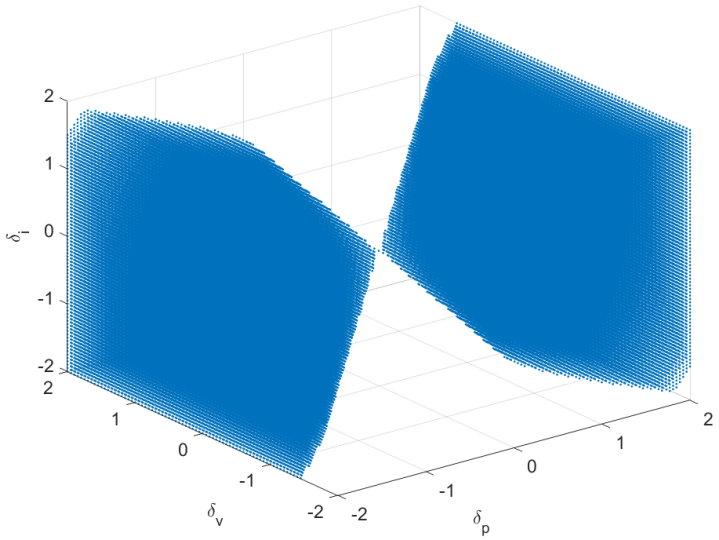


FIGURE 4.4 – Illustration du cône K^- en bleu, des délimitations ($\delta v = 0.5017\,\delta p$ en rouge et $\delta v = -1.6716\,\delta p$ en vert) du système (4.8)



(a) Analyse spectrale du système



(b) Illustration du cône K^-

FIGURE 4.5 – Illustration du système (4.4) avec l'entrée u définie par (4.9) et régie par (4.10)

où le voltage V est une entrée supplémentaire. Nous pouvons alors étudier le spectre de la matrice Jacobienne associée à ce nouveau système sous l'hypothèse que $-2 \leq \partial\alpha(x_p) \leq 5$. Le résultat est visible à la FIGURE *4.5a. On y voit que la 0-dominance est exclue puisque certaines valeurs propres peuvent être positives, mais le système peut à priori être 1-dominant pour un taux $\lambda = 2$ par exemple. En effet, nous pouvons trouver une matrice P définie par exemple par*

$$P = \begin{pmatrix} -3.0947 & 0.9134 & -0.5297 \\ 0.9134 & 3.3774 & 0.1989 \\ -0.5297 & 0.1989 & 0.7173 \end{pmatrix}$$

d'inertie 1 puisque ses valeurs propres sont $\lambda_1 = 3.5095$, $\lambda_2 = 0.7876$ et $\lambda_3 = -3.2971$ pour $\varepsilon = 0.01$.

Remarque : pour les systèmes linéaires, nous avons remarqué que la notion de p-dominance pouvait être étendue aux systèmes dynamiques ouverts. Nous pouvons faire de même dans le cadre des systèmes non-linéaires. En effet, si nous considérons un système de la forme

$$\begin{cases} \dot{x} &= f(x) + Bu, \\ y &= Cx, \end{cases}$$

où $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ et f est une fonction lisse, le système prolongé contient le système précédent ainsi que

$$\begin{cases} \dot{\delta x} &= \partial f(x)\, \delta x + B\, \delta u, \\ \delta y &= C\, \delta x. \end{cases}$$

On dira alors qu'un système dynamique non-linéaire est p-dissipatif avec le taux $\lambda \geq 0$ si le système prolongé satisfait la contrainte

$$\begin{bmatrix} \dot{\delta}x \\ \delta x \end{bmatrix}^t \begin{bmatrix} O & P \\ P & 2\lambda P + \varepsilon I \end{bmatrix} \begin{bmatrix} \dot{\delta}x \\ \delta x \end{bmatrix} \leq \underbrace{\begin{bmatrix} \delta y \\ \delta u \end{bmatrix}^t \begin{bmatrix} Q & L \\ L^t & R \end{bmatrix} \begin{bmatrix} \delta y \\ \delta u \end{bmatrix}}_{:= s(y,u)}, \tag{4.11}$$

pour tout $\delta x \in \mathbb{R}^n$ et tout $\delta u \in \mathbb{R}^m$ et avec P une matrice symétrique d'inertie p, Q, L et R des matrices aux dimensions appropriées et $\varepsilon \geq 0$. La propriété est dite stricte si $\varepsilon > 0$. De plus, l'inégalité (4.11) est équivalence à l'inégalité suivante

$$\begin{bmatrix} \partial f(x)^t P + P\partial f(x) - C^t Q C + 2\lambda P + \varepsilon I & PB - C^t L - C^t Q D \\ B^t P - L^t C - D^t Q C & -R - D^t L - L^t D - D^t Q D \end{bmatrix} \leq 0. \tag{4.12}$$

Il est donc possible d'obtenir la p-dissipativité grâce à la résolution d'inégalités linéaires matricielles (LMIs), tout comme pour la p-dominance.

4.2 Positivité différentielle

Avant de s'intéresser à la p-dominance, F. Forni et R. Sepulchre se sont penchés sur la notion de positivité différentielle [4] qui peut être vu comme un cas particulier de la p-dominance. En réalité, les systèmes strictement 1-dominant sont différentiellement positifs. Les deux concepts utilisent tous les deux la même approche, à savoir l'approche différentielle. En effet, la positivé différentielle se base sur la notion de système linéaire positif, une propriété qui a été énormément exploitée jusqu'à présent. Ceci a donc poussé F. Forni et R. Sepulchre à tenter de généraliser le concept aux dynamiques non-linéaires.

La première étape de l'approche différentielle est de définir la notion dans le cas linéaire, pour ensuite pouvoir le généraliser aux systèmes non-linéaires. Commençons donc par définir ce qu'est un système linéaire positif.

Définition 4.3

Un système $\dot{x} = Ax$ pour $A \in \mathbb{R}^{n\times n}$ est **positif** s'il existe un cône $\mathcal{K} \subseteq \mathbb{R}^n$ invariant vers l'avant, autrement dit

$$e^{At}\mathcal{K} \subseteq \mathcal{K} \tag{4.13}$$

pour tout temps $t \geq 0$.

Dans le cas d'un système positif, le cône invariant peut être obtenu grâce aux vecteurs propres à gauche de la matrice A. En effet, le cône doit uniquement contenir la direction la plus lente, c'est-à-dire le vecteur propre à droite associé à la valeur propre dont la partie réelle est la plus grande. Le théorème suivant, fortement inspiré de la référence [10] propose un cône invariant $\mathcal{K}$.

Théorème 4.2

Soit un système linéaire $\dot{x} = Ax$ où $A \in \mathbb{R}^{n\times n}$, $\lambda_1, \ldots, \lambda_n$ les n valeurs propres de A de sorte que $R(\lambda_1) \geq R(\lambda_j)$ pour $j = 1, \ldots, n$, et les n vecteurs propres linéairement indépendants à gauche et à droite dénotés par w_j et v_j, pour $j = 1, \ldots, n$ respectivement. L'ensemble défini par l'expression

$$\mathcal{K} \;=\; \left\{x \in \mathbb{R}^n \mid {w_1}^t x - \left|{w_j}^t x\right| \geq 0,\ \forall j \geq 2\right\}. \tag{4.14}$$

est un cône et satisfait $e^{At}\mathcal{K} \subseteq \mathcal{K}$.

Démonstration. Démontrons dans un premier temps que l'ensemble $\mathcal{K}$ est un cône. Pour ce faire, nous devons démontrer plusieurs propriétés :

(i) $\alpha\mathcal{K} \subseteq \mathcal{K}$ pour tout $\alpha > 0$. Soient $\alpha > 0$, et $x \in \mathcal{K}$. Le vecteur αx appartient à $\mathcal{K}$ si et seulement si

$$\begin{aligned} & {w_1}^t(\alpha x) - |{w_j}^t(\alpha x)| \;\geq\; 0, \\ \Leftrightarrow \qquad & \alpha\left({w_1}^t x - |{w_j}^t x|\right) \;\geq\; 0, \end{aligned}$$

ce qui est bel et bien vérifié puisque x appartient à $\mathcal{K}$ par hypothèse et que le paramètre α est strictement positif.

(ii) $\mathcal{K} + \mathcal{K} \subseteq \mathcal{K}$. Soient $x, y \in \mathcal{K}$. Démontrons que leur somme appartient toujours à cet ensemble. Or, nous savons que

$$\begin{aligned} {w_1}^t(x+y) - |{w_j}^t(x+y)| \;&=\; {w_1}^t x + {w_1}^t y - |{w_j}^t x + {w_j}^t|\,, \\ &\geq\; {w_1}^t x - |{w_j}^t x| + {w_1}^t y - |{w_j}^t|\,, \\ &\geq\; 0, \end{aligned}$$

puisque x et y appartiennent à $\mathcal{K}$ par hypothèse.

(iii) $\mathcal{K} \cap \mathcal{K} = \{0\}$. Soient $x \in \mathcal{K}$ et $x \in -\mathcal{K}$. Cela signifie que

$$\begin{aligned} {w_1}^t x - |{w_j}^t x| \;&\geq\; 0, \\ -{w_1}^t x - |{w_j}^t x| \;&\geq\; 0. \end{aligned}$$

Cela implique que ${w_i}^t x = 0$ pour tout $j = 1, \ldots, n$. Puisque la matrice A admet n vecteurs propres linéairement indépendants, on en conclut que $x = 0$.

Démontrons finalement que le cône satisfait la condition de positivité différentielle, c'est-à-dire

$$e^{At}\mathcal{K} \subseteq \mathcal{K}.$$

Soit $x \in \mathcal{K}$. Montrons que $e^{At}x$ appartient également à $\mathcal{K}$. Or, nous savons que

$$\begin{aligned} e^{At}x \in \mathcal{K} &\Leftrightarrow {w_1}^t\left(e^{At}x\right) - \left|{w_j}^t\left(e^{At}x\right)\right| \geq 0, \\ &\Leftrightarrow e^{\lambda_1 t}{w_1}^t x - \left|e^{\lambda_j t}{w_j}^t x\right| \geq 0, \\ &\Leftrightarrow e^{\lambda_1 t}\left({w_1}^t x - e^{(Re(\lambda_j)-\lambda_1)t}\left|{w_j}^t x\right|\right) \geq 0, \end{aligned} \tag{4.15}$$

L'expression $Re(\lambda_j) - \lambda_1$ admet toujours une partie réelle négative par hypothèse, impliquant que l'exponentielle est inférieure à 1. D'où, on en déduit directement l'inégalité (4.15) puisque $x \in \mathcal{K}$ par hypothèse. □

Ceci s'explique très facilement puisque nous savons que tout vecteur $x \in \mathbb{R}^n$ peut s'exprimer dans la base formée par les vecteurs propres à droite de la matrice A. Il existe donc des constantes $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ telles que

$$x = \sum_{i=1}^{n} \alpha_i v_i.$$

Or, nous savons que les vecteurs propres à gauches sont orthogonaux à tous les vecteurs propres à droite, hormis celui associé à la même valeur propre. De cette façon, lorsque nous calculons le produit scalaire entre w_i et x, nous obtenons en réalité la $i$ème composante du vecteur x exprimé dans la base des $\{v_i\}_{i=1,\ldots,n}$ multiplié par le produit scalaire entre w_i et v_i. Notons tout de même qu'il est possible de prendre w_i et v_i de sorte que leur produit scalaire vaille 1. Dans ce cas, la condition (4.14) revient à dire que la composante associé v_1 doit être supérieure à la valeur absolue de la composante associée à tous les autres vecteurs v_i, $i = 2, \ldots, n$. Ainsi, le cône $\mathcal{K}$ contient le vecteur propre v_1 sans contenir aucun des autres vecteurs propres à droite. Finalement, illustrons ces notions via quelques exemples.

Exemple 4.4. *Soit le système décrit par*

$$\begin{cases} \dot{x}_1 &= -x_1 + k(x_2 - x_1), \\ \dot{x}_2 &= -x_2 + k(x_1 - x_2), \end{cases}$$

où k est un paramètre strictement positif. Ceci est bien un système linéaire, et les valeurs propres de la matrice sont $\lambda_1 = -1$ et $\lambda_2 = -1 - 2k$. Puisque la matrice est symétrique, les vecteurs propres à gauche et à droite sont confondus. Nous avons donc que $v_1 = w_1 = (1, 1)^t$ et $v_2 = w_2 = (1, -1)^t$. Alors, la condition (4.14) peut se réécrire comme

$$x_1 + x_2 - |x_1 - x_2| \geq 0.$$

Dans le cas où $x_1 \geq x_2$, on obtient que x_2 doit être positif ou nul tandis que dans le cas où $x_2 \geq x_1$, la condition se réduit à imposer que x_1 soit positif ou nul. Autrement dit, il faut que x_1 et x_2 soient positifs ou nuls. Ceci signife que le cône $\mathcal{K}$ n'est rien d'autre que l'orthant positif, à avoir $\mathbb{R}^2_+$.

La positivité différentielle est une manière d'étendre cette définition aux systèmes non-linéaires, en exigeant que leur linéarisation soit positive. Autrement dit, cela signifie que le système prolongé admette un champ de cônes invariants vers l'avant. Avant tout, il est important de revenir sur la notion de système prolongé et l'opérateur de Koopman. Supposons que $\dot{x} = f(x)$ est un système dynamique non-linéaire noté Σ, défini sur $\mathcal{X}$, une variété Riemanienne de n dimensions. La dynamique prolongée $\delta\Sigma$ de Σ est données par (4.7). Comment définir l'opérateur de Koopman pour un tel système ? En réalité, nous pouvons définir le semi-groupe d'opérateurs lié au système augmenté, notés $\tilde{U}^t$, par

$$\tilde{U}^t \tilde{g}(x\,,\,\delta x) \;:=\; \tilde{g}\left(\psi^t(x)\,,\,\partial\psi^t(x)\delta x\right), \tag{4.16}$$

où $\tilde{g}$ est une observable définie sur $T\mathcal{X} = \cup_{x\in\mathcal{X}}\{x\}\times T_x\mathcal{X}$, et à valeurs complexes. Puisque l'opérateur de Koopman est linéaire, qu'il soit défini pour le système initial ou pour le système prolongé, il admet toujours des valeurs et fonctions propres. Il existe alors une manière de connecter l'analyse spectrale d'un système à celle de l'autre. L'article [10] a établi un lemme.

Lemme 4.1

Soit $\phi_\lambda \in \mathbb{C}^1(\mathcal{X})$ une fonction propre de U^t associé au système Σ. Alors, l'opérateur de Koopman $\tilde{U}^t$ associé au système prolongé $\delta\Sigma$ admet les fonctions propres

$$\tilde{\phi}_\lambda^{(1)}(x,\delta x) \;=\; \phi_\lambda(x), \tag{4.17}$$

$$\tilde{\phi}_\lambda^{(2)}(x,\delta x) \;=\; \partial\phi_\lambda(x)\delta x, \tag{4.18}$$

pour tout $(x\,,\,\delta x) \in T\mathcal{X}$.

Nous pouvons maintenant définir la positivité différentielle pour un système non-linéaire.

Définition 4.4

Le système Σ est **différentiellement positif** (par rapport au champ de cônes $\mathcal{K}$) si le flot du système prolongé $\delta\Sigma$ laisse le cône invariant, i.e.

$$\partial\psi^t(x)\mathcal{K}(x) \subseteq \mathcal{K}(\psi^t(x)), \tag{4.19}$$

pour tout $x \in \mathcal{X}$ et pour tout temps $t > 0$. La propriété est stricte si il existe une constante T et un champ de cônes $\mathcal{R}(x) \subset int\mathcal{K} \cup \{0\}$ tels que

$$\partial\psi^t(x)\mathcal{K}(x) \subseteq \mathcal{R}(\psi^t(x)),\ \forall x \in \mathcal{K},\ \forall t > T.$$

Le principe de la positivité différentielle est similaire au principe la positivité des systèmes linéaires. Dans chaque cas, les trajectoires sont contraintes de la même manière et tendent vers un attracteur de dimension 1. Dans le cas linéaire, c'est typiquement une ligne, un rayon tandis que dans le cas non-linéaire, cela peut-être une courbe. Le système peut donc admettre plusieurs points fixes reliés par des arcs, ou encore un cycle limite. Le plus intéressant pour nous concerne le lien qu'il est possible d'établir entre positivité

différentielle et les fonctions propres de l'opérateur de Koopman. En effet, un théorème a été établi dans l'article [10] de A. Mauroy, F. Forni et R. Sepulchre (Proposition 1, [10], p.3), permettant de conclure qu'un système est différentiellement positif et permettant de calculer l'expression du champ de cône $\mathcal{K}(x)$. Reprenons ce résultat, et démontrons-le.

Théorème 4.3

Supposons qu'un système Σ de n dimensions admette un ensemble de fonctions propres de Koopman $\phi_{\lambda_j} \in C^1(\mathcal{X})$ pour $j = 1, \ldots, n$, tel que $Re(\lambda_1) \geq Re(\lambda_j)$. De plus, supposons que l'application linéaire $\partial\Psi(x) : T_x\mathcal{X} \to \mathbb{C}^n$ définie par

$$\partial\Psi(x)\delta x \;=\; [\partial\phi_{\lambda_1}(x)\partial_x, \ldots, \partial\phi_{\lambda_n}(x)\partial_x]$$

soit injective pour tout $x \in \mathcal{X}$. Alors, le système Σ est différentiellement positif si l'une des conditions suivantes est satisfaite :

(1) $\lambda_1 \in \mathbb{R}$. Le champ de cônes $\mathcal{K}(x)$ peut être défini comme

$$\mathcal{K}(x) \;=\; \left\{\delta x \in T_x\mathcal{X} \mid \partial\phi_{\lambda_1}(x)\delta x - \left|\partial\phi_{\lambda_j}(x)\delta x\right| \geq 0,\ \forall j \geq 2\right\}. \tag{4.20}$$

(2) $\lambda_1 \in i\mathbb{R}$ avec $\angle\phi_{\lambda_1} \in \mathbb{C}^1(\mathcal{X})$ et $|\phi_{\lambda_1}|$ est constant sur $\mathcal{X}$. Dans ce cas-là, le champ de cônes peut être défini comme

$$\mathcal{K}(x) \;=\; \left\{\delta x \in T_x\mathcal{X} \mid \partial\angle\phi_{\lambda_1}(x)\delta x - \left|\partial\phi_{\lambda_j}(x)\delta x\right| \geq 0,\ \forall j \geq 2\right\}. \tag{4.21}$$

Le système est strictement différentiellement positivif si $Re(\lambda_1) > Re(\lambda_j)$ pour $j = 2, \ldots, n$.

Démonstration. Démontrons dans un premier temps le cas réel, et montrons que l'ensemble défini est bel et bien un cône. Pour ce faire, nous devons démontrer trois propriétés. Soit $x \in \mathcal{X}$. Montrons que

(i) $\alpha\mathcal{K}(x) \subseteq \mathcal{K}(x)$ pour $\alpha > 0$. Soit $\alpha > 0$ et $\delta x \in \mathcal{K}(x)$. Nous savons que

$$\begin{aligned}
\partial\phi_{\lambda_1}(x)(\alpha\,\delta x) - \left|\partial\phi_{\lambda_j}(x)(\alpha\,\delta x)\right| &= \alpha\,\partial\phi_{\lambda_1}(x)\delta x - \left|\alpha\,\phi_{\lambda_j}(x)\delta x\right|, \\
&= \alpha\,\underbrace{\left(\partial\phi_{\lambda_1}(x)\delta x - \left|\partial\phi_{\lambda_j}(x)\delta x\right|\right)}_{\geq 0} \\
&\geq 0,
\end{aligned}$$

pour tout $j \geq 2$, puisque $\delta x \in \mathcal{K}(x)$ par hypothèse. Cela signifie donc que $\alpha\,\delta x$ appartient bien à $\mathcal{K}(x)$.

(ii) $\mathcal{K}(x) + \mathcal{K}(x) \subseteq \mathcal{K}$. Soient δx_1 et $\delta x_2 \in \mathcal{K}(x)$. L'expression $\delta x_1 + \delta x_2$ appartient à $\mathcal{K}(x)$ si et seulement si

$$\partial\phi_{\lambda_1}(x)(\delta x_1 + \delta x_2) - \left|\partial\phi_{\lambda_j}(x)(\delta x_1 + \delta x_2)\right| \;\geq\; 0.$$

Or, si nous calculons le membre de gauche, nous obtenons

$$\begin{aligned}
& \partial\phi_{\lambda_1}(x)(\delta x_1+\delta x_2) - \left|\partial\phi_{\lambda_j}(x)(\delta x_1+\delta x_2)\right| \\
= \;& \partial\phi_{\lambda_1}(x)\delta x_1 + \partial\phi_{\lambda_1}(x)\delta x_2 - \left|\partial\phi_{\lambda_j}(x)\delta x_1 + \partial\phi_{\lambda_j}(x)\delta x_2\right|, \\
\geq \;& \partial\phi_{\lambda_1}(x)\delta x_1 + \partial\phi_{\lambda_1}(x)\delta x_2 - \left|\partial\phi_{\lambda_j}(x)\delta x_1\right| - \left|\partial\phi_{\lambda_j}(x)\delta x_2\right|, \\
\geq \;& 0,
\end{aligned}$$

pour tout $j \geq 2$, puisque par hypothèse, δx_1 et δx_2 appartiennent à $\mathcal{K}(x)$. Ceci signifie donc que $\delta x_1 + \delta x_2 \in \mathcal{K}(x)$.

(iii) $\mathcal{K}(x) \cap -\mathcal{K}(x) = \{0\}$. Soit $\delta x \in \mathcal{K}(x)$ et $-\mathcal{K}(x)$, cela signifie tout d'abord que

$$\partial\phi_{\lambda_1}(x)\delta x - \left|\partial\phi_{\lambda_j}(x)\delta x\right| \geq 0,$$

impliquant que $\partial\psi_{\lambda_1}(x)\delta x \geq 0$, et ensuite que

$$\begin{aligned}
& \partial\phi_{\lambda_1}(x)(-\delta x) - \left|\partial\phi_{\lambda_j}(x)(-\delta x)\right| \geq 0, \\
\Leftrightarrow \quad & -\partial\phi_{\lambda_1}(x)\delta x - \left|\partial\phi_{\lambda_j}(x)\delta x\right| \geq 0.
\end{aligned}$$

Si on combine les deux inégalités, on obtient que $\partial\phi_{\lambda_j}(x)\delta x = 0$, pour tout $j \geq 2$. Puisque l'application $\partial\Psi(x)$ est injective, cela implique que $\delta x = 0$.

Démontrons finalement que le cône $\mathcal{K}(x)$ satisfait bien la définition (4.19) de positivité différentielle. Soient $x \in \mathcal{X}$ et $\delta x \in \mathcal{K}(x)$. Reprenons l'expression utilisée pour définir le cône $\mathcal{K}(x)$ mais pour les fonctions propres de système augmenté. Nous avons que

$$\begin{aligned}
& \partial\phi_{\lambda_1}(\psi^t(x)\,,\,\partial\psi^t(x)\delta x) - \left|\partial\phi_{\lambda_j}(\psi^t(x)\,,\,\partial\psi^t(x)\delta x)\right| \\
\underset{(4.16)}{=} \;& \tilde{U}^t\phi^{(2)}_{\lambda_1}(x,\delta x) - \left|\tilde{U}^t\phi^{(2)}_{\lambda_j}(x,\delta x)\right|, \\
\underset{(1.6)}{=} \;& e^{\lambda_1 t}\phi^{(2)}_{\lambda_1}(x,\delta x) - \left|e^{\lambda_j t}\phi^{(2)}_{\lambda_j}(x,\delta x)\right|, \\
= \;& e^{\lambda_1 t}\left(\phi^{(2)}_{\lambda_1}(x,\delta x) - e^{(R(\lambda_j)-\lambda_1)t}\left|\phi^{(2)}_{\lambda_j}(x,\delta x)\right|\right), \\
\underset{\text{lemme 4.2}}{=} \;& e^{\lambda_1 t}\left(\partial\phi_{\lambda_1}\delta x - e^{(Re(\lambda_j)-\lambda_1)t}\left|\partial\phi_{\lambda_j}\delta x\right|\right), \\
\underset{\lambda_1 \geq Re(\lambda_j)}{\geq} \;& e^{\lambda_1 t}\left(\partial\phi_{\lambda_1}\delta x - \left|\partial\phi_{\lambda_j}\delta x\right|\right), \\
\geq \;& 0,
\end{aligned}$$

puisque δx appartient au cône par hypothèse. D'où, $\partial\psi^t(x)$ appartient également au cône. Par ailleurs, si on suppose que $\lambda_1 > Re(\lambda_j)$, alors l'inégalité devient stricte et on a la positivité différentielle stricte.

Supposons maintenant que λ_1 soit un imaginaire pur. De la même manière que pour le cas réel, nous devons dans un premier temps montrer que l'ensemble $\mathcal{K}(x)$ décrit bien un cône. Toutefois, la démonstration pour le cas linéaire est transposable au cas complexe, hormis pour le troisième et dernier point. En effet, nous savons que

$$\partial\angle\phi_{\lambda_1}(x)\,\delta x - \left|\partial\phi_{\lambda_j}(x)\,\delta x\right| \;\geq\; 0,$$

et

$$-\partial\angle\phi_{\lambda_1}(x)\,\delta x - \left|\partial\phi_{\lambda_j}(x)\,\delta x\right| \;\geq\; 0,$$

pour tout $j \geq 2$. On en déduit donc que

$$\left\{\begin{array}{rcl} \partial\angle\phi_{\lambda_1}(x)\,\delta x &=& 0, \\ \partial\phi_{\lambda_j}(x)\,\delta x &=& 0,\ \forall j \geq 2. \end{array}\right.$$

Or, nous savons que le module de la fonction propre ϕ_{λ_1} est constant d'où nous pouvons décomposer sa dérivée comme

$$\begin{aligned} \partial\phi_{\lambda_1} &= \partial\left(|\phi_{\lambda_1}|\, e^{i\angle\phi_{\lambda_1}}\right), \\ &= i\phi_{\lambda_1}\partial\angle\phi_{\lambda_1}. \end{aligned} \tag{4.22}$$

On en déduit facilement que $\partial\phi_{\lambda_1}(x)\delta x = 0$. Par injectivité de la fonction $\partial\Psi$, on en conclut que $\delta x = 0$. Ainsi, l'ensemble $\mathcal{K}(x)$ est bien un cône.

Démontrons maintenant que le cône satisfait la définition de positivité différentielle. Autrement dit nous devons montrer que

$$\partial\psi^t(x)\mathcal{K}(x) \subseteq \mathcal{K}(\psi^t(x)),$$

pour tout $x \in \mathcal{X}$. Soient $x \in \mathcal{X}$ et $\delta x \in \mathcal{K}(x)$. L'inclusion est démontrée si et seulement si nous démontrons que

$$\underbrace{\partial\angle\phi_{\lambda_1}(\psi^t(x))\left[\partial\psi^t(x)\,\delta x\right] - \left|\partial\phi_{\lambda_j}(\psi^t(x))\left[\partial\psi^t(x)\,\delta x\right]\right|}_{(*)} \;\geq\; 0,$$

pour tout $j \geq 2$. Tout d'abord, si nous utilisons la relation (4.22), nous pouvons écrire

$$\begin{aligned} \partial\angle\phi_{\lambda_1}(x)\delta x &= \frac{\partial\phi_{\lambda_1}(x)\,\delta x}{i\phi_{\lambda_1}(x)}, \\ &= \frac{\tilde{\phi}^{(2)}_{\lambda_1}(x,\delta x)}{i\phi_{\lambda_1}(x)}. \end{aligned} \tag{4.23}$$

Alors, l'expression $(*)$ peut être réécrite comme

$$\begin{aligned} (*) \quad &= \quad \frac{\tilde{U}^t\tilde{\phi}^{(2)}_{\lambda_1}(x,\delta x)}{iU^t\phi_{\lambda_1}(x)} - \left|\tilde{U}^t\tilde{\phi}^{(2)}_{\lambda_j}(x,\delta x)\right|, \\ &= \quad \frac{\tilde{\phi}^{(2)}_{\lambda_1}(x,\delta x)}{i\phi_{\lambda_1}(x)} - \left|e^{\lambda_j t}\tilde{\phi}^{(2)}_{\lambda_j}(x,\delta x)\right|, \\ &\underset{Re(\lambda_j)\leq 0}{\geq} \quad \frac{\tilde{\phi}^{(2)}_{\lambda_1}(x,\delta x)}{i\phi_{\lambda_1}(x)} - \left|\tilde{\phi}^{(2)}_{\lambda_j}(x,\delta x)\right|, \end{aligned}$$

$$\begin{aligned} &\underset{(4.23)}{=} \quad \partial\angle\phi_{\lambda_1}(x)\,\delta x - \left|\partial\phi_{\lambda_j}(x)\delta x\right|, \\ &\geq \quad 0, \end{aligned}$$

puisque $\delta x \in \mathcal{K}(x)$ par hypothèse. La preuve pour la positivité différentielle stricte est identique à celle pour le cas réel. □

Nous pouvons facilement remarquer que l'expression du cône $\mathcal{K}(x)$ est semblable au cône (4.14) que nous avons défini dans le cas linéaire. En réalité, il s'agit juste d'une simple généralisation où les vecteurs à gauche ont été remplacés par les dérivées des fonctions propres de l'opérateur de Koopman.

Remarque : la positivité différentielle n'est pas réellement équivalente à la 1-dominance, comme le suggère les références [5] et [6]. En effet, si un système est strictement 1-dominant, nous savons que les trajectoires linéarisées du systèmes vont rejoindre l'intérieur du cône $\mathcal{K}^-$, et c'est exactement ce qui est requis pour qu'un système soit strictement différentiellement positif. Le cône défini par la 1-dominance peut être vu comme l'union de deux cônes convexes notés $\mathcal{K}_1$ et $\mathcal{K}_2$, dont l'intersection est uniquement le vecteur nul. Ainsi, la 1-dominance stricte implique la positivité différentielle stricte avec $\mathcal{K}_1$ (ou $\mathcal{K}_2$) comme cône. La différence entre les deux théorie est la même différence qui existe entre contraction projective et contraction verticale. En réalité, la contraction d'un cône est dite projective car cela exprime le fait que les directions transitoires du flot se contractent par rapport aux directions dominantes. De plus, la dominance implique également la contraction verticale, c'est-à-dire la contraction du flot $\partial\psi^t$ dans $\mathcal{V}$ comme nous l'avons énoncé à la Proposition 4.3. Il est important de noter que sans cette contraction, le Théorème 4.1 ne tient pas. Or, la positivité différentielle n'implique pas la contraction verticale dans $\mathcal{V}_x$. C'est également la raison pour laquelle un système différentiellement positif peut admettre un cycle limite tandis qu'un système 1-dominant admet uniquement un ou plusieurs points fixes. Ces remarques sont détaillées dans [6, sections V.B et V.C], dans [5, section II] et dans [4, section VII.B].

4.3 Caractérisation via l'opérateur de Koopman

Dans les sections précédentes, nous avons introduit les concepts de p-dominance et de positivité différentielle. D'une part, nous avons pu démontrer que l'opérateur de Koopman peut être utilisé pour trouver le cône (4.20-4.21) invariant vers l'avant, impliqué dans la définition de la positivité différentielle. Nous avons procédé de manière différentielle, c'est-à-dire que nous avons tout d'abord trouvé un cône dans le cas linéaire, et nous l'avons par la suite étendu aux systèmes non-linéaires via la matrice Jacobienne. D'autre part, positivité différentielle et p-dominant sont étroitements liés puisque tout système strictement 1-dominant est différentiellement positif. Ainsi, si on considère que la p-dominance est une généralisation la positivité différentielle, il devrait être possible de généraliser le cône invariant défini grâce à l'opérateur de Koopman. Procédons encore une fois de manière différentielle, et tentons d'établir dans un premier temps un cône pour un système dynamique linéaire p-dominant. Nous savons que pour la positivité différentielle, ce cône

contient le vecteur propre associé à la valeur propre dominante. Or, nous avons démontré (Propriété 4.1) dans la première section de ce chapitre que dans le cadre d'un système linéaire, le cône $\mathcal{K}^-$ contient les p vecteurs propres de la matrice du système associés aux p valeurs propres qui admettent les plus grandes parties réelles. L'objectif ici est donc de construire un cône similaire à celui défini pour la positivité différentielle (4.14) de sorte qu'il contienne les p premiers vecteurs propres.

Théorème 4.4

Soit un système dynamique linéaire $\dot{x} = Ax$ où $A \in \mathbb{R}^{n\times n}$. Supposons que A admette n valeurs propres λ_i pour $i = 1, \ldots, n$ telles que $Re(\lambda_1) \geq \ldots \geq Re(\lambda_n)$, qu'il existe n vecteurs propres à gauche et à droite linéairement indépendants notés w_i et v_i respectivement. Alors, l'ensemble défini par

$$\mathcal{K}^- = \left\{ x \in \mathbb{R}^n \mid \sum_{i=1}^{p} {w_i}^t x - \left|{w_j}^t x\right| \geq 0,\ \forall j \geq p+1 \text{ et } {w_i}^t x \geq 0,\ 1 \leq i \leq p \right\} \quad (4.24)$$

est un cône et invariant vers l'avant, c'est-à-dire $e^{At}\mathcal{K}^- \subseteq \mathcal{K}^-$.

Démonstration. Démontrons dans un premier temps que l'ensemble $\mathcal{K}^-$ est un cône. Pour ce faire, démontrons les trois propriétés nécessaires :

(i) $\alpha\mathcal{K}^- \subseteq \mathcal{K}^-$, pour tout $\alpha > 0$. Soit $\alpha > 0$ et $x \in \mathcal{K}^-$, le vecteur αx appartient à $\mathcal{K}^-$ si et seulement si pour tout $j \geq p+1$

$$\begin{aligned} \sum_{i=1}^{p} {w_i}^t(\alpha x) - |{w_j}^t(\alpha x)| &\geq 0, \\ \Leftrightarrow \quad \alpha\left(\sum_{i=1}^{p} {w_i}^t x - |{w_j}^t x|\right) &\geq 0. \end{aligned}$$

Cette inégalité est satisfaite puisque $x \in \mathcal{K}^-$ par hypothèse.

(ii) $\mathcal{K}^- + \mathcal{K}^- \subseteq \mathcal{K}^-$. Soient $x, y \in \mathcal{K}^-$. Démontrons que leur somme reste dans cet ensemble. Nous savons que

$$\begin{aligned} \sum_{i=1}^{p} {w_i}^t(x+y) - |{w_j}^t(x+y)| &= \sum_{i=1}^{p} {w_i}^t x + \sum_{i=1}^{p} {w_i}^t y - |{w_j}^t x + {w_j}^t y|, \\ &\geq \sum_{i=1}^{p} {w_i}^t x + \sum_{i=1}^{p} {w_i}^t y - |{w_j}^t x| - |{w_j}^t y|, \\ &\geq 0, \end{aligned}$$

puisque $x, y \in \mathcal{K}^-$.

(iii) $\mathcal{K}^- \cap -\mathcal{K}^- = \{0\}$. Supposons que $x \in \mathcal{K}^-$ et $x \in -\mathcal{K}^-$ et démontrons que cela implique que x soit le vecteur nul. Nous savons par hypothèse que pour tout $j = p+1, \ldots, n$,

$$\sum_{i=1}^{p} {w_i}^t x - \left|{w_j}^t x\right| \geq 0,$$

et

$$-\sum_{i=1}^{p} {w_i}^t x - \left|{w_j}^t x\right| \geq 0.$$

D'où, on en déduit que pour tout $j > p$, le vecteur x est orthogonal à w_j, et la somme des produits scalaires ${w_i}^t x$ est nulle. Or, nous savons par hypothèse que $x \in \mathcal{K}^-$. Ainsi, pour tout $i = 1, \ldots, p$, ${w_i}^t x \geq 0$. Alors, le vecteur x est orthogonal à tous les vecteurs propres à gauche. On en déduit donc que $x = 0$.

Nous venons donc de démontrer que $\mathcal{K}^-$ est bel et bien un cône. Démontrons à présent que ce cône est invariant vers l'avant. Pour ce faire, procédons de la même manière que pour le cas initial où $p = 1$. Soit $x \in \mathcal{K}^-$, l'expression $e^{At}x$ appartient au cône si et seulement si

$$\begin{aligned}
& \sum_{i=1}^{p} {w_i}^t \left(e^{At}x\right) - \left|{w_j}^t \left(e^{At}x\right)\right| \geq 0, \\
\Leftrightarrow \quad & \sum_{i=1}^{p} e^{\lambda_i t} {w_i}^t x - \left|e^{\lambda_j t} {w_j}^t x\right| \geq 0, \\
\Leftrightarrow \quad & e^{\lambda_p t}\left(\sum_{i=1}^{p} e^{(\lambda_i - \lambda_p)t} {w_i}^t x - e^{(Re(\lambda_j) - \lambda_p)t}\left|{w_j}^t x\right|\right) \geq 0,
\end{aligned}$$

Or par hypothèse, les valeurs propres sont classées par ordre croissant en fonction de leur partie réelle, d'où $Re(\lambda_i - \lambda_p) \geq 0$ pour tout $i = 1, \ldots, p$ et $Re(\lambda_j - \lambda_p) \leq 0$ pour tout $j > p$. Ceci implique que

$$\begin{aligned}
e^{\lambda_p t}\left(\sum_{i=1}^{p} e^{(\lambda_i - \lambda_p)t} {w_i}^t x - e^{(Re(\lambda_j) - \lambda_p)t}\left|{w_j}^t x\right|\right) &\geq e^{\lambda_p t}\left(\sum_{i=1}^{p} {w_i}^t x - \left|{w_j}^t x\right|\right), \\
&\geq 0,
\end{aligned}$$

puisque $x \in \mathcal{K}^-$ par hypothèse. □

Que devient le cône si on l'adapte pour les systèmes non-linéaires ? Supposons dans un premier temps que les p premières valeurs propres soient réelles, nous pourrions alors définir le cône par

$$\mathcal{K}^-(x) = \left\{\delta x \in T_x\mathcal{X} \mid \sum_{i=1}^{p} \partial\phi_{\lambda_i}(x)\,\delta x - \left|\partial\phi_{\lambda_j}(x)\,\delta x\right| \geq 0,\ \forall j \geq p+1\right\} \tag{4.25}$$

Ainsi, si nous considérons ce cône ainsi que son inverse, c'est-à-dire $-\mathcal{K}^-(x)$, nous obtiendons le cône dans lequel le système est p-dominant.

Conclusion et perspectives

Dans ce mémoire, nous avons tenté d'étudier du mieux que nous pouvons les systèmes dynamiques non-linéaires au moyen de différentes approches : la première consiste à construire un prédicteur linéaire, tandis que la deuxième méthode utilise l'approche différentielle pour déterminer la dimension du comportement assymptotique.

Ainsi, la première partie de ce mémoire aborde l'outil essentiel de ce travail, à savoir l'opérateur de Koopman. Nous l'avons défini dans les cas discret et continu, et nous en avons tiré sa propriété principale, c'est-à-dire la linéarité. Toutefois, cet opérateur admet un désavantage : sa dimension. Puisqu'il est de dimension infinie, il a été nécessaire de l'approximer de manière finie via la méthode EDMD. Initialement défini pour un système fermé, il a fallu le généraliser aux systèmes ouverts, ainsi qu'à la méthode d'approximation finie. Le deuxième chapitre a abordé la prédiction linéaire de systèmes dynamiques non-linéaires. Cette méthode est couramment utilisées notamment dans la commande prédictive pour pouvoir contrôler un système non-linéaire. Ce cas a d'ailleurs été brièvement étudié dans la dernière section de ce chapitre. Puisque nous approximons linéairement la trajectoire, nous commettons nécessairement une erreur. Cette erreur est l'objet du troisème chapitre. En effet, jusqu'à présent, aucun réel résulat n'a été établi sur l'erreur que nous commetons en utilisant l'opérateur de Koopman. Nous avons donc fourni une méthode théorique pour établir cette erreur, et nous l'avons également analysée en fonction de divers paramètres dans les deuxième et troisème section de ce chapitre. Finalement, le quatrième et dernier chapitre de ce mémoire se concentre sur la théorie de la p-dominance. Celle-ci se base sur la manière la plus instinctive d'obtenir un système linéaire à partir d'un système non-linéaire, à savoir la linéarisation. En effet, tous les résultats de cette théorie s'obtiennent par approche différentielle. L'objectif de ce chapitre a été de généraliser le résultat précédemment établi par Alexandre Mauroy et Fulvio Forni à propos du lien qui relie positivité différentielle et l'opérateur de Koopman. Ainsi, nous avons pu généraliser le cône initialement défini pour la positivité différentielle aux systèmes p-dominants.

Ce travail pourrait évidemment être poursuivi longuement. En effet, la borne de l'erreur que nous avons décrite pourrait être affinée et comparée avec les résultats obtenus à la deuxième section du chapitre 3. De plus, le lien entre p-dominance et l'opérateur de Koopman pourrait être largement étendu.

Bibliographie

[1] ABRAHAM I., DE LA TORRE G. & MURPHEY T.D., *Model-Based Control Using Koopman Operators*, Robotics : Science and Systems Proceedings, 2017

[2] ANSARI A. R. & MURPHEY T. D., *Sequential action control : Closedform optimal control for nonlinear and nonsmooth systems*, IEEE Transactions on Robotics, vol. 32, num. 5, pp. 1196 – 1214, 2016

[3] ANTONOPOULOU D. C., *A note on the one-dimensional L_2-projection error of smooth functions and applications to space-time finite element approximation*

[4] FORNI F. & SEPULCHRE R., *Differentially positive systems*, IEEE Transactions on Automatic Control, vol. 61, num. 2, pp. 346 - 359, 2016

[5] FORNI F. & SEPULCHRE R., *A dissipativity theorem for p-dominant systems*, 56th IEEE Conference on Decision and Control, pp. 3467 - 3472, 2017

[6] FORNI F. & SEPULCHRE R., *Differential dissipativity theory for dominance analysis*, IEEE Transaction on Automatic Control, pp. 99 - 109 , 2017

[7] HAUSER J.,*A projection operator approach to the optimization of trajectory functionals*, IFAC Proceedings Volumes, vol. 35, num. 1, pp. 377 – 382, 2002

[8] KAISER E., KUTZ J.N., BRUNTON S.L., *Data-driven discovery of Koopman eigenfunctions for control*, American Physical Society, Division of Fluid Dynamics, 2017

[9] KORDA M. & MEZIĆ I., *Linear predictors for nonlinear dynamical systems : Koopman operator meets model predictive control*, Automatica, vol. 93, pp. 149 - 160, 2016

[10] MAUROY A., FORNI F. & SEPULCHRE R., *An operator-theoretic approach to differential positivity*, 54th IEEE Conference on Decision and Control, pp. 7028 - 7033, 2015

[11] MAUROY A., *Méthodes avancées pour les systèmes non linéaires*, UNamur, année académique 2017-2018

[12] PEITZ S. & KLUS S., *Koopman operator-based model reduction for switched-system control of PDEs*, SIAM Journal on Control and Optimization, 2017

[13] PROCTOR J.L., BRUNTON S.L. & KUTZ J.N., *Generalizing Koopman Theory to Allow for Inputs and Control*, SIAM Journal on Applied Dynamical Systems, vol. 17, num. 1, pp. 909 - 930, 2018

[14] WIKIPEDIA, *La suite logistique*, https ://fr.wikipedia.org/wiki/Suite_logistique, 16 mai 2019

[15] WIKIPEDIA, *Théorie des bifurcations*, https ://fr.wikipedia.org/wiki/Th%C3%A9orie_des_bifurcations, 16 mai 2019

[16] WILLIAMS M.O., KEVREKIDIS I.G. & ROWLEY C.W., *A Data–Driven Approximation of the Koopman Operator : Extending Dynamic Mode Decomposition*, Journal of Nonlinear Science, vol. 25, num. 6, pp. 1307 - 1346, Springer Science+Business Media, New York, 2015

[17] WILLIAMS M.O., HEMATI M.S. , DAWSON S.T.M. , KEVREKIDIS I.G. & ROWLEY C.W., *Extending Data-Driven Koopman Analysis to Actuated Systems*, IFAC, vol. 49, num. 18, pp. 704 - 709, 2016

[18] WINKIN J. & MAUROY A., *Systèmes complexes commandés*, UNamur, année académique 2017-2018

Annexe A

Codes Matlab

Nous reprenons dans cette annexe les codes principaux que nous avons utilisés pour générer les résultats de ce mémoire. Autrement dit, la première section reprend la fonction de lift utilisées dans la référence [9] tandis que la deuxième section reprend la fonction de lift pour les polynômes. Ensuite, la troisème section reprend le code principal de la méthode de Korda [9] développée dans le deuxième chapitre. Finalement, la dernière section reprend lafonction du système dynamique de Van der Pol que nous avons utilisée dans la fonction principale.

A.1 Fonction de lift : base radiale

```
function [y,dy] = phi(x,c,param)
% Precondition :    x, vecteur etat dimension n
%                   c, centres des fonctions de lift
%                   param.N, nombre de fonctions de lift
% Postcondition : y vecteur lifte de dimension param.N

    N = param.N;

    % Premiere partie :  fonction identite
    y(1) = x(1);
    y(2) = x(2);

    % Deuxieme partie : fonctions de lift
    % thin plate radial basis function

    for i = 3:N
        % radial basis function
        r = norm(x-c(i-2,:)');
        y(i) = r*r*log(r);
    end

end
```

A.2 Fonction de lift : base polynômiale

```
function[y] = phi_pol(x)
% Precondition : x, vecteur etat dimension n
%
% Postcondition : y vecteur lifte de dimension N

    % Premiere partie : fonction identite
    y(1) = x(1);
    y(2) = x(2);

    % Deuxieme partie : fonctions de lift
    % polynomes

    n = 3;
    for i = 1:1:10
        for j = 1:1:10
            y(n) = (x(1)^i)*(x(2)^j);
            n = n+1;
        end
    end

end
```

A.3 Code principal

```
clear all
clc

% Choisir le parametre qu'on souhaite faire varier:
% pas = [0.1,0.05,0.01,0.005,0.001];
% nbre_traj = [10 25 50 100 200];
% N = [12, 27, 52, 102, 202, 302];

% adapter size() avec le parametre choisi:
figure
for nb = 1:1:size(N)
    error_loc1 = [];
    error_loc2 = [];

    for b = 1:1:20
        n = 2; % dimension du systeme
        m = 1; % dimension de l'entree
        pas = 0.01; % pas d'integration
        nbre_traj = 100; % a decocher si nbre_traj est le
            parametre etudie
```

```
20          % a decocher quand on souhaite garder le meme nombre de
                donnees
21          % t_end = 1000*pas;
22          t_end = 10;
23          % a decocher si N n est pas le parametre etudie
24          % N = 102;
25          c = [];
26          for i = 3:N
27              c(i-2,:) = [unifrnd(-1,1) ; unifrnd(-1,1)];
28          end
29
30          for traj = 1:1:nbre_traj
31              K = size([0:pas:t_end],2)-1;
32              % condition initiale aleatoire
33              % Exemple article Korda : loi uniforme sur [-1,1]
34              CI = [unifrnd(-1,1) ; unifrnd(-1,1)];
35              u = ltePRBS(randi([0 1000000],1,1),K+1,'signed')';
36              param.u = u;
37              param.pas = pas;
38              fdin = @(t,x) VanDerPol(t,x,param);
39
40              % Creation des donnees : integration
41              [t,yOut] = ode45(fdin,0:pas:t_end,CI);
42              X = yOut(1:K,:)';
43              Y = yOut(2:K+1,:)';
44
45              % Fonctions de lift :
46              param2.N = N;
47              Xlift = [];
48              Ylift = [];
49              for i = 1:size(X,2)
50                  Xlift(:,i) = phi(X(:,i),c,param2);
                                                    % Korda
51                  Ylift(:,i) = phi(Y(:,i),c,param2);
                                                    % Korda
52
53                  % a decocher si on utilise la base polynomiale:
54                  % Xlift(:,i) = phi_pol(X(:,i));
                                                    % Korda
55                  % Ylift(:,i) = phi_pol(Y(:,i));
                                                    % Korda
56              end
57              N = size(Xlift,1);              % dimension espace
                    lift
58              U = u(1:size(u,2)-1);
59
60              if traj > 1
61                  % Korda
```

```
62                  Xlift_tot = [Xlift_tot, Xlift];
63                  Ylift_tot = [Ylift_tot, Ylift];
64                  Utot = [Utot,U];
65                  Xtot = [Xtot, X];
66                  Ktot = Ktot+K;
67              else
68                  % Korda
69                  Xlift_tot = Xlift;
70                  Ylift_tot = Ylift;
71                  Utot = U;
72                  Xtot = X;
73                  Ktot = K;
74              end
75          end
76          % KORDA
77          % Construction estimateur :
78          AB = Ylift_tot/([Xlift_tot; Utot]);
79          A = AB(1:N,1:N); % A matrice carree N x N
80          B = AB(1:N,N+1:N+m);% B matrice rectangulaire X x m
81          C = Xtot/Xlift_tot;
82
83          % Example :
84          x_in = [-0.1;-0.5]; % condition initiale
85
86          % choix du controle dans la fonction VanDerPol2
87          t_end_est = 3;
88          param.pas = pas;
89          param.freq = 1/0.3;
90          param.amp = 1;
91          fdin = @(t,x) VanDerPol2(t,x,param);
92          [t,xOut] = ode45(fdin,0:pas:t_end_est,x_in);
93
94          % Estimation :
95          z = [];
96          z(:,1) = phi(x_in,c,param2); % condition initiale ->
                lifter condition initiale de x
97          % a decocher si on utilise les fonctions polynomiales :
98          % z(:,1) = phi_pol(x_in); % condition initiale -> lifter
                condition initiale de x
99          i = 1;
100         amp = 1;
101         freq = 1/0.3;
102         for t = 0:pas:t_end_est-pas
103             z(:,i+1) = A*z(:,i) + B*amp*square(2*pi*freq*t);
104             x_hat(:,i) = C*z(:,i+1);
105             i = i+1;
106         end
107         error_glob1(b) = sqrt(mean((x_hat(1,:)'-xOut(1:size(
```

```
            x_hat,2),1)).^2));
108         error_glob2(b) = sqrt(mean((x_hat(2,:)'-xOut(1:size(
                x_hat,2),2)).^2));
109         error_loc1(b,:) = abs(x_hat(1,:)'-xOut(1:size(x_hat,2)
                ,1));
110         error_loc2(b,:) = abs(x_hat(2,:)'-xOut(1:size(x_hat,2)
                ,2));
111     end
112
113     % On plot l'erreur globale en fonction de chaque parametre
114     moyenne1(nb) = mean(error_glob1);
115     ecart_type1(nb) = std(error_glob1);
116
117     subplot(4,1,1)
118     plot(N(nb)*ones([1  20]),error_glob1,'*')
119     hold on
120
121     moyenne2(nb) = mean(error_glob2);
122     ecart_type2(nb) = std(error_glob2);
123
124     subplot(4,1,3)
125     plot(N(nb)*ones([1  20]),error_glob2,'*')
126     hold on
127
128     % on plot en chaque point l'erreur moyenne en fonction du
            temps
129     moy_1 = [];
130     moy_2 = [];
131
132     for i = 1:size(x_hat,2)
133         moy_1(i) = mean(error_loc1(:,i));
134         moy_2(i) = mean(error_loc2(:,i));
135     end
136
137     subplot(4,1,2)
138     semilogy(0:pas:t_end_est-pas, moy_1)
139     hold on
140     subplot(4,1,4)
141     semilogy(0:pas:t_end_est-pas, moy_2)
142     hold on
143 end
```

A.4 Fonction du système de Van der Pol

```
1 function[dy] = VanDerPol(t,x,param)
2 % Precondition :     x, vecteur etat dimension n
3 %                    t, le temps
4 %                    param, les parametres necessaires
```

```
5  % Postcondition :   dy contient la dynamique de VanDerPol au
       temps t et au
6  %                   point x
7      u = param.u;
8      pas = param.pas;
9      j = ceil((t+1e-6)/pas);
10     dy = [ 2*x(2); -0.8*x(1) + 2*x(2) - 10*x(2)*x(1).^2 + u(j)];
11
12 end
```