



50376
198
291

THESE

Nouveau Régime

Présentée à

L'UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE

Pour obtenir le titre de

DOCTEUR en INFORMATIQUE

par

Christophe Guyon



Calcul Symbolique

pour la Planification de Trajectoire
des Systèmes Dynamiques Nilpotents

Thèse soutenue le 7 décembre 1995, devant la Commission d'Examen
(version provisoire)

Président	: Jean-Marc GEIB	Université de LILLE I
Rapporteurs	: Gérard DUCHAMP	Université de ROUEN
	: Matthias KAWSKI	Rutger's University, ARIZONA
Examineurs	: Geneviève DAUPHIN	Ecole Centrale de LILLE
	: Gérard JACOB	Université de LILLE I
	: Jean-Paul LAUMOND	L.A.A.S., TOULOUSE

Table des matières

1	Introduction	5
2	Notions préliminaires de géométrie différentielle	12
2.1	Champs de vecteurs	12
2.1.1	Exemple: une pièce de monnaie	12
2.1.2	La variété des configurations M	13
2.1.3	L'espace tangent $TM _x$ et le fibré tangent TM	14
2.1.4	Les champs de vecteurs X sur la variété	15
2.1.5	Le crochet de Lie de 2 champs de vecteurs	15
2.1.6	Algèbre de Lie d'une famille de champs de vecteurs	16
2.2	Systèmes analytiques: géométrie et intégration	17
2.2.1	Systèmes analytiques affines en la commande	17
2.2.2	Le flot d'un champ de vecteurs	17
2.2.3	Algèbre de Lie de commande	19
2.2.4	Les distributions Δ	20
3	Aspects combinatoires	22
3.1	Le codage par le monoïde libre	22
3.1.1	Des mots pour le codage	22
3.1.2	Ordre lexicographique	22
3.1.3	Algèbres non commutatives et de mélange	23
3.1.4	Série formelle S	23
3.1.5	Transformée d'évaluation \mathcal{E} . Série de Chen	23

3.1.6	Série génératrice d'un système dynamique, \mathcal{Y} , série double . . .	24
3.2	Algèbre de Lie libre et base de Lyndon	25
3.2.1	Les mots de Lyndon	25
3.2.2	Crochetage des mots de Lyndon	26
3.2.3	Algorithme de génération des mots de Lyndon	26
3.2.4	Mots de Širšov	27
3.2.5	L'algèbre de Lie libre	27
3.2.6	Normalisation des polynômes de Lie	27
3.3	Algèbre enveloppante	28
3.3.1	L'algèbre enveloppante, La base de Poincaré–Birkoff–Witt . . .	28
3.3.2	La base duale: les S_l	28
3.4	Opérateur de transport \mathcal{H}	29
3.4.1	La factorisation de la série double	29
3.4.2	Définition	30
4	Systèmes nilpotents	31
4.1	Définition. Coordonnées exponentielles	31
4.2	Nilpotentisation	32
4.3	Système nilpotent libre	33
4.3.1	Algèbre de Lie nilpotente libre	33
4.3.2	Groupe et système nilpotents libres	33
4.3.3	Réalisation minimale d'un système nilpotent libre	34
4.3.4	Forme canonique des systèmes nilpotents libres	35
4.3.5	Calculs dans le groupe nilpotent libre	36
4.4	Systèmes chaînés	36
4.4.1	Propriété	37
4.4.2	Mise sous forme chaînée	37
4.5	Forme triangulaire des systèmes nilpotents	38
4.6	Triangularité et corps différentiels	38
4.6.1	Platitude. Triangularité. Extensions de corps différentielles . .	38

4.6.2	Exemple du "Plate-Ball"	39
5	Planification de trajectoire d'un système nilpotent	41
5.1	Mise en équation	41
5.2	Coordonnées exponentielles de la cible	42
5.2.1	Inversion du système étendu	42
5.2.2	Méthode géométrique	43
5.2.3	Intégration des flots	44
6	Identification de l'entrée	48
6.1	Coordonnées exponentielles de l'entrée	48
6.1.1	Cas des entrées constantes par morceaux	48
6.1.2	Cas des entrées polynomiales	54
6.1.3	Cas des entrées difficiles à intégrer formellement	54
6.2	Le système de 5 équations	55
6.2.1	Méthode de Gröbner	55
6.2.2	Méthode d'identification	66
6.2.3	Solution, simulations, pour la pièce qui roule	74
6.3	Systèmes avec dérive	77
6.4	Systèmes chaînés	79
6.4.1	Système chaîné de 4 équations	79
6.4.2	N équations	81
6.5	Jusqu'à 19 équations	81
6.5.1	Remarques préliminaires	81
6.5.2	Une méthode en 3 mouvements	84
7	Identification de la sortie plate d'un sous système	87
7.1	Méthode d'interpolation	87
7.2	Méthode des sous systèmes plats	89
8	Identification dans le groupe de Lie	92

8.1	Intégrer un champ de vecteur à la fois	92
8.2	Cas particuliers de la loi produit	93
8.3	Algorithme.	94
8.3.1	Solution générique	96
8.3.2	Solutions particulières	97
9	Conclusion	101
A	Exemple de système plat	103
A.1	Définition	104
A.2	Exemple: la pièce est plate	104
A.3	La planification de trajectoire:	104
B	Programmes Axiom	106
B.1	Les S_l	107
B.2	Evaluation des $S_l: \mathcal{E}(S_l)$	114
C	Calculs dans le groupe nilpotent libre	117

Chapitre 1

Introduction

La planification de trajectoire est l'étude du lien entre une commande et la trajectoire qu'elle produit dans l'espace d'état (ou des configurations) du système dynamique étudié.

Ce problème trouve ses applications principalement en robotique. Deux sous-problèmes se dégagent et ont fait l'objet de nombreux travaux. On peut chercher à *se rapprocher d'une trajectoire théorique idéale* initialement "dessinée" (Lafférière et Sussmann [29, 47], Laumond et Soarès [45]). On peut aussi chercher à précalculer une commande qui produit une trajectoire *conduisant le système, en un temps fixé, sur un état désigné comme cible* (Fliess [11], Laumond [30, 31]). Toute solution de chacun de ces deux sous-problèmes peut donner des éléments de réponse (au moins partiels) au problème de l'évitement d'obstacles. Soit en approchant d'une courbe qui les évite, soit en décomposant le trajet à l'aide de cibles intermédiaires successives. C'est pourquoi ce problème a été largement étudié ces dernières années tant par des automaticiens, des informaticiens que des mathématiciens.

Dans le cas de systèmes comportant autant de commandes que de variables d'état, appelé *systèmes holonomes* on peut inverser un système d'équations différentielles et obtenir ainsi une réponse au problème. Dans le cas *non holonome* le problème est plus dur (Bellaïche, Laumond et Chyba [1], Lafférière et Sussmann [29], Fliess [11], Sastry et Liu [42]). Il peut s'agir par exemple de calculer une commande (les tours de volant et la vitesse de rotation des roues) permettant de garer automatiquement une voiture. La difficulté du créneau vient en effet de la non holonomie de ce système : une voiture ne peut pas se déplacer latéralement. On peut aussi vouloir mettre en ordre les faces colorées du Rubik's Cube, piloter en attitude un satellite avec seulement deux commandes (ou tenter de jouer à Block Out (Tetris en 3D) avec deux touches de rotation seulement!).

Le présent travail est consacré à la recherche, par le calcul exact, de solutions pour le second sous-problème : le calcul, d'une commande *conduisant exactement sur une cible*, ceci dans le cadre général de *systèmes non-holonomes*. Par ailleurs on se

limitera au cas des *systèmes nilpotents* et l'on cherchera la commande dans une classe particulièrement simple: *commandes constantes par morceaux*, ou *commandes polynomiales*.

On utilisera abondamment la combinatoire (pour la préparation des calculs) et le calcul formel (pour la résolution de systèmes algébriques) avec paramètres de grande dimension.

L'étude de la combinatoire des algèbres de Lie libres (Campbell [3], Dynkin [6, 7], Haussdorff [15], Klyachko [26], Lyndon [32], Ree [40], Širšov [44], Schützenberger [43], Wever [50] et Witt [51]) est en effet préalable à toute implantation efficace. Les bases de cette théorie, initiée par Viennot [49], font l'objet du livre de Reutenauer [41]. Les résultats plus récents utilisent les fonctions symétriques pour un calcul rapide de formules "à la Baker-Hausdorff" (Krob, Leclerc, Novelli, Thibon [28]) et la construction de famille à un paramètre de projecteurs de Lie qui interpolent entre les idempotents de Dynkin, Solomon et Klyachko (Krob, Duchamp, Leclerc, Thibon [27])

Différentes approches sont naturelles pour aborder ce type de problèmes, dont certaines nous serviront de préparations, on en présente ici un échantillon représentant les courants qui nous sont apparus représentatifs.

Les méthodes de *linéarisation autour d'un point de fonctionnement* par bouclage statique, ou dynamique, ne permettent pas de résoudre le problème en général pour les systèmes non linéaires (d'Andréa-Novel [5]). Elles ne fournissent en effet qu'une approximation locale des trajectoires.

L'algèbre différentielle (Fliess [9], Fliess et Glad [10] Fliess, Lévine, Martin, Rouchon [11]) permet de résoudre élégamment et simultanément les deux sous-problèmes cités pour certains systèmes dits *systèmes plats*. On cherche tout d'abord le jeu de polynômes différentiels appelé *sortie plate* à partir duquel (et de ses dérivées) toutes les grandeurs de la dynamique se retrouve algébriquement. On assigne alors à la *sortie plate* toute trajectoire "assez lisse" menant les variables de configuration du système à la position finale désirée. De nombreux systèmes issus de la mécanique sont plats. Par contre, le calcul d'une sortie plate (ou "linéarisante") ne peut généralement être découverte qu'après une étude spécifique approfondie du système. En outre cette méthode ne permet pas, lorsqu'on revient à la description pour les commandes initiales, de calculer par exemple des commandes constantes par morceaux, ou polynomiales.

L'approche par le *calcul extérieur* (Gardner et Shadwick [12]), issu des travaux d'Elie Cartan, permet de s'extraire du calcul en coordonnées locales, et de fournir des informations structurelles intrinsèques sur la géométrie du système. Grâce à l'étude des formes de Pfaff, et à un critère dû à Goursat, Murray, Sastry et Tilbury [36, 35, 48] ont pu mettre un certain nombre de systèmes dynamiques issus de la mécanique sous *forme chaînée*. Ceux-ci sont plats, et on peut donc les commander aussi en assignant une trajectoire lisse à leur sortie-linéarisante. A noter également que la forme de Goursat est une condition suffisante, mais non nécessaire de mise sous forme chaînée.

L'approche par la combinatoire des champs de vecteurs a permis à Laumond et Souères [45] de *calculer complètement les trajectoires optimales* d'un petit robot (le robot Hilare) soumis à des bornes sur les angles de pivotement. Cette approche reste très spécifique au système étudié. Elle a cependant été l'une des premières à mettre en évidence la difficulté de la commande des systèmes non-holonomes.

La première classe raisonnable de systèmes dynamiques pour laquelle on puisse espérer le calcul de commandes exactes qui inclue la classe des systèmes chaînés (ou multi-chaînés) est celle des systèmes dont l'*algèbre de Lie est nilpotente* (Hermès, Lundell et Sullivan [18], Hermès [16, 17], Kawski [25]). En effet, dans ce cas les "coordonnées canoniques exponentielles" permettent de ramener les calculs dans le groupe de Lie du système à des équations algébriques (Jacob [21, 22], Oussous et Petitot [38], Guyon et Jacob [13]). Nous ne disposons pas non plus de caractérisation algébrique ou structurelle des systèmes dynamiques nilpotentisables par bouclage. Cependant des conditions suffisantes ont pu être données pour qu'un système dynamique puisse être arbitrairement approché par des systèmes nilpotents (Hermès [17]). La classe des systèmes nilpotents est donc une classe non triviale de systèmes pour lesquels le problème mérite d'être résolu.

Formellement, le problème est le suivant :

- on se donne un système (Σ) analytique et affine en la commande, dont la dynamique est donnée par :

$$(\Sigma) \quad \frac{d}{dt}x \doteq \frac{d}{dt} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = u_1 X_1(x) + u_2 X_2(x) + \cdots + u_m X_m(x) \\ x(0) = x_0$$

où x est l'état du système, $u = (u_1, u_2, \dots, u_m)$ est l'entrée, ou commande, et X_1 et X_2 sont deux champs de vecteurs analytiques.

- On se donne un "élément-cible" q de l'espace d'état (une configuration à atteindre).

On se propose alors de calculer une entrée $u = (u_1, u_2)^T$ menant le système (Σ) au point d'arrivée q .

Dans toute sa généralité ce problème est ouvert à l'heure actuelle. Par ailleurs il est à noter que nous nous sommes restreints à 2 entrées.

Les hypothèses minimales qui semblent requises pour espérer résoudre le problème sont les suivantes :

- les champs de vecteur sont complets : c'est-à-dire définis partout dans une com-

posante connexe contenant l'état initial x_0 et l'état cible q

- le système (Σ) est complètement commandable. Ou encore, ce qui revient au même, l'algèbre de Lie de commande de (Σ) vérifie la condition de rang (Lie Algebra Rank Condition) :

$$\text{Lie}(X_1, \dots, X_m)|_x = \mathbb{R}^n, \quad \text{en tout point } x \text{ de l'espace d'état}$$

En d'autres termes, l'algèbre de Lie de commande est de dimension maximale n , égale à la dimension de l'espace d'état. Ceci, d'après le théorème de Chow [4], nous assure l'existence d'au moins une commande exacte de x_0 à q .

Il s'agit en conclusion de rendre le théorème de Chow effectif dans le cas intéressant car non trivial d'un système non-holonyme.

Pour toutes les notions et résultats généraux concernant les variétés différentielles d'une part et les systèmes dynamiques de l'autre, nous renverrons aux références de bases suivantes : Olver [37], et Isidori [20].

Le cas des systèmes nilpotents se prête particulièrement bien au calcul algébrique. Ceci peut être mis en évidence comme suit : dans le cas d'un système nilpotent vérifiant la condition du rang (L.A.R.C.), en adoptant pour le calcul de l'état atteint par le système, initialisé en un état x_0 , la forme d'un développement en produit d'exponentielles :

$$x(t) = \lambda_{x_0} \circ e^{\xi_1(t)Y_1} e^{\xi_2(t)Y_2} \dots e^{\xi_d(t)Y_d} \circ Id_x$$

Dans ce développement, introduit par Hector Sussmann [46] (voir aussi Huillet, Moin, Salut [19], et Reutenauer [41] pour la combinatoire associée),

- les d champs de vecteurs Y_i sont certains crochets de Lie itérés des champs de vecteurs de définition du système, et peuvent être calculés à l'aide d'une base de l'algèbre de Lie libre (base de Hall [14], ou de Lyndon [32], ou de Širšov [44]) qui forment en tout point x un système de générateurs de l'algèbre de Lie de commande.
- Les d fonctions $\xi_i(t)$ forment alors un système de coordonnées locales sur la variété au voisinage de x_0 .

(Cette formule peut aussi être présentée comme la généralisation du calcul du flot d'un champ de vecteurs dans le cas d'un système commandé indépendamment par plusieurs champs de vecteurs).

Dès lors, la connaissance du résultat suivant dû à M. Kawski [24] montre que le problème se ramène à un problème algébrique dans le cas des systèmes nilpotents. Son résultat peut alors s'interpréter comme suit : les coordonnées locales exponentielles

$\xi_i(t)$ du point courant $x(t)$, d'un système guidé par une commande u , vérifient un système d'équations différentielles triangulaire polynomial en les ξ_i et linéaire en la commande u , c'est-à-dire de la forme :

$$(\Sigma N) \quad \begin{cases} \dot{\xi}_1 = f_1(u) \\ \dot{\xi}_2 = f_2(\xi_1, u) \\ \dot{\xi}_3 = f_3(\xi_2, \xi_1, u) \\ \vdots \\ \dot{\xi}_d = f_n(\xi_{d-1}, \dots, \xi_1, u) \end{cases}$$

Par ailleurs les équations de (ΣN) deviennent beaucoup plus simples si l'on change les coordonnées locales en ξ_q , pour mettre le système sous une forme nilpotente libre effectivement calculable. Par exemple à l'ordre 4, un tel système nilpotent libre :

$$(\Sigma NL(4)) \quad \begin{cases} \dot{\xi}_1 = u_1 \\ \dot{\xi}_2 = u_2 \\ \dot{\xi}_{21} = \xi_2 u_1 \\ \dot{\xi}_{211} = \xi_{21} u_1 \\ \dot{\xi}_{221} = \frac{1}{2} \xi_2^2 u_1 \end{cases}$$

(déjà étudié au tournant du siècle par E. Cartan.)

Une possibilité est alors de calculer les coordonnées locales exponentielles dans la variété de la cible q :

$$q = \lambda_{x_0} \circ \left(\prod_{l \in \mathcal{B}} e^{q_l Y_l} \right) \circ Id_x$$

où \mathcal{B} est une base de l'algèbre nilpotente libre, puis de résoudre le problème de la planification non plus pour le système d'origine mais pour (ΣNL) .

Dans le présent mémoire, nous avons systématiquement testé et comparé les diverses méthodes permettant, pour un système à deux entrées, le calcul d'une commande conduisant exactement sur la cible, dans le cas où cette commande est soit constante par morceaux soit polynomiale.

Les outils informatiques utilisés et programmés pour ce travail sont :

- programmés sous Axiom : les domaines et catégories implantées par M. Petitot [38] pour le calcul dans groupe nilpotent libre, l'algèbre de Lie libre et les champs de vecteurs auxquels s'ajoutent les domaines et paquetages des mots de Lyndon, des éléments duaux des polynômes de la base de Poincaré-Birkoff-Witt pour des mots de Lyndon (les " S_i "), de la transformée d'évaluation de ces derniers ($\mathcal{E}(S_i)$), de calcul de formes quadratiques
- utilisés sous Axiom : les bases de Gröbner, la résolution numérique

- programmés sous Maple : les algèbres de Lie et les champs de vecteurs par M. Bigotte [2], le calcul et la résolution des équations en mode interprété.
- utilisés sous Maple : les bases de Gröbner, la résolution numérique

Les objectifs sont de résoudre le problème par une méthode qui permet de satisfaire au plus grand nombre des critères suivants :

- la cible est n'importe quelle configuration *possible*
- la méthode de calcul est systématique, c'est à dire indépendante du système particulier étudié
- la méthode est symbolique, *i.e.* elle considère les coordonnées de la cible comme des paramètres (ceci pourrait permettre par exemple le calcul embarqué)
- elle peut prendre en compte des variables libres supplémentaires qui laisse un ou plusieurs degrés de liberté à la solution (afin d'ouvrir une étude de l'optimisation ou de la stabilisation du système)
- elle donne une solution exacte

Les apports de ce travail se résument ainsi :

- *le lien avec les systèmes plats* est fait sur le terrain de l'algèbre différentielle où l'on introduit la notion de *sous système plat* d'un système nilpotent. Cette notion est rendue effective dans le cas de systèmes nilpotents libres où une forme canonique est calculée qui fait apparaître clairement le sous système plat.
- *l'étude complète du système nilpotent libre à l'ordre 4* : $\Sigma NL(4)$. Ce système à 2 entrées, 5 équations différentielles (et le même nombre de variables d'états) est en terme du nombre d'équations le plus petit système nilpotent libre *non plat*, il présente dans ce cadre certaines difficultés intrinsèques typiques du problème posé. Le bilan est le suivant :
 - pour des entrées polynomiales : on calcule une solution de degré 3 dans tous les cas et l'on montre que son degré est minimal (pas de solution pour les degrés plus petits sauf pour des cas singuliers). Ceci est fait en ramenant le problème à une seule équation en une variable, de degré 5
 - pour des entrées constantes par morceaux on calcule une solution formelle complète (en tenant compte de tous les cas singuliers) en 4 paliers, dont on montre également que c'est le nombre minimal.
 - en organisant certaines propriétés remarquables de la loi produit du groupe nilpotent libre on construit une solution générique en 6 exponentielles. Ses caractéristiques sont : la simplicité des expressions et un bon comportement dynamique.

-
- enfin pour un système avec dérive sur la première entrée on extrait des suites de formes quadratiques qui permettent de connaître le domaine d'accessibilité en fonction du nombre de paliers constants de l'autre entrée
 - Pour les systèmes de plus grande dimension, en concaténant les solutions en entrées polynomiales de 3 cas particuliers on trouve –par une méthode linéaire– une solution symbolique complète pour des systèmes nilpotents ayant jusqu'à 19 équations, ceci incluant les systèmes nilpotents libres à l'ordre 6 : c'est l'ordre le plus élevé auquel nous sommes parvenus.

Le présent mémoire commence par introduire les notions de géométrie différentielle utiles aux systèmes dynamiques jusqu'au théorème de Chow (chapitre 2) et celles de combinatoire sur les mots dans les structures libres (monoïde, algèbre et groupe de Lie, algèbre enveloppante, base duale) menant à l'opérateur de transport (chapitre 3).

On présente alors les systèmes nilpotents, les systèmes nilpotents libres et leur pendant combinatoire (groupes et algèbres de Lie *nilpotentes* libres) ainsi que les autres classes de systèmes non linéaires définies lors des recherches sur la planification de trajectoire à savoir les systèmes chaînés, sous forme triangulaire et les systèmes plats, en faisant apparaître leurs relations mutuelles (chapitre 4).

Vient alors la mise en équation du problème permettant d'aboutir aux équations algébriques (chapitre 5). Les différentes méthodes pour calculer les coordonnées exponentielles d'un point de la variété y sont développées.

Les 3 principales approches pour la résolution de la planification de trajectoire d'un système nilpotent libre peuvent alors être présentée. La première consiste à intégrer le système nilpotent libre pour des entrées constantes par morceaux ou polynomiales et à identifier les coefficients (chapitre 6). La deuxième utilise la notion de platitude d'un système. Pour un système sous forme canonique le sous système plat forme un système chaîné, menant à des équations linéaires. On commence alors plutôt la résolution par ce sous système comme pour un système *entièrement plat*, puis on tente la résolution des équations restantes (chapitre 7). La dernière approche consiste à travailler directement sur le groupe nilpotent libre en composant des flots sur un champ de vecteur à la fois (chapitre 8).

Chapitre 2

Notions préliminaires de géométrie différentielle

Pour les notions présentées dans ce chapitre, on peut se référer aux ouvrages généraux de Olver [37] et Isidori [20]

2.1 Champs de vecteurs

2.1.1 Exemple : une pièce de monnaie

Nous considérons le petit système dynamique introduit par Murray et Sastry [48] sous le nom:

“The rolling penny”

Il s’agit d’une pièce de monnaie de rayon R , roulant sur sa tranche, et dont on a un point de la circonférence a été marqué.

L’état (ou la configuration) de “la pièce” (voir figure 2.1) a un instant donné est repéré par 4 variables d’état :

- la position (x_1, x_2) du point de contact avec le plan,
- l’angle x_4 que fait la position marquée avec la verticale
- l’angle x_3 de pivotement (autour de la verticale) repéré par une direction horizontale fixée.

On dispose de 2 commandes qui sont:

- la vitesse de roulement (sans glissement de la pièce sur le plan)

- la vitesse de pivotement (autour d'un axe vertical passant par le centre de la pièce).

La dynamique de ce système est donc décrite, sous forme vectorielle, par les 4 équations différentielles :

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} &= \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} u_1(t) + \begin{bmatrix} R \cos(x_3) \\ R \sin(x_3) \\ 0 \\ R \end{bmatrix} u_2(t) \\ &= X_1 u_1 + X_2 u_2 \end{aligned}$$

On remarque que le membre droit est une expression *affine* par rapport à la commande $u(t)$. *Remarque : par la suite on prendra $R = 1$*

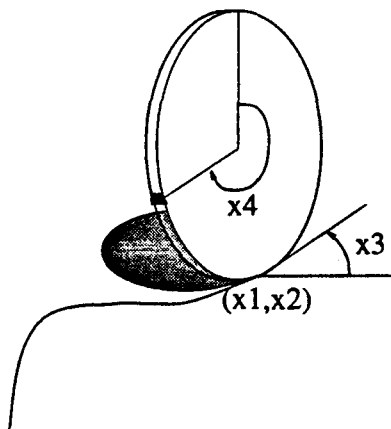


FIG. 2.1 - pièce de monnaie

2.1.2 La variété des configurations M

Une configuration de la pièce de monnaie peut donc être représentée par le vecteur $(x_1, x_2, x_3, x_4)^T$ de \mathbb{R}^4 . On peut utiliser cette représentation, ainsi dans un mouvement de la pièce la coordonnée x_4 repèrera aussi le *nombre de tours* effectués.

Mais si seule la position de la marque importe, il faudra compter x_4 modulo 2π . L'espace des configurations ne sera plus représenté par \mathbb{R}^4 que it localement.

On peut de cette façon raisonner sur des objets plus généraux que des ouverts de \mathbb{R}^n comme des cercles, des cylindres, des tores, etc ...

Plus généralement, il suffira

- de recouvrir la variété des configurations par un ensemble d'ouverts appelés cartes locales,

- de mettre en bijection chacune de ces cartes avec un ouvert de \mathbb{R}^n . On dispose ainsi localement de *fonctions coordonnées*.
- Il est bien sûr requis une "condition de recollement" entre deux cartes si leur intersection est non vide.
- Enfin, on requiert que toutes les fonctions entre ouverts de \mathbb{R}^n obtenues en composant les changements de coordonnées et leurs inverses soient analytiques.

C'est ainsi que l'on définit la structure de **variété analytique**.

Enfin une application $f : M_1 \rightarrow M_2$ est appelée *homomorphisme de variétés analytiques*, ou plus simplement fonction analytique, si composée avec les coordonnées locales (et leurs inverses), les fonctions de \mathbb{R}^n dans \mathbb{R}^n obtenues sont des fonctions analytiques réelles.

2.1.3 L'espace tangent $TM|_x$ et le fibré tangent TM

On va maintenant chercher à étudier les *directions infinitésimales* de mouvement le long de la variété M , c'est à dire à définir les vecteurs tangents à la variété. On les retrouve très naturellement en regardant de près les courbes (analytiques) tracées sur la variété.

En effet, soit une courbe dans M , passant par un point x , et définie par un paramétrage $\sigma : t \mapsto \sigma(t)$ analytique avec $\sigma(t_0) = x$, soit F l'ensemble des fonctions analytiques de M vers \mathbb{R} , on peut construire une application $W : F \rightarrow \mathbb{R}$ comme suit:

$$\begin{aligned} W : F &\rightarrow \mathbb{R} \\ f &\mapsto Wf = \left. \frac{d}{dt}(f \circ \sigma)(t) \right|_{t_0} \end{aligned}$$

- Le vecteur W , appartenant au dual de l'espace vectoriel F^* , est appelé *vecteur tangent à la courbe paramétrée σ au point x* . C'est une dérivation de l'ensemble des fonctions analytiques définies dans un voisinage de x .
- Deux courbes paramétrées σ_1 et σ_2 possédant un contact d'ordre supérieur ou égal à un définissent le même vecteur tangent.
- La fonction analytique Wf est appelée *dérivée de la fonction f dans la direction de la courbe σ* .

En tout point x , l'ensemble des vecteurs tangents, muni de sa structure d'espace vectoriel, est appelé l'*espace tangent à la variété au point x* , et noté classiquement $TM|_x$.

On pourra aussi considérer globalement tous les espaces tangents en tous les points de la variété M , en faisant la réunion disjointe de tous les $TM|_x$, on définit ainsi le *fibré tangent* TM :

$$TM = \bigcup_{x \in M} TM|_x$$

Si l'on choisit au point x une carte locale définie par les fonctions coordonnées x_1, x_2, \dots, x_n , l'espace tangent à M en x admet pour base :

$$\left. \frac{\partial}{\partial x_1}(\cdot) \right|_x, \left. \frac{\partial}{\partial x_2}(\cdot) \right|_x, \dots, \left. \frac{\partial}{\partial x_n}(\cdot) \right|_x$$

On retrouve ainsi la définition classique: un vecteur tangent est la donnée, en tout point x de M , d'un vecteur de \mathbb{R}^n

2.1.4 Les champs de vecteurs X sur la variété

Un champ de vecteurs X analytique sur M est une application qui à tout point x de M associe un vecteur $X(x) \in TM|_x$ tangent à M en ce point et qui est analytique sur M . On peut donc l'écrire, si l'on choisit au point x des coordonnées locales x_1, x_2, \dots, x_n :

$$X(x) = \vartheta_1(x) \frac{\partial}{\partial x_1} + \vartheta_2(x) \frac{\partial}{\partial x_2} + \dots + \vartheta_n(x) \frac{\partial}{\partial x_n}$$

L'ensemble des champs de vecteurs sur M forme un espace vectoriel.

Un exemple physique de champ de vecteurs souvent présenté est celui de vitesse d'un fluide (on prend alors pour variété un ouvert de \mathbb{R}^3). Le vecteur "vitesse du fluide au point (x, y, z) " définit un "champ de vecteur vitesse" $v|_{(x,y,z)}$.

On peut également donner l'exemple des champs de vecteurs de la pièce de monnaie donnés sous forme d'opérateurs différentiels et non plus vectorielle :

$$\begin{aligned} X_1 &= \frac{\partial}{\partial x_3} \\ X_2 &= \cos(x_3) \frac{\partial}{\partial x_1} + \sin(x_3) \frac{\partial}{\partial x_2} + \frac{\partial}{\partial x_4} \end{aligned}$$

2.1.5 Le crochet de Lie de 2 champs de vecteurs

On appelle *action d'un champ de vecteurs* X sur une fonction $f : M \rightarrow \mathbb{R}$, la fonction de M dans \mathbb{R} :

$$\begin{aligned} X(f)(x) &= \left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} f(\exp(\varepsilon X)x) \\ &= \sum_{i=1}^n \vartheta^i(x) \frac{\partial f}{\partial x^i}(x) \end{aligned}$$

On définit alors le crochet de Lie de deux champs de vecteurs X et Y noté $[X, Y]$ comme étant l'unique champ de vecteurs vérifiant :

$$[X, Y](f) = X(Y(f)) - Y(X(f))$$

ou encore en coordonnées locales, si :

$$X = \sum_{i=1}^m \vartheta^i(x) \frac{\partial}{\partial x^i}, \quad Y = \sum_{i=1}^m \omega^i(x) \frac{\partial}{\partial x^i}$$

alors :

$$\begin{aligned} [X, Y] &= \sum_{i=1}^n (X(\omega^i) - Y(\vartheta^i)) \frac{\partial}{\partial x^i} \\ &= \sum_{i=1}^n \sum_{k=1}^n \left(\vartheta^j \frac{\partial \omega^i}{\partial x^j} - \omega^j \frac{\partial \vartheta^i}{\partial x^j} \right) \frac{\partial}{\partial x^i} \end{aligned}$$

Une autre définition plus géométrique du crochet de Lie sera donnée plus loin (section 2.2.2) lors de la définition des flots d'un champ de vecteurs.

Les propriétés fondamentales du crochet de Lie sont :

- bilinéarité

$$\begin{cases} [cX + c'X', Y] = c[X, Y] + c'[X', Y] \\ [X, cY + c'Y'] = c[X, Y] + c'[X, Y'] \end{cases}$$

où c, c' sont des constantes

- antisymétrie

$$[X, Y] = -[Y, X]$$

- identité de Jacobi

$$[Z, [X, Y]] + [Y, [Z, X]] + [X, [Y, Z]] = 0$$

2.1.6 Algèbre de Lie d'une famille de champs de vecteurs

L'algèbre de Lie engendrée par la famille de champs de vecteurs $X = (X_1, \dots, X_m)$ tangents à la variété M pour le crochet de Lie est notée $\mathcal{L}ie(X)$.

On note aussi $\mathcal{L}ie(X)|_x$ la fibre de $\mathcal{L}ie(X)$ au dessus du point x . C'est donc un espace vectoriel de dimension au plus égal à la dimension n de M .

L'analyticité des champs de vecteurs garantit que pour tout point x de M , on peut trouver un voisinage U de x et une famille libre de champs de vecteurs qui, en tout x de U , engendrent $\mathcal{L}ie(X)|_x$:

$$\dim(\mathcal{L}ie(X)|_U) \leq n$$

L'algèbre de Lie $\mathcal{L}ie(X)$ est donc localement de dimension finie ($\leq n$). On dit qu'elle est de rang plein si cette dimension est égale à n .

2.2 Systèmes analytiques: géométrie et intégration

2.2.1 Systèmes analytiques affines en la commande

Dans tout ce travail, nous étudierons les *systèmes affines en la commande*, dont la dynamique est décrite par un système d'équations différentielles ordinaires (Σ) de la forme :

$$(\Sigma) \quad \frac{d}{dt}x = X_0(x) + u_1X_1(x) + u_2X_2(x) + \cdots + u_mX_m(x)$$

où :

- $x = (x_1, \dots, x_n)^T$ est le vecteur d'état. Il appartient à une variété analytique M , appelée aussi espace des configurations.
- $u(t) = (u_0(t), u_1(t), u_2(t), \dots, u_m(t))^T$ désigne le *vecteur d'entrée* (ou *vecteur de commande*) du système à la date t
- X_1, X_2, \dots, X_m sont m *champs de vecteurs analytiques* tangents à la variété M

L'étude de la commande des systèmes analytiques affines en la commande peut être rendue difficile, en raison de la présence du terme de dérive associé au champ de vecteur X_0 . On sera donc souvent amené à étudier les systèmes sans dérive, d'équation :

$$(\Sigma) \quad \left\{ \frac{d}{dt}x = u_1X_1(x) + u_2X_2(x) + \cdots + u_mX_m(x) \right.$$

(où la partie autonome du système a disparu.)

2.2.2 Le flot d'un champ de vecteurs

Dans le cas d'une seule entrée, le système est défini par un seul champ de vecteurs:

$$\dot{x} = u_1(t)X_1(x)$$

et son intégration revient à calculer les flots du champ de vecteur X_1 .

On appelle *courbe intégrale d'un champ de vecteur* X une courbe analytique paramétrée $x = \phi(\varepsilon)$ dont le vecteur tangent en tout point coïncide avec la valeur de X en ce point :

$$\frac{d}{d\varepsilon}\phi(\varepsilon) = X|_{\phi(\varepsilon)}$$

pour tout ε .

Théorème 2.2.1 *Toute courbe intégrale M peut être prolongée en une et une seule courbe intégrale maximale. (C'est-à-dire un et une seule courbe intégrale que l'on ne peut pas prolonger sur un intervalle plus grand pour le paramètre ε).*

Un champ de vecteur X est dit *complet* si ses courbes intégrales maximales, initialisées en tout point de M , sont définies pour toutes les valeurs du paramètre $\varepsilon \in \mathbb{R}$. En coordonnées locales où le champ de vecteur s'écrit :

$$X|_x = \vartheta_1(x) \frac{\partial}{\partial x_1} + \vartheta_2(x) \frac{\partial}{\partial x_2} + \cdots + \vartheta_n(x) \frac{\partial}{\partial x_n}$$

l'équation différentielle devient :

$$\frac{dx_i}{d\varepsilon} = \vartheta_i(x), \quad i = 1, \dots, m$$

La condition supplémentaire :

$$\phi(0) = x_0$$

garantit l'unicité de la solution.

Remarques

- Si le champ de vecteur X est nul en x_0 la courbe intégrale $\phi(\varepsilon)$ initialisée en x_0 est réduite au point x_0 .
- On ne change pas la courbe intégrale en multipliant un champ de vecteurs par une constante ou par une fonction numérique analytique $f : U \subset M \rightarrow \mathbb{R}$ ne s'annulant pas sur l'ouvert U considéré, on ne fait que changer la paramétrisation de la courbe (vitesse de parcours) en proportion.

Si X est un champ de vecteur, on note $\Psi(\varepsilon, x)$ la courbe intégrale maximale de X passant par $x \in M$ pour la valeur $\varepsilon = 0$ du paramètre. Alors Ψ est appelé le flot engendré par X .

$$\Psi(0, x) = x$$

$$\frac{d}{d\varepsilon} \Psi(\varepsilon, x) = X|_{\Psi(\varepsilon, x)}$$

$$\Psi(\delta, \Psi(\varepsilon, x)) = \Psi(\varepsilon + \delta, x)$$

où $x \in M$. On utilisera aussi la notation exponentielle des flots :

$$\exp(\varepsilon X)x \equiv \Psi(\varepsilon, x)$$

ce qui fait apparaître pour la première fois l'exponentiation qui fera l'objet de plus amples développements par la suite.

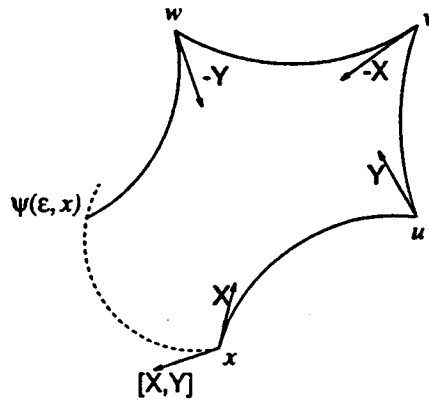


FIG. 2.2 - crochet de Lie

Les propriétés mentionnées auparavant demeurent, elles s'écrivent :

$$\exp((\delta + \varepsilon)X)x = \exp(\delta X) \exp(\varepsilon X)x$$

$$\exp(0X)x = x$$

$$\frac{d}{d\varepsilon} [\exp(\varepsilon X)x] = X|_{\exp(\varepsilon X)x}$$

On peut alors donner une définition plus géométrique du crochet de Lie de deux champs de vecteurs. On considère tout d'abord le commutateur :

$$\psi(\varepsilon, x) = \exp(\sqrt{\varepsilon}X) \exp(\sqrt{\varepsilon}Y) \exp(-\sqrt{\varepsilon}X) \exp(-\sqrt{\varepsilon}Y)x$$

qui définit une courbe analytique pour $\varepsilon \geq 0$ suffisamment petit. Le crochet de Lie $[X, Y]|_x$ est le vecteur tangent à cette courbe, au point d'arrivée $\psi(0, x) = x$ (cf. figure [2.2])

$$[X, Y]|_x = \frac{d}{d\varepsilon} \Big|_{\varepsilon^+} \psi(\varepsilon, x)$$

2.2.3 Algèbre de Lie de commande

L'algèbre de Lie engendrée par la famille de champs de vecteurs $X = (X_1, \dots, X_m)$ tangents à la variété M pour le crochet de Lie est notée $\mathcal{L}ie(X)$. Elle représente toutes les directions infinitésimales que le système peut suivre géométriquement. Certaines d'entre elles sont directement commandables : celles des champs de vecteurs de définition du système (X_1, \dots, X_m) . Dans le cas d'un système avec dérive, cependant, l'un des champs de vecteurs n'est pas accessible en commande.

Elle forme localement un espace vectoriel de dimension finie :

$$\mathcal{L}ie(X)|_x \cong \mathbb{R}^r, \quad r \leq n$$

L'analyticit  des champs de vecteurs garantit que l'on peut trouver dans un voisinage de x une famille de champs de vecteurs qui engendre cet espace vectoriel. Si les dimensions sont  gales

$$\mathcal{L}ie(X)|_x \cong \mathbb{R}^n, \quad x \in M$$

on dit que l'alg bre de Lie de commande du syst me est de rang plein. Cette condition est aussi connue des automaticiens sous le sigle : Lie Algebra Rank Condition (LARC). Lorsqu'elle est v rifi e on peut montrer que cette famille consiste en exactement n champs de vecteurs. Dans tous les cas l'extension du voisinage autour du point x consid r    une r gion plus grande conduit   augmenter le nombre de champs de vecteur de la famille g n ratrice, ceci pour compenser d' ventuelles chutes de rang.

2.2.4 Les distributions Δ

Dans le cas de plusieurs entr es:

$$\dot{x} = u_1(t)X_1(x) + \cdots + u_m(t)X_m(x)$$

on peut consid rer que le syst me est d fini par une application $g : t \mapsto g_u = u_1(t)X_1 + \cdots + u_m(t)X_m$ qui associe   tout temps t un champ de vecteurs g_u . L'ensemble des champs de vecteurs g_u , est le cadre g om trique dans lequel  voluera l' tat du syst me, et s'appelle une distribution.

Aspect g om trique:

Une *distribution analytique* sur M est une application Δ qui associe   tout point x de M le sous-espace vectoriel de $TM|_x$ engendr  par les valeurs en x d'une famille (X_1, \dots, X_d) de champs de vecteurs analytiques (d finis sur un ouvert U contenant x).

En tout x de la vari t , $\Delta|_x$ est donc un sous-espace vectoriel de l'espace tangent   M en x .

Une distribution Δ est dite *involutive* ssi elle est "stable par crochet de Lie", plus pr cis ment : si sur un ouvert U elle est engendr e par une famille (X_1, \dots, X_d) de champs de vecteurs analytiques, on impose:

$$[X_i, X_j] = h_{ij}^k X_k$$

o  les h_{ij}^k sont des fonctions analytiques.

Une variété intégrale de Δ est une sous-variété V de M vérifiant :

$$\forall x \in M, \quad \Delta|_x = TV|_x$$

On appelle sous-variété maximale de Δ toute variété intégrale V de Δ connexe (par arc), qui n'est contenue strictement dans aucune autre variété intégrale de Δ .

Une distribution Δ est dite complètement intégrable si par tout point $x \in M$ passe une sous-variété intégrale de Δ .

Théorème 2.2.2 (Frobenius, Hermann) *Une distribution Δ est complètement intégrable ssi elle est involutive et de rang constant*

Le lien avec les trajectoires

Le lien entre la complète intégrabilité et la possibilité de cheminer le long de trajectoires est alors donnée par le théorème de Chow [4]:

Théorème 2.2.3 *Pour un système sans dérive, défini sur une variété analytique connexe, dont l'algèbre de Lie de commande est de rang plein, il existe une trajectoire admissible $x(t)$ joignant 2 points quelconques de l'espace d'état.*

Il est à remarquer que ce résultat garantit l'existence d'une trajectoire admissible. Il reste alors à trouver une commande admissible qui réalise une telle trajectoire, c'est là dans un premier temps toute la difficulté de la commande des systèmes non holonomes. Par ailleurs, la commande des systèmes avec dérive dans les conditions du théorème de Chow est encore plus délicate. Le caractère affine du système (qu'on peut également voir comme une entrée constante $u_0(t) \equiv 1$ pour le champ de vecteur correspondant) entraîne, en chaque point, un mouvement du système peut ne se plus déployer que dans un "cône d'accessibilité" (et non plus dans un espace vectoriel tangent).

Chapitre 3

Aspects combinatoires

Le codage des séries génératrices des systèmes dynamiques par des séries formelles et la représentation des systèmes bilinéaires par des \mathbb{R} -automates ont établi un lien important entre la théorie des langages, la combinatoire et la théorie du contrôle. Plus récemment le lien avec l'algèbre de Lie libre et le groupe de Lie libre est apparu naturellement dans l'étude des systèmes analytiques. Les calculs d'algèbre enveloppante et de dualité conduisent aux représentations en produit d'exponentielles (ou composées de flots).

3.1 Le codage par le monoïde libre

3.1.1 Des mots pour le codage

On considère un alphabet \mathcal{Z} ordonné et fini. Le monoïde libre \mathcal{Z}^* est aussi l'ensemble des mots construits sur l'alphabet \mathcal{Z} . On note ε le mot vide et \mathcal{Z}^+ l'ensemble des mots non vides.

Dans tous les exemples, on supposera que $K = \mathbb{R}$ et que $\mathcal{Z} = \{1, 2\}$.

3.1.2 Ordre lexicographique

Nous supposons par la suite que l'on a choisi un ordre total sur les lettres de \mathcal{Z} .

L'ordre lexicographique sur \mathcal{Z}^* est défini comme suit:

$$\begin{aligned} \varepsilon &< w \quad \forall w \in \mathcal{Z}^+ \\ xu &< xv \quad \text{ssi } u < v \quad (x, u, v \in \mathcal{Z}^*) \end{aligned}$$

Ces mots serviront à coder aussi bien des opérateurs différentiels que des intégrales itérées (cf. séries formelles, section 3.1.4).

3.1.3 Algèbres non commutatives et de mélange

On notera $K \langle Z \rangle$ l'algèbre (associative) libre des polynômes non commutatifs sur Z . On définit sur l'algèbre non commutative $K \langle Z \rangle$ un produit commutatif appelé produit de mélange (en anglais "shuffle") et noté ω . Il peut être défini sur les mots puis étendu par prolongement linéaire aux polynômes et aux séries par les formules:

$$\begin{aligned} \varepsilon \omega u &= u \omega \varepsilon = u \\ x u \omega y v &= x(u \omega y v) + y(x u \omega v) \end{aligned}$$

avec $x, y \in Z$ et $u, v \in Z^*$.

Exemple : $1 \omega 22 = 122 + 212 + 221$

L'algèbre résultante est appelée *algèbre de mélange*.

3.1.4 Série formelle S

Une *série formelle* S est une application de Z^* dans un corps K :

$$\begin{aligned} S &: X^* \rightarrow K \\ w &\mapsto \langle S|w \rangle \end{aligned}$$

Une série formelle sera souvent désignée par une somme formelle infinie :

$$S = \sum_{w \in X^*} \langle S|w \rangle w$$

On notera $K \ll X \gg$ l'ensemble des séries formelles sur K

Le *produit scalaire* d'une série formelle S et d'un polynôme Q noté $\langle S|Q \rangle$ est le scalaire défini par :

$$\langle S|Q \rangle = \sum_{w \in X^*} \langle S|w \rangle \langle Q|w \rangle$$

3.1.5 Transformée d'évaluation \mathcal{E} . Série de Chen

Les séries formelles permettent de coder le comportement entrée/sortie des systèmes contrôlables (analytiques affines en la commande). La fonction de décodage est appelée *transformée d'évaluation* (voir [34]). On associe tout d'abord à chaque commande u_1, u_2 du système une lettre de codage 1, 2 et une fonction numérique représentant son intégrale ξ_1, ξ_2 . Pour toute lettre $z \in Z = 1, 2$ et pour tout $t \in \mathbb{R}$ on a donc :

$$u_z \rightarrow z \rightarrow \xi_z(t) = \int_0^t u_z(\tau) d\tau$$

L'évaluation d'un mot $w \in X^*$ pour l'entrée $u = (u_1, u_2)^T$ (dans le cas d'un alphabet à deux lettres) est alors la fonction notée $\mathcal{E}_u(w)(t)$ définie par induction sur la longueur de w :

$$\begin{cases} \mathcal{E}_u(\epsilon)(t) &= 1 \\ \mathcal{E}_u(wz)(t) &= \int_0^t \mathcal{E}_u(w)(\tau) d\xi_z(\tau) \end{cases}$$

définition que la linéarité de l'intégration permet d'étendre aux polynômes, puis aux séries (sous réserve de convergence).

On définit alors la *série de Chen* $\mathcal{C}_u(t)$ d'une entrée u définie sur l'intervalle de temps $[0, t]$ par :

$$w \mapsto \langle \mathcal{C}_u(t) | w \rangle = \mathcal{E}_u(w)(t)w$$

Une des propriétés importantes de la transformée d'évaluation est son comportement face au produit de 2 séries : *l'évaluation du produit de mélange de 2 séries est égal au produit des évaluations respectives de chaque série*

$$\mathcal{E}_u(S \circ T) = \mathcal{E}_u(S) \cdot \mathcal{E}_u(T)$$

3.1.6 Série génératrice d'un système dynamique, \mathcal{Y} , série double

On associe tout d'abord aux champs de vecteurs les lettres des entrées correspondantes :

$$u_x \rightarrow x \rightarrow X_x$$

Par suite on définit le morphisme qui à tout mot w associe l'opérateur différentiel obtenu par composition des champs de vecteur décodés un à un :

$$\mathcal{Y}(w) = \begin{cases} Id & , \text{ si } w = \epsilon \\ X_z \circ (\mathcal{Y}(v)) & , \text{ si } w = zv, \text{ où } z \text{ est une lettre} \end{cases}$$

morphisme qu'on étend canoniquement aux polynômes non commutatifs.

Le vecteur d'état à l'instant t d'un système (Σ) initialisé en x_0 est donné par la "formule fondamentale" de Fliess [8]

$$x(t) = \lambda_{x_0} \circ [\langle \mathcal{C}_u(t) | w \rangle] [\mathcal{Y}(w) \circ Id_x]$$

où λ_{x_0} représente l'évaluation en x_0 . L'application qui associe à la commande u définie sur $[0, t[$ l'état atteint $x(t)$ est appelée *fonctionnelle causale analytique* (causale parce que $x(t)$ ne dépend "que du passé").

Si on introduit la série génératrice S (ou "série de Fliess") du système:

$$w \mapsto \langle S | w \rangle = \lambda_{x_0} \circ [\mathcal{Y}(w) \circ Id_x]$$

on peut écrire plus simplement:

$$\begin{aligned} x(t) &= \sum_{w \in \mathcal{Z}^*} \langle C_u(t)|w \rangle \langle S|w \rangle \\ &= \langle C_u(t) || S \rangle \end{aligned}$$

où la dernière notation avec la double barre indique la somme infinie, (convergente dans un voisinage de l'instant initial) qui prolonge naturellement le produit scalaire.

Il nous sera aussi utile de revenir aux notations sous forme d'opérateurs, en rappelant que $C_u(t)$ (resp. S) est défini à partir de l'opérateur d'évaluation \mathcal{E} (resp. de l'opérateur \mathcal{Y}). On introduit pour cela le produit tensoriel et la somme suivante qui est une représentation de l'application identité de $\mathbb{R} \langle \mathcal{Z} \rangle$ dans lui même:

$$\sum_{w \in \mathcal{Z}^*} w \otimes w$$

Le produit tensoriel des mots induit un produit tensoriel des opérateurs, et l'on obtient alors la formule:

$$x(t) = \lambda_{x_0} \circ \left[(\mathcal{E}_u \otimes \mathcal{Y}) \circ \left(\sum_{w \in \mathcal{Z}^*} \langle C_u(t)|w \rangle \langle S|w \rangle \right) \right]$$

L'intérêt de cette formule réside dans le fait que \mathcal{E}_u est un morphisme d'algèbres commutatives pour le produit de mélange, et que \mathcal{Y} est un morphisme d'algèbres associatives (non commutatives) pour le produit usuel (ou "de Cauchy") des séries formelles.

3.2 Algèbre de Lie libre et base de Lyndon

3.2.1 Les mots de Lyndon

Nous supposons choisi un ordre total sur les lettres de \mathcal{Z} , et \mathcal{Z}^ muni de l'ordre lexicographique.*

Definition 3.2.1 *Un mot $l \in \mathcal{Z}^+$ est dit "de Lyndon" ssi il est strictement inférieur à chacun de ses facteurs droits propres.*

Exemple: 11212 est un mot de Lyndon car il est inférieur à ses 4 facteurs droits 1212, 212, 12, 2.

Voici la liste des mots de Lyndon (triée par ordre lexicographique croissant) que nous utiliserons par la suite pour effectuer nos calculs.

$$\{1, 1112, 112, 1122, 12, 122, 1222, 2\}$$

		<i>card</i> (\mathcal{Z})						
		2	3	4	5	6	7	... 13
<i>k</i> (longueur maximale des mots)	1	2	3	4	5	6	7	... 13
	2	3	6	10	15	21	28	... 91
	3	5	14	30	55	91		
	4	8	32	90				
	5	14	80					
	6	23						
	7	41						
	8	71						

TAB. 3.1 - Nombre de mots de Lyndon

Proposition 3.2.1 *Tout mot du monoïde libre se factorise de façon unique en produit décroissant de mots de Lyndon.*

$$\forall w \in \mathcal{Z}^+, \quad w = l_1 l_2 \cdots l_n, \quad l_1 \geq l_2 \geq \cdots \geq l_n$$

Le nombre de mots de Lyndon de k lettres au plus peut être calculé par la fonction de Möbius. Le tableau 3.1 en donne les valeurs quand celles-ci sont inférieures à 100.

3.2.2 Crochetage des mots de Lyndon

Soit $l = uv$ avec $u, v \in \mathcal{Z}^+$, on dira que cette factorisation est *standard* ssi v est le plus long facteur droit de l qui est un mot de Lyndon. A chaque mot l , on associera un monôme de Lie noté $[l]$ défini récursivement par:

$$\begin{aligned} [l] &= l && \text{ssi } l \in \mathcal{Z} \\ [l] &= [[u], [v]] && \text{ssi la factorisation } uv \text{ est standard} \end{aligned}$$

3.2.3 Algorithme de génération des mots de Lyndon

Ceci montre comment les mots de Lyndon peuvent coder des champs de vecteurs et leurs crochets de Lie "sans rien faire", on prend simplement garde à bien mémoriser la factorisation standard lors de la génération des mots (*cf.* les sources en annexe C)

3.2.4 Mots de Širšov

les mots de Širšov sont le miroir des mots de Lyndon, les propriétés s'en transportant de façon naturelle. Leur utilisation lors de la construction d'une base de l'algèbre de Lie libre permet de garder à l'interprétation des crochets des mots le sens usuel des crochets de Lie des champs de vecteurs ainsi qu'une décomposition en produit croissant d'exponentielles des mots de Širšov pour la série double et non pas décroissant comme pour les mots de Lyndon. On notera \mathcal{S} l'ensemble des mots de Širšov et \mathcal{S}_s l'ensemble des mots de Širšov de longueur strictement inférieure à s

3.2.5 L'algèbre de Lie libre

Soit \mathcal{Z} un alphabet totalement ordonné. Soit $\mathcal{M}(\mathcal{Z})$ le magma libre engendré par \mathcal{Z} , i.e. l'ensemble des mots complètement parenthésés construits sur l'alphabet \mathcal{Z} . On désignera par $\mathcal{L}(\mathcal{Z})$ l'algèbre non associative libre engendrée par \mathcal{Z} , i.e. l'ensemble des combinaisons linéaires de $\mathcal{M}(\mathcal{Z})$ à coefficients dans le corps K .

L'algèbre de Lie libre sur \mathcal{Z} , notée $\text{Lie} \langle \mathcal{Z} \rangle$, s'identifie au quotient de $\mathcal{L}(\mathcal{Z})$ par l'idéal \mathcal{J} engendré par les éléments de la forme:

$$\begin{aligned} Q(x) &= [x, x] && \text{pour } x \in \mathcal{L} \langle \mathcal{Z} \rangle, \\ J(x, y, z) &= [x[y, z]] + [y[z, x]] + [z[x, y]] && \text{pour } x, y, z \in \mathcal{L} \langle \mathcal{Z} \rangle. \end{aligned}$$

Une telle algèbre vérifie en effet les axiomes des *algèbres de Lie*:

$$\begin{aligned} [p, p] &= 0 \\ [p, q] &= -[q, p] && \text{(antisymétrie)} \\ [[p, q], r] + [[q, r], p] + [[r, p], q] &= 0 && \text{(relations de Jacobi)} \end{aligned}$$

pour tout polynômes $p, q, r \in \text{Lie} \langle \mathcal{Z} \rangle$

Tout polynôme de Lie peut être interprété comme un polynôme en variables non commutatives en considérant les crochets comme des commutateurs:

$$\forall p, q \in \text{Lie} \langle \mathcal{Z} \rangle, \quad [p, q] = pq - qp. \quad (3.1)$$

3.2.6 Normalisation des polynômes de Lie

On donne ici une forme normale des polynômes de Lie (voir Schützenberger [43]). Soient a, b, c des monômes de Lie quelconques. Considérons le système de réécriture suivant, prenant en compte dans \mathcal{Z}^* l'ordre lexicographique :

$$\left\{ \begin{array}{ll} [a, a] & \longrightarrow 0 \\ [a, b] & \longrightarrow -[b, a] \quad \text{ssi } a > b \\ [[a, b], c] & \longrightarrow [a, [b, c]] + [[a, c], b] \quad \text{ssi } b < c \end{array} \right.$$

Proposition 3.2.2 *le système de réécriture est confluent (le résultat est indépendant de l'ordre d'application des règles) et noethérien (la réécriture s'arrête en un nombre fini de pas).*

Les crochets irréductibles qui s'en déduisent forment la *base de Lyndon*.

3.3 Algèbre enveloppante

3.3.1 L'algèbre enveloppante, La base de Poincaré–Birkoff–Witt

Théorème 3.3.1 (Poincaré-Birkoff-Witt)

Si une algèbre de Lie \mathcal{L} admet une base $\{P_1, P_2, \dots, P_n, \dots\}$ alors son algèbre enveloppante admet comme base les produits de la forme $P_{i_1} P_{i_2} \dots P_{i_n}$ avec $P_{i_1} \geq P_{i_2} \geq \dots \geq P_{i_n}$ et $n \geq 0$.

Ce théorème appliqué à la base de Lyndon de l'algèbre de Lie libre nous fournit une base de l'algèbre $K \langle \mathcal{Z} \rangle$ des polynômes non commutatifs, que nous noterons *PBWL*.

Remarque: Il existe une bijection entre les deux bases \mathcal{Z}^* et *PBWL*.

Soit $w \in \mathcal{Z}^*$, considérons la factorisation en produit décroissant de mots de Lyndon:

$$w = l_1 l_2 \dots l_n \quad \text{avec } l_1 \geq l_2 \dots \geq l_n.$$

Il lui correspond alors un élément Q_w de la base *PBWL* défini par

$$Q_w = [l_1][l_2] \dots [l_n].$$

où $[l_1], [l_2], \dots, [l_n]$ désignent les polynômes de Lie associés aux mots de Lyndon l_1, l_2, \dots, l_n .

Remarque: Les polynômes Q_w sont homogènes de degré $|w|$.

3.3.2 La base duale : les S_l

Definition 3.3.1 *La "base duale" $PBWL^* = \{S_w \in K \langle \mathcal{Z} \rangle \mid w \in \mathcal{Z}^*\}$ est définie par les relations d'orthogonalité :*

$$\langle S_w \mid Q_{w_1} \rangle = \delta_w^{w_1} \quad \forall w, w_1 \in \mathcal{Z}^*.$$

Remarque:

- Ces équations de définitions forment en fait un "système triangulaire" permettant de calculer effectivement les S_l par de simples inversions de matrices triangulaires diagonales constantes.
- Les polynômes S_w sont homogènes de degré $|w|$.

Construction effective

Les polynômes S_w vérifient les formules récursives suivantes :

1. $S_\varepsilon = 1$
2. $S_{xu} = xS_u$ si x désigne la première lettre du mot de Lyndon xu .
3. $S_w = \frac{S_{l_1}^{\alpha_1} w S_{l_2}^{\alpha_2} \dots w S_{l_k}^{\alpha_k}}{\alpha_1! \alpha_2! \dots \alpha_k!}$ si w n'est pas un mot de Lyndon et admet pour factorisation décroissante $w = l_1^{\alpha_1} l_2^{\alpha_2} \dots l_k^{\alpha_k}$, $l_1 > l_2 > \dots > l_k$

Proposition 3.3.1 Les polynômes S_w forment une base de $K \langle \mathcal{Z} \rangle$.

3.4 Opérateur de transport \mathcal{H}

On savait déjà que la série de Chen était l'exponentielle d'une série de Lie lorsqu'à la même époque plusieurs auteurs ont fait apparaître ce résultat : on peut voir l'état comme un produit infini d'exponentielles d'éléments d'une base de l'algèbre de Lie. Cette représentation s'avérait canonique et surtout proposait une méthode pour le calcul des coefficients de ces développements. Huillet, Monin et Salut ([19]) en 1990 s'appuyaient sur les travaux de Magnus sur les opérateurs différentiels pour les étendre aux opérateurs de transport, Sussmann ([46]) partait de la série de fliess. Ils utilisaient la base de P.Hall. Reutenauer ([33]) arrivait à un résultat similaire travaillant sur la combinatoire des mots de Lyndon cette fois-ci. C'est dans la lignée des travaux de ce dernier que se situe la présentation qui en est faite ici.

3.4.1 La factorisation de la série double

Dans l'ensemble des séries doubles $\mathbb{R} \langle \langle \mathcal{Z}^* \otimes \mathcal{Z} \rangle \rangle$, on peut définir l'exponentielle d'un élément homogène de la façon suivante :

$$e^{S \otimes Q} = \sum_{j \in \mathbb{N}} \frac{1}{j!} S^{w_j} \otimes Q^j$$

Théorème 3.4.1 *La série diagonale $\sum_{w \in Z^*} w \otimes w$ admet le développement suivant :*

$$\sum_{w \in Z^*} w \otimes w = \prod_{b \in \mathcal{B}} e^{S_b \otimes Q_b}$$

où les b viennent dans l'ordre strictement croissant si l'on prend pour \mathcal{B} l'ensemble des mots de Širšov , ou encore un ordre strictement décroissant en prenant pour \mathcal{B} l'ensemble des mots de Lyndon.

Ce produit infini se réduit à un produit fini dans le cas où l'algèbre de Lie des Q_b est une algèbre de Lie nilpotente.

3.4.2 Définition

Le retour à la géométrie se fait en prenant l'image de ce produit par $\mathcal{E} \otimes \mathcal{Y}$, On définit ainsi de façon unique l'opérateur de transport :

$$\mathcal{H} = \prod_{b \in \mathcal{B}} e^{\xi_b(t) Y_b}$$

L'état du système s'obtient alors de la manière suivante :

$$x(t) = \lambda_{x(0)} \circ \mathcal{H} \circ Id_x$$

Bien sûr un tel développement n'est en général pas accessible. On n'en calcule qu'un nombre fini de termes au moyen de la transformée d'évaluation (voir l'annexe C pour les programmes) et du morphisme canonique \mathcal{Y} .

Chapitre 4

Systemes nilpotents

4.1 Définition. Coordonnées exponentielles

La série centrale descendante d'une algèbre de Lie \mathfrak{g} est définie récursivement par :

$$C^1 \mathfrak{g} = \mathfrak{g}, \quad C^{k+1} \mathfrak{g} = [\mathfrak{g}, C^k \mathfrak{g}]$$

une algèbre de Lie est alors dite nilpotente d'ordre s lorsque sa série centrale descendante se finit au $s^{\text{ième}}$ terme :

$$C^s \mathfrak{g} = 0$$

Dans le cas d'un système commandable ceci caractérise de façon formelle l'idée que tous les champs de vecteurs de l'algèbre de Lie faisant intervenir au moins (s) crochets sont nuls.

Definition 4.1.1 *Un système dont l'algèbre de Lie de contrôle est nilpotente est dit nilpotent.*

Rappelons que pour de tels systèmes l'opérateur de transport peut s'exprimer comme un produit *fini* d'exponentielles :

$$\mathcal{H} = \prod_{\substack{l \in \mathcal{S}, |l| < s \\ \text{ordre croissant}}} \exp(\xi_l Y_l)$$

l'état se retrouve alors par la formule :

$$x(t) = \lambda_{x(0)} \circ \mathcal{H} \circ Id_x$$

L'expression *coordonnées exponentielles* désignera les ξ_l

4.2 Nilpotentisation

On ne connaît pas à l'heure actuelle d'algorithme permettant de rendre un système nilpotent par bouclage, sauf dans le cas des systèmes chaînés (voir section 4.4). Pour les systèmes non chaînés il faut encore les calculer à la main

Exemple de la pièce de monnaie

Rappelons les équations de la pièce de monnaie (sa dynamique est décrite plus en détail dans la section 2.1.1) :

$$(\Sigma) \quad \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} u_1(t) + \begin{bmatrix} \cos(x_3) \\ \sin(x_3) \\ 0 \\ 1 \end{bmatrix} u_2(t)$$

ce système n'est pas nilpotent, cependant on peut le rendre tel par bouclage, en posant :

$$\begin{cases} u_1 = v_1 \\ u_2 = [x_1 \sin(x_3) - x_2 \cos(x_3)]v_1 + v_2 \end{cases}$$

il devient alors :

$$(\Sigma_{\text{nilp}}) \quad \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} \cos(x_3)[x_1 \sin(x_3) - x_2 \cos(x_3)] \\ \sin(x_3)[x_1 \sin(x_3) - x_2 \cos(x_3)] \\ 1 \\ x_1 \sin(x_3) - x_2 \cos(x_3) \end{bmatrix} v_1(t) + \begin{bmatrix} \cos(x_3) \\ \sin(x_3) \\ 0 \\ 1 \end{bmatrix} v_2(t)$$

$$= Z_1(x) v_1(t) + Z_2(x) v_2(t)$$

en effet en calculant son algèbre de Lie :

$$Y_1 = Z_1$$

$$Y_2 = Z_2$$

$$\begin{aligned} Y_{21} &= Y_2 Y_1 - Y_1 Y_2 \\ &= \sin(x_3) \frac{\partial}{\partial x_1} - \cos(x_3) \frac{\partial}{\partial x_2} \end{aligned}$$

$$\begin{aligned} Y_{211} &= Y_{21} Y_1 - Y_1 Y_{21} \\ &= \frac{\partial}{\partial x_4} \end{aligned}$$

$$Y_{221} = 0$$

$$Y_{2111} = 0$$

on vérifie aisément que les champs de vecteurs d'ordre supérieur sont égaux à 0 , (Σ_{nilp}) est donc nilpotent d'ordre 4.

4.3 Système nilpotent libre

4.3.1 Algèbre de Lie nilpotente libre

Soit \mathcal{Z} un alphabet, par définition, l'algèbre de Lie *nilpotente libre* à l'ordre s $\text{Lie}_s \langle \mathcal{Z} \rangle$ est un modèle initial (unique à isomorphisme près) dans la catégorie des K -algèbres de Lie nilpotentes à l'ordre s . Autrement dit, elle vérifie la propriété universelle suivante :

$$\begin{aligned} \exists \alpha : \mathcal{Z} \rightarrow \text{Lie}_s \langle \mathcal{Z} \rangle \text{ tel que } \forall \varphi : \mathcal{Z} \rightarrow \mathcal{L}, \quad {}^s \mathcal{L} = (0), \\ \exists ! \bar{\varphi} : \text{Lie}_s \langle \mathcal{Z} \rangle \rightarrow \mathcal{L}, \quad \varphi = \bar{\varphi} \circ \alpha \end{aligned}$$

On construit une algèbre $\text{Lie}_s \langle \mathcal{Z} \rangle$ nilpotente à l'ordre s en considérant le quotient de $\text{Lie} \langle \mathcal{Z} \rangle$ par l'idéal des polynômes de Lie engendré par les crochets de longueur strictement supérieure à n . Cette algèbre est isomorphe (en tant qu'espace vectoriel) à

$$\text{Lie}_s \langle \mathcal{Z} \rangle = \{p \in \text{Lie} \langle \mathcal{Z} \rangle \mid \text{degre}(p) < s\}$$

On choisit de représenter les polynômes de $p \in \text{Lie}_s \langle \mathcal{Z} \rangle$ dans la base de Širšov en se restreignant aux mots l tels que $|l| < s$. On note $\{l_1, l_2, \dots, l_d\}$ cet ensemble.

4.3.2 Groupe et système nilpotents libres

On considère le groupe de Lie \mathcal{G}_s associé à $\text{Lie}_s \langle \mathcal{Z} \rangle$ l'algèbre de Lie nilpotente libre. Dans ce cas (voir notamment Ree [40]) :

$$\forall g \in \mathcal{G}_s, \quad \exists ! p \in \text{Lie}_s \langle \mathcal{Z} \rangle, \quad g = e^p$$

Théorème 4.3.1 *Tout élément $g \in \mathcal{G}_s$ se décompose de façon unique en produit croissant d'exponentielles :*

$$g = e^{\xi_1[l_1]} e^{\xi_2[l_2]} \dots e^{\xi_d[l_d]}$$

où $\xi_1, \xi_2, \dots, \xi_d \in K$.

Par définition, le *système nilpotent libre* a pour espace d'état le groupe \mathcal{G}_s et sa dynamique est donnée par les champs de vecteurs $(X_i)_{i \in I}$ invariants à gauche (ou à droite) qui engendrent l'algèbre de Lie $\text{Lie}_s < \mathcal{Z} >$ du groupe \mathcal{G}_s .

4.3.3 Réalisation minimale d'un système nilpotent libre

On cherche ici à calculer les équations dynamiques d'un système dont l'algèbre de Lie des champs de vecteurs soit nilpotente libre. Dans le cas d'un groupe nilpotent libre les coordonnées exponentielles, définissent un système de coordonnées pour le groupe, pour son algèbre de Lie et pour la variété.

On pose :

$$x_l = \mathcal{E}(S_l), \quad \text{pour } |l| \leq n$$

où les S_l sont donnés pour tout mot de Širšov $l = wi$ (où i est une lettre et w un mot) par :

$$\begin{aligned} S_{wi} &= S_w i \\ &= \frac{S_{l_1}^{\alpha_1} S_{l_2}^{\alpha_2} \dots S_{l_k}^{\alpha_k}}{\alpha_1! \alpha_2! \dots \alpha_k!} i \end{aligned}$$

en supposant que w vérifie

$$w = l_1^{\alpha_1} l_2^{\alpha_2} \dots l_k^{\alpha_k}, \quad l_1 > l_2 > \dots > l_k$$

Les polynômes S_l correspondent aux séries génératrices des coordonnées exponentielles considérées comme fonctionnelles d'entrée / sortie. Leur réalisation est donnée par :

$$\dot{x}_l = \frac{x_{l_1}^{\alpha_1} x_{l_2}^{\alpha_2} \dots x_{l_k}^{\alpha_k}}{\alpha_1! \alpha_2! \dots \alpha_k!} u_i(t)$$

Pour ces systèmes l'action d'un champ de vecteurs Y_i sur une coordonnée correspond au résiduel à gauche sur les mots de codage :

$$\begin{aligned} Y_i \circ x_l &= x_w && \text{ssi } l = wi \\ &= 0 && \text{sinon} \end{aligned}$$

Ainsi à l'ordre 5 :

$$(\Sigma NL(5)) \quad \left\{ \begin{array}{l} \dot{\xi}_1 = u_1 \\ \dot{\xi}_2 = u_2 \\ \dot{\xi}_{21} = \xi_2 u_1 \\ \dot{\xi}_{211} = \xi_{21} u_1 \\ \dot{\xi}_{221} = \frac{1}{2!} \xi_2^2 u_1 \\ \dot{\xi}_{2111} = \xi_{211} u_1 \\ \dot{\xi}_{2211} = \xi_{221} u_1 \\ \dot{\xi}_{2221} = \frac{1}{3!} \xi_2^3 u_1 \end{array} \right.$$

4.3.4 Forme canonique des systèmes nilpotents libres

Tout système dynamique $\dot{x} = u_1 Y_1 + u_2 Y_2$ nilpotent d'ordre (s) est (localement) obtenu par morphisme à partir du système nilpotent libre $\Sigma NL(s)$. Ce dernier est défini par :

- son espace d'état, confondu avec le groupe nilpotent libre

$$\mathcal{G}_s : \text{ engendré par } \{e^p, p \in \mathcal{L}ie_s(1, 2)\}$$

où les éléments (p) de l'algèbre de Lie nilpotente libre $\mathcal{L}ie_s(1, 2)$ sont de degré : $\deg(p) \leq s$

- son algèbre de Lie de commande (l'algèbre de Lie nilpotente libre)

Le changement de coordonnées se fait de la façon suivante. On intègre explicitement les flots du système, l'opérateur de transport étant un produit fini d'exponentielles. On obtient ainsi explicitement l'application qui donne les (x) en fonction des (ξ). On calcule alors le changement de coordonnées inverse dans le cas de systèmes nilpotents libres. Si le système n'est pas libre on obtient une des applications inverse dont on ne sait si elle est satisfaisante d'un point de vue automatique. Un exemple de calcul est donnée dans la section du calcul des coordonnées exponentielles de la cible pour le problème de tir : méthode d'intégration des flots (section 5.2).

Dans le cas où le système de départ est également nilpotent libre et ses champs de vecteurs sont polynomiaux le résultat est plus fort :

Lemma 4.3.1 *deux réalisations polynomiales du système nilpotent libre sont égales modulo un changement polynomial des coordonnées locales .*

Ceci se démontre par le calcul dans le groupe nilpotent libre.

4.3.5 Calculs dans le groupe nilpotent libre

Exemple pour un groupe d'ordre 4 \mathcal{G}_4 :

Les principaux calculs réalisés symboliquement dans \mathcal{G}_4 avec Axiom utilisent la forme canonique en produits croissants d'exponentielles de Lie (ordre lexicographique sur les mots de Širšov). Toutefois ils sont présentés ici dans l'ordre lexicographique par longueur pour en simplifier la lecture et sous forme vectorielle (le vec).

Changement de coordonnées :

$$p = c_1 [1] + c_2 [2] + c_{[21]} [21] + c_{[211]} [211] + c_{[221]} [221]$$

$$e^p = \begin{pmatrix} c_1 \\ c_2 \\ c_{21} + \frac{1}{2}c_2c_1 \\ c_{211} + \frac{1}{2}c_{21}c_1 + \frac{1}{6}c_2c_1^2 \\ c_{221} + \frac{1}{2}c_2c_{21} + \frac{1}{6}c_2^2c_1 \end{pmatrix}$$

on peut ainsi transformer l'exponentielle d'une somme en produits croissants d'exponentielles.

Produit de 2 éléments $g, h \in \mathcal{G}_3$:

$$g \bullet h = \begin{pmatrix} g_1 \\ g_2 \\ g_{21} \\ g_{211} \\ g_{221} \end{pmatrix} \bullet \begin{pmatrix} h_1 \\ h_2 \\ h_{21} \\ h_{211} \\ h_{221} \end{pmatrix}$$

$$= \begin{pmatrix} g_1 + h_1 \\ g_2 + h_2 \\ g_{21} + h_{21} + g_2h_1 \\ g_{211} + h_{211} + g_{21}h_1 + \frac{1}{2!}g_2h_1^2 \\ g_{221} + h_{221} + g_2h_{21} + \frac{1}{2!}g_2^2h_1 \end{pmatrix}$$

Calcul de l'inverse d'un élément du groupe : il suffit de résoudre le système d'équations algébriques issu de l'identification de $g \circ g'$ avec l'élément neutre 1 où g' est l'inverse recherchée de g . Ces équations se résolvent linéairement.

4.4 Systèmes chaînés

Les systèmes chaînés sont les plus simples des systèmes nilpotents. Le système chaîné, nilpotent à l'ordre n (n équations) s'écrit :

$$(\Sigma_{\text{chain}}) \begin{cases} \dot{\xi}_1 & = u_1 \\ \dot{\xi}_2 & = u_2 \\ \dot{\xi}_{21} & = u_1 \xi_2 \\ \dot{\xi}_{211} & = u_1 \xi_{21} \\ & \vdots \\ \dot{\xi}_{2\underbrace{1\dots 1}_{n-1}} & = u_1 \xi_{2\underbrace{1\dots 1}_{n-2}} \end{cases}$$

ses champs de vecteurs :

$$Y_1 = \begin{pmatrix} 1 \\ 0 \\ \xi_2 \\ \xi_{21} \\ \vdots \\ \xi_{21^{n-1}} \end{pmatrix} \quad Y_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Les systèmes chaînés sont *nilpotents* car seules les combinaisons linéaires des champs de vecteurs :

$$(Y_1, Y_2, Y_{12} = \text{ad}_{Y_1}^1 Y_2, \dots, Y_{1^{n_2}} = \text{ad}_{Y_1}^{n_2} Y_2)$$

sont non identiquement nulles.

4.4.1 Propriété

Les coordonnées exponentielles et les coordonnées habituelles d'un point ξ sont les mêmes (à une permutation près).

$$\begin{aligned} \xi &= (\xi_1, \xi_2, \xi_{21}, \dots, \xi_{21^n})^T \\ &= \lambda_0 \circ e^{\xi_1 [1]} e^{\xi_{21^n} [21^n]} \dots e^{\xi_{21} [21]} e^{\xi_2 [2]} \circ Id_x \end{aligned}$$

4.4.2 Mise sous forme chaînée

Tout système nilpotent de dimension 4, à 2 entrées et donc 2 champs de vecteurs peut se mettre sous forme chaînée par bouclage statique.

4.5 Forme triangulaire des systèmes nilpotents

Un système dynamique est dit sous forme *strictement triangulaire* ssi il existe un ordre sur les variables d'état $x = (x_1, \dots, x_n)$ tel que

$$\begin{cases} \dot{x}_1 = f_1(u) \\ \dot{x}_2 = f_2(x_1, u) \\ \dot{x}_3 = f_3(x_2, x_1, u) \\ \vdots \\ \dot{x}_n = f_n(x_{n-1}, \dots, x_1, u) \end{cases}$$

Tout système nilpotent peut être mis sous une forme strictement triangulaire et polynomiale par un simple changement de coordonnées locales. Ceci est une conséquence directe d'un résultat dû à M. Kawski [24]:

Théorème 4.5.1 Soient X_1, \dots, X_m des champs de vecteurs réels analytiques sur une variété réelle analytique M qui engendrent une algèbre de Lie nilpotente $L = L(X_1, \dots, X_m)$. Si $p \in M$ est tel que $\dim L(p) = n$, alors il existe des coordonnées locales (x_1, \dots, x_n) dans un voisinage de p et une dilatation $\delta_t(x) = (t^{r_1}x_1, \dots, t^{r_n}x_n)$ telle que relativement à ces coordonnées, X_1, \dots, X_m ont des coefficients polynomiaux et sont de degré -1 par rapport à cette dilatation δ_t .

Ceci va permettre de faire le lien avec les systèmes plats sur un terrain commun : celui des corps différentiels

4.6 Triangularité et corps différentiels

4.6.1 Platitude. Triangularité. Extensions de corps différentielles

Dans [9], M. Fliess définit un système dynamique comme une extension différentielle $\mathcal{D} = k \langle x, u, y \rangle$ finiment engendrée du corps différentiel k des coefficients. La sortie y est dite *plate* ssi

1. Les quantités $y = (y_1, \dots, y_p)$ ne sont liées par aucune relation algébrique-différentielle.
2. Les quantités $x = (x_1, \dots, x_n)$ et $u = (u_1, \dots, u_m)$ dépendent algébriquement (non différentiellement) de y et ses dérivées \dot{y}, \ddot{y}, \dots , autrement dit, l'extension \mathcal{D} est algébrique sur l'extension transcendante pure $k \langle y \rangle$.

(un exemple de traitement de la planification de trajectoire par la méthode des systèmes plats figure en annexe A) La forme triangulaire polynomiale (section 4.5) s'interprète très simplement. On obtient l'extension $\mathcal{D} = k \langle x, u \rangle$ à partir de l'extension transcendante pure $K_0 = k \langle u \rangle$ en construisant une tour de corps différentiels en ajoutant les quantités une à une :

$$K_0 \subset K_1 \subset \dots \subset K_n = \mathcal{D}$$

tels que $\dot{x}_i \in K_{i-1}, K_i = K_{i-1} \langle x_i \rangle$ pour $i = 1..n$.

On dispose du résultat suivant (voir [23]):

Lemma 4.6.1 *Soit K un corps différentiel de caractéristique nulle. Soit ξ un élément d'une extension différentielle tel que $\dot{\xi} = a \in K$, où a n'est pas une dérivée dans K . Alors ξ est transcendant sur K , $K \langle \xi \rangle$ est une extension de Picard-Vessiot de K et le groupe de Galois différentiel de $K \langle \xi \rangle / K$ est isomorphe au groupe additif des constantes de K .*

On a donc la classification suivante:

1. Les systèmes plats qui s'obtiennent à partir des extensions transcendentes pures en ajoutant des quantités algébriques.
2. Les systèmes nilpotents qui s'obtiennent à partir des extensions transcendentes pures en ajoutant des intégrales et des quantités algébriques.

A noter que ce nombre n'est pas unique car on peut construire plusieurs tours d'extensions différentielles atteignant le même système (corps différentiel) suivant le corps de départ.

Le nombre minimal de quantités intégrales correspond au défaut de platitude du système nilpotent.

4.6.2 Exemple du "Plate-Ball"

Considérons par exemple le système suivant, qui donne une approximation du système "Plate-Ball" :

$$\left\{ \begin{array}{l} \dot{x}_1 = u_1 \\ \dot{x}_2 = u_2 \\ \dot{x}_3 = u_1 x_2 \\ \dot{x}_4 = u_1 x_3 \\ \dot{x}_5 = \frac{1}{2} u_1 x_2^2 \end{array} \right\} \quad \begin{array}{l} \text{sous-système plat} \\ \text{sortie plate: } y = (x_1, x_4) \end{array}$$

Une première construction du corps \mathcal{D} représentant le système est faite à partir de $K_0 = k \langle u \rangle$:

$$K_0 \subset K_1 = K_0 \langle x_1 \rangle \subset K_2 = K_1 \langle x_2 \rangle \quad (4.1)$$

$$\subset K_3 = K_2 \langle x_3 \rangle \subset K_4 = K_3 \langle x_4 \rangle \quad (4.2)$$

$$\subset K_5 = K_4 \langle x_5 \rangle = \mathcal{D} \quad (4.3)$$

Une deuxième est effectuée en partant cette fois de $K_0 = k \langle u \rangle$:

$$K_0 \subset K_1 = K_0 \langle x_5 \rangle$$

en effet les quatre premières équations forment un sous-système qui est plat, de sortie plate $y = (x_1, x_4)$. On vérifie en effet que les autres grandeurs de ce sous système se déduisent algébriquement de la sortie plate et de ses dérivées :

$$x_3 \dot{x}_1 = \dot{x}_4 \quad \Rightarrow \quad x_3 \in K_0$$

$$x_2 \dot{x}_1 = \dot{x}_3 \quad \Rightarrow \quad x_2 \in K_0$$

$$u_2 = \dot{x}_2 \quad \Rightarrow \quad u_2 \in K_0$$

$$u_1 = \dot{x}_1 \quad \Rightarrow \quad u_1 \in K_0$$

par ailleurs $\dot{x}_5 \in K_0$ et $x_5 \notin K_0$.

Le défaut de platitude de ce système est donc 1.

Chapitre 5

Planification de trajectoire d'un système nilpotent

5.1 Mise en équation

- 1^e partie : calcul des coordonnées de la cible
- 2^e partie : motion planning dans le système nilpotent libre

Si l'on remplace dans le développement exponentiel les coordonnées $\xi_1, \xi_2, \dots, \xi_n$ par des constantes k_1, k_2, \dots, k_n , ce vecteur constant représente des "coordonnées locales en x_0 " du point q :

$$q = \lambda_{x_0} \circ e^{k_1 Y_1} e^{k_2 Y_2} \dots e^{k_n Y_n} \circ Id_x$$

Le système considéré étant par hypothèse, complètement commandable, tout point de M admet donc une présentation (au moins), sous cette forme. Il faut donc calculer explicitement l'un de ces jeux de coordonnées exponentielles cibles, c'est l'objet de la troisième partie de ce chapitre.

Par ailleurs, si l'on peut agir indépendamment sur les champs de vecteurs Y_i , cette formule permet d'obtenir le point q simplement par une composition successive des flots de Y_1, Y_2, \dots, Y_d . Le problème de la commande exacte se trouve ainsi résolu. Ceci n'est pas possible dans le cas des *systèmes non holonomes* étudiés ici. Par exemple pour un système de dimension 5 à 2 entrées :

$$\dot{x} = u_1(t)Z_1(x) + u_2(t)Z_2(x)$$

dont on suppose que l'algèbre de Lie engendrée par Z_1 et Z_2 soit un espace vectoriel de dimension 5, admettant comme générateurs des champs de vecteurs $Y_1, Y_2, Y_3,$

Y_4, Y_5 en tout point linéairement indépendants. Dans ce cas on ne commande le système qu'au travers des 2 directions infinitésimales définies par Z_1 et Z_2 , et toute intervention de Z_1 et Z_2 entrainera des répercussions sur les 5 coordonnées de l'état atteint.

La deuxième partie du problème consiste donc à réaliser effectivement une commande qui donne lieu aux mêmes coordonnées exponentielles. Plusieurs méthodes seront développées au cours des chapitres suivants, elles visent toutes à résoudre le problème que l'on peut désormais qualifier de "planification de trajectoire du système nilpotent libre". La différence essentielle avec le problème de départ étant le caractère essentiellement algébrique de ce nouveau problème et la propriété singulière de ces systèmes dont la variété forme un groupe.

5.2 Coordonnées exponentielles de la cible

- méthode numérique (Sussmann)
- méthode géométrique (redressement)
- méthode d'intégration effective des flots

5.2.1 Inversion du système étendu

Cette méthode fut proposée par Sussmann (see [29], [47]). Elle consiste à choisir dans un premier temps une courbe paramétrée de l'espace d'état ($\gamma(t)$) (aussi simple que possible) menant le système du point de départ au point d'arrivée souhaité. On cherche alors à faire épouser cette trajectoire par un système *holonome* appelé *système étendu*. Ce dernier s'obtient à partir du système de départ en ajoutant suffisamment de directions infinitésimales (champs de vecteurs) pour obtenir un ensemble générateur de l'algèbre de Lie de commande du système en tout point de l'espace d'état. On calcule pour cela les crochets de Lie de longueur croissante à partir des champs de vecteurs initiaux, la condition sur le rang de l'algèbre de Lie assurant que ce procédé aboutit. Il suffit alors de décomposer la trajectoire (telle une droite parcourue à vitesse constante) sur cet ensemble :

$$\dot{\gamma}(t) = \sum_{b \in \mathcal{C}} \alpha_b(t) Y_b(\gamma(t))$$

Dans l'exemple de la pièce, $x(t) = \gamma(t)$ est solution du système étendu :

$$(\Sigma_{\text{ext}}) \quad \dot{x} = \alpha_1(t)Y_1(x) + \alpha_2(t)Y_2(x) + \alpha_3(t)Y_{12}(x) + \alpha_4(t)Y_{112}(x)$$

Ce système possède bien sûr la même algèbre de Lie de commande que le système de départ (Σ_S). Pour le calcul des composantes $\alpha(t)$ on fait le choix d'une trajectoire γ en prenant par exemple une droite parcourue à vitesse constante et on inverse le système d'équations (le système étendu est holonome).

On calcule alors les coordonnées exponentielles z_i de l'élément du groupe de Lie correspondant à l'entrée α en intégrant le système nilpotent libre pour 4 entrées à l'ordre 3 en ayant supprimés les champs de vecteurs nuls. C'est à dire les 7 équations :

$$\begin{aligned} \dot{z}_1 &= \alpha_4 & \dot{z}_4 &= \alpha_1 z_2 \\ \dot{z}_2 &= \alpha_3 & \dot{z}_5 &= \alpha_1 z_3 & , & \dot{z}_7 &= \alpha_1 \\ \dot{z}_3 &= \alpha_2 & \dot{z}_6 &= \alpha_1 z_5 \end{aligned}$$

en posant $Z_i = z_i(T)$ on a donc :

$$g_{\text{ext}} = e^{Z_1 [112]} e^{Z_2 [12]} e^{Z_3 [2]} e^{Z_4 [1][12]} e^{Z_5 [1][2]} e^{Z_6 [1][1][2]} e^{Z_7 [1]}$$

on réordonne les exponentielles par changement de coordonnées et loi de groupe pour obtenir les composantes dans la bonne base :

$$\begin{aligned} g_{\text{ext}} &= e^{Z_3 [2]} e^{(\frac{1}{2} Z_3^2 Z_7 + Z_3 Z_5) [122]} e^{(Z_3 Z_7 + Z_5 + Z_2) [12]} \\ &{}_0 e^{(\frac{1}{2} Z_3 Z_7^2 + Z_5 + Z_2 Z_7 + Z_6 + Z_4 + Z_1) [112]} e^{Z_7 [1]} \end{aligned}$$

5.2.2 Méthode géométrique

Il parfois possible d'obtenir géométriquement les coordonnées exponentielles de la cible. Ceci par une composition finie de mouvement le long de courbes intégrales des champs de vecteurs de l'algèbre de Lie de commande. On réordonne alors les exponentielles (flots) en leur décomposition canonique.

Par exemple pour la pièce de monnaie, pour mener le système du point de départ $x(0) = p$ au point d'arrivée $x(T) = q$ on les suppose égaux à $p = (-p_1, p_2, 0, 0)$ et $q = (0, 0, q_3, q_4)$, ce qui peut être fait sans perte de généralité. On peut alors tirer partie des quatres champs de vecteurs de (Σ_S) : $Y_1, Y_{12}, Y_{112}, Y_2 - Y_{112}$ de l'algèbre de Lie de commande . Ceux-ci deviennent en effet très simples en certaines régions de l'espace état :

- Y_1 quand $(x_1 = 0 \text{ et } x_2 = 0)$ —fait tourner le plan de la pièce :

$$Y_1 = (0, 0, 1, 0)^T$$

- Y_{12} quand ($x_3 = 0$) —fait glisser la pièce sur le côté droit :

$$Y_{12} = (0, -1, 0, 0)^T$$

- Y_{112} —fait déraiper la pièce sur place sans la faire avancer :

$$Y_{112} = (0, 0, 0, 1)^T$$

- $Y_3 = Y_2 - Y_{112}$ quand ($x_3 = 0$) —fait glisser la pièce tout droit sans rotation :

$$Y_3 = (1, 0, 0, 0)^T$$

On construit alors le chemin suivant : “avancer grace à (Y_3) de $p = (-p_1, p_2, 0, 0)$ à $(0, p_2, 0, 0)$, puis glisser sur la droite en suivant (Y_{12}) jusqu'à l'origine $(0, 0, 0, 0)$, pivoter de l'angle q_3 en utilisant (Y_1) amène en $(0, 0, q_3, 0)$, on déraipe enfin jusqu'à la position finale $(0, 0, q_3, q_4)$ désirée grace à (Y_{112})”.

Ce mouvement en quatre temps correspondant chacun à un seul champ de vecteurs s'exprime naturellement comme la composée des flots associés :

$$g_{\text{ext}} = e^{q_4 [112]} e^{q_3 [1]} e^{p_2 [12]} e^{p_1 ([2] - [112])}$$

expression qu'on réordonne. On obtient :

$$g_{\text{ext}} = e^{p_2 [12]} e^{(-p_1 + q_4 + p_2 q_3) [112]} e^{(p_1 + q_3) [1]} \quad (5.1)$$

5.2.3 Intégration des flots

On sait pouvoir écrire l'état du système comme un produit fini d'exponentielles. Par exemple à l'ordre 3 :

$$x(t) = e^{\xi_1 Y_1} e^{\xi_{211} Y_{211}} e^{\xi_{21} Y_{21}} e^{\xi_{221} Y_{221}} e^{\xi_2 Y_2} \circ Id|_{x(0)} \quad (5.2)$$

Notons y_1 le point de la variété où le champ de vecteur Y_1 mène le système en un temps ξ_1 . Cela revient à intégrer le flot de Y_1 pendant ξ_1 . On a alors :

$$x(t) = e^{\xi_{211} Y_{211}} e^{\xi_{21} Y_{21}} e^{\xi_{221} Y_{221}} e^{\xi_2 Y_2} \circ Id|_y \quad (5.3)$$

$$y = \Psi_{Y_1}(\xi_1, x(0)) \quad (5.4)$$

en itérant ce procédé d'intégration des champs de vecteurs on obtient finalement x comme une expression des ξ_l qu'on inverse. Exemple pour la main articulée :

$$\begin{cases} \dot{x}_1 &= u_1 \\ \dot{x}_2 &= u_2 \\ \dot{x}_3 &= x_1 u_2 - x_2 u_1 \\ \dot{x}_4 &= x_3 u_1 \\ \dot{x}_5 &= x_3 u_2 \\ x(0) &= (0, 0, 0, 0, 0) \end{cases}$$

L'algèbre de Lie de commande de ce système est engendrée par les champs de vecteurs $Y_1, Y_2, Y_{21} = [Y_2, Y_1], Y_{211} = [Y_{21}, Y_1], Y_{221} = [Y_2, Y_{21}], :$

$$Y_1 = \frac{\partial}{\partial x_1} - x_2 \frac{\partial}{\partial x_3} + x_3 \frac{\partial}{\partial x_4} \quad (5.5)$$

$$Y_2 = \frac{\partial}{\partial x_2} + x_1 \frac{\partial}{\partial x_2} + x_3 \frac{\partial}{\partial x_5} \quad (5.6)$$

$$Y_{21} = -2 \frac{\partial}{\partial x_3} + x_1 \frac{\partial}{\partial x_4} + x_2 \frac{\partial}{\partial x_5} \quad (5.7)$$

$$Y_{211} = -3 \frac{\partial}{\partial x_4} \quad (5.8)$$

$$Y_{221} = 3 \frac{\partial}{\partial x_5} \quad (5.9)$$

$$(5.10)$$

les crochets suivants sont tous nuls.

Le mouvement se fait en 5 étapes :

1. : de $x(0)$ en y suivant Y_1 pendant ξ_1

$$\begin{cases} \dot{x}_1 &= 1 \\ \dot{x}_2 &= 0 \\ \dot{x}_3 &= -x_2 \\ \dot{x}_4 &= x_3 \\ \dot{x}_5 &= 0 \end{cases} \rightarrow y = \begin{pmatrix} \xi_1 + x_1(0) \\ x_2(0) \\ x_2(0)\xi_1 + x_3(0) \\ x_2(0)\frac{\xi_1^2}{2} + x_3(0)\xi_1 + x_4(0) \\ x_5(0) \end{pmatrix}$$

2. : de y en z suivant Y_{211} pendant ξ_{211}

$$\begin{cases} \dot{x}_1 &= 0 \\ \dot{x}_2 &= 0 \\ \dot{x}_3 &= 0 \\ \dot{x}_4 &= -3 \\ \dot{x}_5 &= 0 \end{cases} \rightarrow z = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ -3\xi_{211} + y_4 \\ y_5 \end{pmatrix}$$

3. : de z en t suivant Y_{21} pendant ξ_{21}

$$\begin{cases} \dot{x}_1 = 0 \\ \dot{x}_2 = 0 \\ \dot{x}_3 = -2 \\ \dot{x}_4 = x_1 \\ \dot{x}_5 = -x_2 \end{cases} \rightarrow t = \begin{pmatrix} y_1 \\ y_2 \\ -2\xi_{21} + y_3 \\ y_1\xi_{21} - 3\xi_{21} + y_4 \\ y_2\xi_{21} + y_5 \end{pmatrix}$$

4. : de t en u suivant Y_{221} pendant ξ_{221}

$$\begin{cases} \dot{x}_1 = 0 \\ \dot{x}_2 = 0 \\ \dot{x}_3 = 0 \\ \dot{x}_4 = 0 \\ \dot{x}_5 = 3 \end{cases} \rightarrow u = \begin{pmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ 3\xi_{221} + t_5 \end{pmatrix}$$

5. : de u en $x = x(t)$ suivant Y_2 pendant ξ_2

$$\begin{cases} \dot{x}_1 = 0 \\ \dot{x}_2 = 1 \\ \dot{x}_3 = x_1 \\ \dot{x}_4 = 0 \\ \dot{x}_5 = x_3 \end{cases} \rightarrow v = \begin{pmatrix} t_1 \\ \xi_2 + t_2 \\ t_1\xi_2 + t_3 \\ t_4 \\ t_1\xi_2^2/2 + t_3\xi_2 + 3\xi_{221} + t_5 \end{pmatrix}$$

au final on trouve ainsi :

$$\begin{cases} x_1 = \xi_1 + x_1(0) \\ x_2 = \xi_2 + x_2(0) \\ x_3 = \xi_2(\xi_1 + x_1(0)) - 2\xi_{21} + x_2(0)\xi_1 + x_3(0) \\ x_4 = (\xi_1 + x_1(0))\xi_{21} - 3\xi_{21} + x_2(0)\frac{\xi_2^2}{2} + x_3(0)\xi_1 + x_4(0) \\ x_5 = (\xi_1 + x_1(0))\frac{\xi_2^2}{2} + (-2\xi_{21} + x_2(0)\xi_1 + x_3(0))\xi_2 + \xi_{221} - x_2(0)\xi_{21} + x_5(0) \end{cases}$$

ou simplement en partant de conditions initiales nulles ($x(0) = 0$) :

$$\begin{cases} x_1 = \xi_1 \\ x_2 = \xi_2 \\ x_3 = -2\xi_{12} + \xi_1\xi_2 \\ x_4 = -3\xi_{112} + \xi_1\xi_{12} \\ x_5 = 3\xi_{122} - 2\xi_{12}\xi_2 + \frac{1}{2}\xi_1\xi_2^2 \end{cases}$$

système d'équations polynomiales algébriques qu'il suffit d'inverser pour obtenir l'expression symbolique du changement de coordonnées : des coordonnées de Lydon de la cible en fonction de cette dernière.

On trouve de même le changement de coordonnées pour le système de la pièce (avec des conditions initiales nulles) :

$$\begin{cases} x_1 = \xi_{21} \sin(\xi_1) + \xi_2 \cos(\xi_1) \\ x_2 = -\xi_{21} \cos(\xi_1) + \xi_2 \sin(\xi_1) \\ x_3 = \xi_1 \\ x_4 = \xi_{211} + \xi_2 \end{cases}$$

Chapitre 6

Identification de l'entrée

6.1 Coordonnées exponentielles de l'entrée

L'intérêt des calculs développés ici est de donner une expression formelle des coordonnées exponentielles amenées par une entrée. Le calcul des coordonnées de Lyndon d'un vecteur d'entrée $u(t) = (a(t), b(t))$ revient à intégrer un système d'équations différentielles : le système nilpotent libre d'ordre s (dans le cas d'un système nilpotent à l'ordre s).

6.1.1 Cas des entrées constantes par morceaux

Formules combinatoires des coordonnées exponentielles d'une entrée constante par morceaux

Dans le cas du système $\Sigma NL(4)$ on trouve l'expression combinatoire des coordonnées exponentielles en fonction des paramètres des entrées pour des entrées constantes par morceaux. Pour ces dernières le vecteur $u(t) = (a(t), b(t))$ change de valeur à (N) "tops d'horloge" successifs $T_0 = 0, T_1, \dots, T_N = T$ (T est la date finale). Entre chaque top le vecteur d'entrée est constant sur l'intervalle noté $\Delta_i = [T_{i-1}, T_i]$, où $(i = 1, \dots, N)$:

$$\begin{aligned} u(t)|_{t \in \Delta_i} &= (a_i, b_i) \\ \Delta_i &= [T_{i-1}, T_i] \end{aligned}$$

Il nous faut alors calculer 5 intégrales itérées, $(\xi_1, \xi_2, \xi_{12}, \xi_{112}, \xi_{122})$. On les décompose tout d'abord sur chacun des intervalles qu'elles recouvrent :

$$\xi_i(t)_{t \in \Delta_i} = \int_{\Delta_1} \cdot + \int_{\Delta_2} \cdot + \dots + \int_{\Delta_{i-1}} \cdot + \int_{T_{i-1}}^t \cdot$$

où l est l'un des 5 mots de Širšov . Les calculs suivants établissent les expressions de ces intégrales en fonctions des variables A_i et B_i qui représentent les aires :

$$\begin{aligned} A_i &= |\Delta_i| \times a_i = (t_i - t_{i-1}) \times a_i \\ B_i &= |\Delta_i| \times b_i = (t_i - t_{i-1}) \times b_i \end{aligned}$$

qui serviront par la suite de variables à identifier.

ξ_1 : on calcule d'abord les valeurs intermédiaires :

$$\begin{aligned} \xi_1(t)_{t \in \Delta_i} &= \int_0^t u_1(\tau) d\tau \\ &= \left(\sum_{k=1}^{i-1} \int_{\Delta_k} u_1(\tau) d\tau \right) + \int_{T_{i-1}}^t u_1(\tau) d\tau \\ &= \left(\sum_{k=1}^{i-1} \int_{\Delta_k} a_k d\tau \right) + \int_{T_{i-1}}^t a_i d\tau \\ &= \sum_{k=1}^{i-1} A_k + a_i(t - T_{i-1}) \end{aligned}$$

la valeur finale est donc :

$$\xi(T) = \sum_{i=1}^N A_i$$

ξ_2 : les calculs sont les mêmes il suffit de remplacer les a_i par b_i .

Valeurs intermédiaires :

$$\xi_2(t)_{t \in \Delta_i} = \sum_{k=1}^{i-1} B_k + b_i(t - T_{i-1})$$

la valeur final est donnée par :

$$\xi_2(T) = \sum_{i=1}^N B_i$$

ξ_{21} : valeurs intermédiaires :

$$\begin{aligned}
 \xi_{21}(t)_{t \in \Delta_i} &= \int_0^t u_1(\tau) \xi_2(\tau) d\tau \\
 &= \sum_{k=1}^{i-1} \int_{\Delta_k} u_1(\tau) \left(\sum_{h=1}^{k-1} B_h + b_k(\tau - T_{k-1}) \right) d\tau \\
 &\quad + \int_{T_{i-1}}^t u_1(\tau) \left(\sum_{k=1}^{i-1} B_k + b_i(\tau - T_{i-1}) \right) d\tau \\
 &= \sum_{k=1}^{i-1} a_k \left(\sum_{h=1}^{k-1} B_h(T_k - T_{k-1}) + b_k \frac{(T_k - T_{k-1})^2}{2} \right) \\
 &\quad + a_i \left(\sum_{k=1}^{i-1} B_k(t - T_{i-1}) + b_i \frac{(t - T_{i-1})^2}{2} \right) \\
 &= \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} A_k B_h + \sum_{k=1}^{i-1} \frac{A_k B_k}{2} \\
 &\quad + a_i(t - T_{i-1}) \sum_{k=1}^{i-1} B_k + a_i b_i \frac{(t - T_{i-1})^2}{2}
 \end{aligned}$$

en effet :

$$\int_{T_j}^t \in \Delta_{j+1} (\tau - T_j)^n d\tau = \frac{(t - T_j)^{n+1}}{n+1}$$

la valeur final est donnée par :

$$\xi_{21}(T) = \sum_{i=1}^N \sum_{k=1}^{i-1} A_i B_k + \sum_{i=1}^N \frac{A_i B_i}{2}$$

ξ_{211} : valeurs intermédiaires :

$$\begin{aligned}
 \xi_{211}(t)_{t \in \Delta_i} &= \int_0^t u_1(\tau) \xi_{21}(\tau) d\tau \\
 &= \sum_{k=1}^{i-1} \int_{\Delta_k} u_1(\tau) \left(\sum_{h=1}^{k-1} \sum_{l=1}^{h-1} A_h B_l + \sum_{h=1}^{k-1} \frac{A_h B_h}{2} \right. \\
 &\quad \left. + a_k(\tau - T_{k-1}) \sum_{h=1}^{k-1} B_h + a_k b_k \frac{(\tau - T_{k-1})^2}{2} \right) d\tau \\
 &\quad + \int_{T_{i-1}}^t u_1(\tau) \left(\sum_{k=1}^{i-1} \sum_{h=1}^{k-1} A_k B_h + \sum_{k=1}^{i-1} \frac{A_k B_k}{2} \right. \\
 &\quad \left. + a_i(\tau - T_{i-1}) \sum_{k=1}^{i-1} B_k + a_i b_i \frac{(\tau - T_{i-1})^2}{2} \right) d\tau
 \end{aligned}$$

après intégration :

$$\begin{aligned} \xi_{211}(t)_{t \in \Delta_i} = & \sum_{k=1}^{i-1} a_k \left(\left(\sum_{h=1}^{k-1} \sum_{l=1}^{h-1} A_h B_l + \sum_{h=1}^{k-1} \frac{A_h B_h}{2} \right) (T_k - T_{k-1}) \right. \\ & \left. + a_k \frac{(T_k - T_{k-1})^2}{2} \sum_{h=1}^{k-1} B_h + a_k b_k \frac{(T_k - T_{k-1})^3}{6} \right) \\ & + a_i \left(\left(\sum_{k=1}^{i-1} \sum_{h=1}^{k-1} A_k B_h + \sum_{k=1}^{i-1} \frac{A_k B_k}{2} \right) (t - T_{i-1}) \right. \\ & \left. + a_i \frac{(t - T_{i-1})^2}{2} \sum_{k=1}^{i-1} B_k + a_i b_i \frac{(t - T_{i-1})^3}{6} \right) \end{aligned}$$

après renommage :

$$\begin{aligned} \xi_{211}(t)_{t \in \Delta_i} = & \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} \sum_{l=1}^{h-1} A_k A_h B_l + \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} \frac{A_k A_h B_h}{2} \\ & + \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} \frac{A_k^2 B_h}{2} + \sum_{k=1}^{i-1} \frac{A_k^3}{6} \\ & + a_i (t - T_{i-1}) \left(\sum_{k=1}^{i-1} \sum_{h=1}^{k-1} A_k B_h + \sum_{k=1}^{i-1} \frac{A_k B_k}{2} \right) \\ & + a_i^2 \frac{(t - T_{i-1})^2}{2} \sum_{k=1}^{i-1} B_k + \frac{a_i^2 b_i (t - T_{i-1})^3}{6} \end{aligned}$$

La valeur final est donnée par :

$$\begin{aligned} \xi_{211}(T) = & \sum_{i=1}^N \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} A_i A_k B_h + \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^{i-1} A_i A_k B_k \\ & + \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^{i-1} A_i^2 B_k + \frac{1}{3!} \sum_{i=1}^N A_i^2 B_i \end{aligned}$$

ξ_{221} : valeurs intermédiaires :

$$\begin{aligned}
 \xi_{221}(t)_{t \in \Delta_i} &= \frac{1}{2} \int_0^t u_1(\tau) \xi_2^2(\tau) d\tau \\
 &= \frac{1}{2} \sum_{k=1}^{i-1} \int_{\Delta_k} u_1(\tau) \left(\sum_{h=1}^{k-1} B_h^2 + 2 \sum_{h=1}^{k-1} \sum_{l=1}^{h-1} B_h B_l \right. \\
 &\quad \left. + 2b_k(\tau - T_{k-1}) \sum_{h=1}^{k-1} B_h + b_k^2(\tau - T_{k-1})^2 \right) d\tau \\
 &\quad + \frac{1}{2} \int_{T_i}^t u_1(\tau) \left(\sum_{k=1}^{i-1} B_k^2 + 2 \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} B_k B_h \right. \\
 &\quad \left. + 2b_k(\tau - T_{i-1}) \sum_{k=1}^{i-1} B_k + b_k^2(\tau - T_{i-1})^2 \right) d\tau
 \end{aligned}$$

après intégration :

$$\begin{aligned}
 \xi_{221}(t)_{t \in \Delta_i} &= \frac{1}{2} \sum_{k=1}^{i-1} a_k (T_k - T_{k-1}) \left(\left(\sum_{h=1}^{k-1} B_h^2 + 2 \sum_{h=1}^{k-1} \sum_{l=1}^{h-1} B_h B_l \right) \right. \\
 &\quad \left. + 2b_k \frac{(T_k - T_{k-1})^2}{2} \sum_{h=1}^{k-1} B_h + b_k^2 \frac{(\tau - T_{k-1})^3}{3} \right) \\
 &\quad + \frac{1}{2} a_i \left((t - T_{i-1}) \left(\sum_{k=1}^{i-1} B_k^2 + 2 \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} B_k B_h \right) \right. \\
 &\quad \left. + 2b_k \frac{(t - T_{i-1})^2}{2} \sum_{k=1}^{i-1} B_k + b_k^2 \frac{(\tau - T_{i-1})^3}{3} \right)
 \end{aligned}$$

après distribution et renommage :

$$\begin{aligned}
 \xi_{221}(t)_{t \in \Delta_i} &= \frac{1}{2} \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} A_k B_h^2 + \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} \sum_{l=1}^{h-1} A_k B_h B_l \\
 &\quad + \frac{1}{2} \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} A_k B_k B_h + \frac{1}{6} \sum_{k=1}^{i-1} A_k B_k^2 \\
 &\quad + \frac{1}{2} a_i \left((t - T_{i-1}) \left(\sum_{k=1}^{i-1} B_k^2 + 2 \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} B_k B_h \right) \right. \\
 &\quad \left. + 2b_k \frac{(t - T_{i-1})^2}{2} \sum_{k=1}^{i-1} B_k + b_k^2 \frac{(\tau - T_{i-1})^3}{3} \right)
 \end{aligned}$$

La valeur final est donnée par :

$$\begin{aligned} \xi_{221}(T) = & \sum_{i=1}^N \sum_{k=1}^{i-1} \sum_{h=1}^{k-1} A_i B_k B_h + \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^{i-1} A_i B_k^2 \\ & + \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^{i-1} A_i^2 B_k + \frac{1}{3!} \sum_{i=1}^N A_i B_i^2 \end{aligned}$$

Dans le cas d'entrées constantes par morceaux (N paliers) il suffit donc d'utiliser les formules suivantes :

$$\left\{ \begin{array}{l} \xi_1 = \sum_{i=1}^N A_i \\ \xi_2 = \sum_{i=1}^N B_i \\ \xi_{21} = \frac{1}{2} \sum_{i=1}^N A_i B_i + \sum_{1 \leq k < i \leq N} A_i B_k \\ \xi_{211} = \frac{1}{3!} \sum_{i=1}^N A_i^2 B_i + \frac{1}{2} \sum_{1 \leq k < i \leq N} A_i A_k B_k \\ \quad + \frac{1}{2} \sum_{1 \leq k < i \leq N} A_i^2 B_k + \sum_{1 \leq h < k < i \leq N} A_i A_k B_h \\ \xi_{221} = \frac{1}{3!} \sum_{i=1}^N A_i B_i^2 + \frac{1}{2} \sum_{1 \leq k < i \leq N} A_i B_i B_k \\ \quad + \frac{1}{2} \sum_{1 \leq k < i \leq N} A_i B_k^2 + \sum_{1 \leq h < k < i \leq N} A_i B_k B_h \end{array} \right.$$

Calcul dans le groupe de Lie nilpotent libre

Une autre méthode consiste à utiliser la loi de groupe. Il faut pour cela définir dans un premier temps la concaténation de deux entrées qui consiste à "mettre deux fonctions bout à bout". Plus formellement la concaténation notée $v \# w$ de deux valeurs $v(t) = (v_1, v_2)$, $\forall t \in [0, T_v[$ et $w(t) = (w_1, w_2)$, $\forall t \in [0, T_w[$ du vecteur d'entrée u est définie par :

$$v \# w(T) = \begin{cases} v(t) & \text{si } t \in [0, T_1[\\ w(t - T_1) & \text{si } t \in [T_1, T_2[\end{cases}$$

les coordonnées exponentielles des opérateurs de transport $\mathcal{H}_v, \mathcal{H}_w, \mathcal{H}_{v \# w}$ associés à $v, w, v \# w$ (respectivement) sont alors celles des éléments du groupe nilpotent libre

correspondants :

$$\begin{aligned} & g_v \\ & g_w \\ g_{v\#w} &= g_v \circ g_w \end{aligned}$$

Par ailleurs dans le cas d'une entrée *constante* $u = (a, b)$, la trajectoire dans l'espace d'état est simplement la courbe intégrale du champ de vecteur $aX_1 + bX_2$. Les coordonnées exponentielles du point atteint au temps T sont donc celles de l'exponentielle :

$$\begin{aligned} g_u &= \exp\left(\left(\int_0^t d\tau\right) (a[1] + b[2])\right) \\ &= \exp(A[1] + B[2]) \end{aligned}$$

où $A = aT$ et $B = bT$

Dans le cas d'une entrée constante par morceaux $u(t) = (a_i, b_i)$ où $t \in [T_{i-1}, T_i]$, $i = 1, \dots, N$ on peut donc se ramener au calcul du produit suivant :

$$g_u = e^{A_1[1]+B_1[2]} e^{A_2[1]+B_2[2]} \dots e^{A_N[1]+B_N[2]}$$

—où $A_i = a_i(t_i - t_{i-1})$ and $B_i = b_i(t_i - t_{i-1})$ —

6.1.2 Cas des entrées polynomiales

Le calcul d'intégrales itérées pour des fonctions polynomiales ne pose a priori aucune difficulté. Notons simplement que chaque coordonnée exponentielle associée est un polynôme homogène quand la variable (T) a reçu une valeur (*i.e.* un polynôme homogène sur les paramètres).

6.1.3 Cas des entrées difficiles à intégrer formellement

Il va de soi qu'on préférera obtenir des coordonnées exponentielles symboliques plutôt que scalaires (du symbolique on pourra toujours revenir au numérique). Cependant les coordonnées exponentielles d'un point de l'espace d'état de certains systèmes ne pourront être obtenues que par la méthode des systèmes étendus (Sussmann). Or en toute généralité cette méthode implique le calcul d'intégrales itérées sur des fonctions liées aux champs de vecteurs initiaux du système, on ne saura donc les intégrer symboliquement que dans des cas particuliers (fonctions polynomiales, trigonométriques, exponentielles).

6.2 Le système de 5 équations

6.2.1 Méthode de Gröbner

Comme le montre les résultats qui suivent, la méthode de Gröbner ne peut pas permettre d'aller très loin dans la résolution de ce type de problème car les bases calculées explosent assez vite. Par ailleurs la solution qu'elles offrent est dans le cas général un polynôme qui nécessite une discussion supplémentaire pour savoir s'il possède des solutions réelles.

Les premiers calculs montrent un cas particulièrement simple où une solution linéaire générique peut être dégagée pour des entrées constantes par morceaux. Il n'en va pas de même pour la suivante calculée toujours avec les 5 équations de $\Sigma NL(4)$ mais pour des entrées polynomiales cette fois-ci.

Une solution générique en 3 paliers constants

Solution générique aux 5 équations du système nilpotent libre à l'ordre 3.

Entrées constantes par morceaux : 3 paliers.

Méthode de Gröbner.

On génère les équations

```

> N := 3;
>
> ##### Coordonnees de l entree :
> x1 := sum( A[i] ,i=1..N);
> x2 := sum( B[i] ,i=1..N);
> x21 := 1/2*sum( B[i]*A[i] ,i=1..N) + sum( B[i] * sum( A[j] ,j=i+1..N)
> ,i=1..N-1);
> x211 := 1/6*sum( B[i]*A[i]^2 ,i=1..N)
> + 1/2*sum( B[i] * sum( A[j]^2 ,j=i+1..N) ,i=1..N-1)
> + 1/2*sum( B[i] * A[i] * sum( A[j] ,j=i+1..N) ,i=1..N-1)
> + sum( B[i] * sum( A[j] * sum( A[k] ,k=j+1..N) ,j=i+1..N-1)
> , i=1..N-2);
> x221 := 1/6*sum( B[i]^2*A[i] ,i=1..N)
> + 1/2*sum( B[i]^2 * sum( A[j] ,j=i+1..N) ,i=1..N-1)
> + 1/2*sum( B[i] * sum( B[j]*A[j] ,j=i+1..N) ,i=1..N-1)
> + sum( B[i] * sum( B[j] * sum( A[k] ,k=j+1..N) ,j=i+1..N-1)
> , i=1..N-2);
>
>
> ##### les equations :
> f1 := k1 - x1;
> f2 := k2 - x2;
> f21:= 2*k21 - 2*x21;
> f211 := 6*k211 - 6*x211;
> f221 := 6*k221 - 6*x221;
>

```


>

$$N := 3$$

$$x1 := A_1 + A_2 + A_3$$

$$x2 := B_1 + B_2 + B_3$$

$$x21 := \frac{1}{2} B_1 A_1 + \frac{1}{2} B_2 A_2 + \frac{1}{2} B_3 A_3 + B_1 (A_2 + A_3) + B_2 A_3$$

$$x211 := \frac{1}{6} B_1 A_1^2 + \frac{1}{6} B_2 A_2^2 + \frac{1}{6} B_3 A_3^2 + \frac{1}{2} B_1 (A_2^2 + A_3^2) + \frac{1}{2} B_2 A_3^2 \\ + \frac{1}{2} B_1 A_1 (A_2 + A_3) + \frac{1}{2} B_2 A_2 A_3 + B_1 A_2 A_3$$

$$x221 := \frac{1}{6} B_1^2 A_1 + \frac{1}{6} B_2^2 A_2 + \frac{1}{6} B_3^2 A_3 + \frac{1}{2} B_1^2 (A_2 + A_3) + \frac{1}{2} B_2^2 A_3 \\ + \frac{1}{2} B_1 (B_2 A_2 + B_3 A_3) + \frac{1}{2} B_2 B_3 A_3 + B_1 B_2 A_3$$

$$f1 := k1 - A_1 - A_2 - A_3$$

$$f2 := k2 - B_1 - B_2 - B_3$$

$$f21 := 2k21 - B_1 A_1 - B_2 A_2 - B_3 A_3 - 2B_1 (A_2 + A_3) - 2B_2 A_3$$

$$f211 := 6k211 - B_1 A_1^2 - B_2 A_2^2 - B_3 A_3^2 - 3B_1 (A_2^2 + A_3^2) - 3B_2 A_3^2 \\ - 3B_1 A_1 (A_2 + A_3) - 3B_2 A_2 A_3 - 6B_1 A_2 A_3$$

$$f221 := 6k221 - B_1^2 A_1 - B_2^2 A_2 - B_3^2 A_3 - 3B_1^2 (A_2 + A_3) - 3B_2^2 A_3 \\ - 3B_1 (B_2 A_2 + B_3 A_3) - 3B_2 B_3 A_3 - 6B_1 B_2 A_3$$

On élimine A1 et B1 des deux équations linéaires

```
> z1 := f1 + A[1] : A[1] := z1 :
> z2 := f2 + B[1] : B[1] := z2 :
>
```

On calcule la base de Gröbner des 3 équations restantes :

```
> with (grobner,gbasis) :
>
> equations := [ f21, f211, f221 ] :
> variables := [ seq(A[i], i=2..N) , seq(B[i],i=2..N)] ;
>
> base := gbasis(equations,variables):
>
> for e in base do
>     print('-----'):
>     e;
> od;
>
```

$variables := [A_2, A_3, B_2, B_3]$

$$\begin{aligned}
 & (-8k^2k_1^2k_{21} + 16k_1k_{21}^2k_2 - 24k_{21}k_2k_{211} + k^3k_1^3 \\
 & \quad + 6k^2k_1k_{211} + 6k_2k_1^2k_{221} - 24k_1k_{21}k_{221} + 36k_{221}k_{211} \\
 &)B_2 + (-8k^2k_1^2k_{21} + 12k_1k_{21}^2k_2 - 12k_{21}k_2k_{211} + k^3k_1^3 \\
 & \quad + 6k^2k_1k_{211} + 6k_2k_1^2k_{221} - 12k_1k_{21}k_{221})B_3 + (\\
 & 8k^2k_{21}^2 - 2k^3k_{21}k_1 - 36k_2k_{21}k_{221} + 6k^2k_1k_{221} \\
 & \quad + 36k_{221}^2)A_2 - 16k^2k_1k_{21}^2 + 8k^3k_1^2k_{21} - 12k^2k_{211}k_{21} \\
 & \quad - k^4k_1^3 - 6k_1^2k_{221}k_2^2 + 36k_1k_{221}^2 + 36k_2k_{211}k_{221} \\
 & \quad + 24k_{21}^3k_2 - 48k_{21}^2k_{221}
 \end{aligned}$$

$$\begin{aligned}
 & (6k_{211} - 2k_1k_{21})B_3 + (-4k_2k_{21} + k^2k_1 + 6k_{221})A_3 - 6k_2k_{211} \\
 & \quad + 2k_2k_1k_{21} - 6k_1k_{221} + 4k_{21}^2
 \end{aligned}$$

$$\begin{aligned}
& (-4k_2k_{21}^2 + 8k_2^2k_1k_{21} + 12k_{221}k_{21} - k_2^3k_1^2 - 12k_2k_1k_{221} \\
& \quad - 6k_2^2k_{211})B_2 + (14k_2^2k_1k_{21} - 2k_2^3k_1^2 - 18k_2k_1k_{221} \\
& \quad - 12k_2k_{21}^2 + 24k_{221}k_{21} - 6k_2^2k_{211})B_3 \\
& \quad + (-6k_2k_1k_{21} + k_2^2k_1^2 + 6k_1k_{221} + 6k_2k_{211})B_2B_3 \\
& \quad + (-6k_2k_1k_{21} + k_2^2k_1^2 + 6k_1k_{221} + 6k_2k_{211})B_3^2 \\
& \quad + 16k_2^2k_{21}^2 - 8k_2^3k_{21}k_1 - 48k_2k_{21}k_{221} + k_2^4k_1^2 \\
& \quad + 12k_2^2k_1k_{221} + 36k_{221}^2
\end{aligned}$$

En prenant B_3 comme paramètre on rend le système linéaire. On calcule donc une base pour les 3 autres variables.

```

> variables := [ A[2], A[3], B[2] ];
>
> base := gbasis(equations,variables):
>
> for e in base do
>   print('-----'):
>   e;
> od;
>
>

```

$variables := [A_2, A_3, B_2]$

$$\begin{aligned}
& (-4k_2k_{21} + k_2^2k_1 + 6k_{221})A_3 + 6B_3k_{211} - 6k_2k_{211} + 2k_2k_1k_{21} \\
& \quad + 4k_{21}^2 - 2B_3k_1k_{21} - 6k_1k_{221}
\end{aligned}$$

$$\begin{aligned}
& -6k_2^4k_1^3k_{21} + 36k_2^3k_1^2k_{21}^2 - 36k_1^2k_{221}^2B_3 - 216k_1k_{221}^2k_{21} \\
& \quad - 48k_2B_3k_{211}k_{21}^2 + 36k_2B_3^2k_{211}^2 + 36k_2^3k_{211}^2 \\
& \quad - 48k_2k_{21}^4 + 216k_{211}k_{221}^2 + 12k_2^3k_1^3k_{221} \\
& \quad - 432k_2k_{211}k_{221}k_{21} + 144k_2^2k_{211}k_{21}^2 + 18k_2^4k_{211}k_1^2 \\
& \quad - 18k_2^3B_3k_{211}k_1^2 - 144k_2^3k_{211}k_1k_{21} + 6k_2^3k_1^3k_{21}B_3 \\
& \quad - 2k_2^2k_{21}B_3^2k_1^3 + 216k_2^2k_{211}k_1k_{221} - 144k_2^2k_1^2k_{21}k_{221} \\
& \quad - 6k_2^2k_1^3k_{221}B_3 - 180k_2B_3k_{211}k_1k_{221} \\
& \quad + 84k_2k_1^2k_{21}k_{221}B_3 + 144k_2k_1k_{21}^2k_{221} \\
& \quad - 12k_{21}B_3^2k_1^2k_{221} - 24k_{21}^2k_{221}B_3k_1 + 36B_3^2k_{211}k_{221}k_1 \\
& \quad + 144B_3k_{211}k_{221}k_{21} + 96k_{21}^3k_{221} + 6k_2^2B_3^2k_{211}k_1^2
\end{aligned}$$

$$\begin{aligned}
& - 36 k^2 k_1^2 k_{21}^2 B_3 + 144 k^2 B_3 k_{211} k_1 k_{21} \\
& - 48 k_2 B_3^2 k_{211} k_{21} k_1 + 12 k_2 k_{21}^2 B_3^2 k_1^2 + 108 k_2 k_1^2 k_{221}^2 \\
& - 72 k^2 B_3 k_{211}^2 + (k_1^3 k_2^5 - k^4 B_3 k_1^3 + 6 k^4 k_{211} k_1 \\
& - 12 k^4 k_1^2 k_{21} - 6 k^3 B_3 k_{211} k_1 + 10 k^3 B_3 k_1^2 k_{21} \\
& + 36 k^3 k_1 k_{21}^2 - 24 k^3 k_{211} k_{21} + 18 k^3 k_1^2 k_{221} \\
& - 16 k^2 k_{21}^3 - 12 k^2 B_3 k_1^2 k_{221} - 108 k^2 k_1 k_{21} k_{221} \\
& - 24 k^2 B_3 k_1 k_{21}^2 + 24 k^2 B_3 k_{211} k_{21} + 36 k^2 k_{221} k_{211} \\
& + 60 k_2 B_3 k_1 k_{21} k_{221} + 72 k_2 k_{21}^2 k_{221} + 72 k_2 k_1 k_{221}^2 \\
& - 36 k_2 B_3 k_{211} k_{221} - 72 k_{221}^2 k_{21} - 36 B_3 k_1 k_{221}^2) A_2
\end{aligned}$$

$$\begin{aligned}
& (- 8 k^2 k_1 k_{21} + k^3 k_1^2 - 6 k_2 B_3 k_{211} + 6 k^2 k_{211} + 4 k_2 k_{21}^2 \\
& + 6 k_2 k_{21} B_3 k_1 + 12 k_2 k_1 k_{221} - k^2 k_1^2 B_3 - 6 k_{221} B_3 k_1 \\
& - 12 k_{221} k_{21}) B_2 + 8 k^3 k_{21} k_1 - 16 k^2 k_{21}^2 - 14 k^2 k_{21} B_3 k_1 \\
& - k^4 k_1^2 + 2 k^3 k_1^2 B_3 - 12 k^2 k_1 k_{221} + 48 k_2 k_{21} k_{221} \\
& + 18 k_2 k_{221} B_3 k_1 - 36 k_{221}^2 + 6 B_3^2 k_1 k_2 k_{21} - B_3^2 k_1^2 k_2^2 \\
& - 6 B_3^2 k_1 k_{221} + 12 k_{21}^2 B_3 k_2 - 24 k_{21} B_3 k_{221} - 6 k_2 B_3^2 k_{211} \\
& + 6 k^2 B_3 k_{211}
\end{aligned}$$

On en déduit les expressions de A2 , A3 et B2 par simple division; A1 et B1 sont alors complètement déterminés, B3 reste libre.

```
> degree(base[1], A[3]);
```

```
1
```

```
> Q := quo( base[3], B[2], B[2], 'R' ): B[2] := - Q/R :
> Q := quo( base[2], A[2], A[2], 'R' ): A[2] := - Q/R :
> Q := quo( base[1], A[3], A[3], 'R' ): A[3] := - Q/R :
>
>
```

```
> A[1];
```

$$\begin{aligned}
& k_1 + (k_1^3 k_2^5 - k^4 B_3 k_1^3 + 6 k^4 k_{211} k_1 - 12 k^4 k_1^2 k_{21} \\
& - 6 k^3 B_3 k_{211} k_1 + 10 k^3 B_3 k_1^2 k_{21} + 36 k^3 k_1 k_{21}^2 \\
& - 24 k^3 k_{211} k_{21} + 18 k^3 k_1^2 k_{221} - 16 k^2 k_{21}^3 \\
& - 12 k^2 B_3 k_1^2 k_{221} - 108 k^2 k_1 k_{21} k_{221} - 24 k^2 B_3 k_1 k_{21}^2
\end{aligned}$$

$$\begin{aligned}
& + 24 k^2 B_3 k_{211} k_{21} + 36 k^2 k_{221} k_{211} + 60 k_2 B_3 k_1 k_{21} k_{221} \\
& + 72 k_2 k_{21}^2 k_{221} + 72 k_2 k_1 k_{221}^2 - 36 k_2 B_3 k_{211} k_{221} \\
& - 72 k_{221}^2 k_{21} - 36 B_3 k_1 k_{221}^2) / (-6 k^2 k_1^3 k_{21} \\
& + 36 k^2 k_1^2 k_{21}^2 - 36 k_1^2 k_{221}^2 B_3 - 216 k_1 k_{221}^2 k_{21} \\
& - 48 k_2 B_3 k_{211} k_{21}^2 + 36 k_2 B_3^2 k_{211}^2 + 36 k^2 k_{211}^2 \\
& - 48 k_2 k_{21}^4 + 216 k_{211} k_{221}^2 + 12 k^2 k_1^3 k_{221} \\
& - 432 k_2 k_{211} k_{221} k_{21} + 144 k^2 k_{211} k_{21}^2 + 18 k^2 k_{211} k_1^2 \\
& - 18 k^2 B_3 k_{211} k_1^2 - 144 k^2 k_{211} k_1 k_{21} + 6 k^2 k_1^3 k_{21} B_3 \\
& - 2 k^2 k_{21} B_3^2 k_1^3 + 216 k^2 k_{211} k_1 k_{221} - 144 k^2 k_1^2 k_{21} k_{221} \\
& - 6 k^2 k_1^3 k_{221} B_3 - 180 k_2 B_3 k_{211} k_1 k_{221} \\
& + 84 k_2 k_1^2 k_{21} k_{221} B_3 + 144 k_2 k_1 k_{21}^2 k_{221} \\
& - 12 k_{21} B_3^2 k_1^2 k_{221} - 24 k_{21}^2 k_{221} B_3 k_1 + 36 B_3^2 k_{211} k_{221} k_1 \\
& + 144 B_3 k_{211} k_{221} k_{21} + 96 k_{21}^3 k_{221} + 6 k^2 B_3^2 k_{211} k_1^2 \\
& - 36 k^2 k_1^2 k_{21}^2 B_3 + 144 k^2 B_3 k_{211} k_1 k_{21} \\
& - 48 k_2 B_3^2 k_{211} k_{21} k_1 + 12 k_2 k_{21}^2 B_3^2 k_1^2 + 108 k_2 k_1^2 k_{221}^2 \\
& - 72 k^2 B_3 k_{211}^2) + (-4 k_2 k_{21} + k^2 k_1 + 6 k_{221}) / (6 B_3 k_{211} \\
& - 6 k_2 k_{211} + 2 k_2 k_1 k_{21} + 4 k_{21}^2 - 2 B_3 k_1 k_{21} - 6 k_1 k_{221})
\end{aligned}$$

> B[1];

$$\begin{aligned}
& k_2 + (-8 k^2 k_1 k_{21} + k^2 k_1^2 - 6 k_2 B_3 k_{211} + 6 k^2 k_{211} + 4 k_2 k_{21}^2 \\
& + 6 k_2 k_{21} B_3 k_1 + 12 k_2 k_1 k_{221} - k^2 k_1^2 B_3 - 6 k_{221} B_3 k_1 \\
& - 12 k_{221} k_{21}) / (8 k^2 k_{21} k_1 - 16 k^2 k_{21}^2 - 14 k^2 k_{21} B_3 k_1 \\
& - k^2 k_1^2 + 2 k^2 k_1^2 B_3 - 12 k^2 k_1 k_{221} + 48 k_2 k_{21} k_{221} \\
& + 18 k_2 k_{221} B_3 k_1 - 36 k_{221}^2 + 6 B_3^2 k_1 k_2 k_{21} - B_3^2 k_1^2 k_{21}^2 \\
& - 6 B_3^2 k_1 k_{221} + 12 k_{21}^2 B_3 k_2 - 24 k_{21} B_3 k_{221} - 6 k_2 B_3^2 k_{211} \\
& + 6 k^2 B_3 k_{211}) - B_3
\end{aligned}$$

> A[2];

$$\begin{aligned}
& - (k_1^3 k_2^5 - k^2 B_3 k_1^3 + 6 k^2 k_{211} k_1 - 12 k^2 k_1^2 k_{21} \\
& - 6 k^2 B_3 k_{211} k_1 + 10 k^2 B_3 k_1^2 k_{21} + 36 k^2 k_1 k_{21}^2 \\
& - 24 k^2 k_{211} k_{21} + 18 k^2 k_1^2 k_{221} - 16 k^2 k_{21}^3 \\
& - 12 k^2 B_3 k_1^2 k_{221} - 108 k^2 k_1 k_{21} k_{221} - 24 k^2 B_3 k_1 k_{21}^2 \\
& + 24 k^2 B_3 k_{211} k_{21} + 36 k^2 k_{221} k_{211} + 60 k_2 B_3 k_1 k_{21} k_{221}
\end{aligned}$$

$$\begin{aligned}
& + 72 k_2 k_{21}^2 k_{221} + 72 k_2 k_1 k_{221}^2 - 36 k_2 B_3 k_{211} k_{221} \\
& - 72 k_{221}^2 k_{21} - 36 B_3 k_1 k_{221}^2) / (-6 k_2^4 k_1^3 k_{21} \\
& + 36 k_2^3 k_1^2 k_{21}^2 - 36 k_1^2 k_{221}^2 B_3 - 216 k_1 k_{221}^2 k_{21} \\
& - 48 k_2 B_3 k_{211} k_{21}^2 + 36 k_2 B_3^2 k_{211}^2 + 36 k_2^3 k_{211}^2 \\
& - 48 k_2 k_{21}^4 + 216 k_{211} k_{221}^2 + 12 k_2^3 k_1^3 k_{221} \\
& - 432 k_2 k_{211} k_{221} k_{21} + 144 k_2^2 k_{211} k_{21}^2 + 18 k_2^4 k_{211} k_1^2 \\
& - 18 k_2^3 B_3 k_{211} k_1^2 - 144 k_2^3 k_{211} k_1 k_{21} + 6 k_2^3 k_1^3 k_{21} B_3 \\
& - 2 k_2^2 k_{21} B_3^2 k_1^3 + 216 k_2^2 k_{211} k_1 k_{221} - 144 k_2^2 k_1^2 k_{21} k_{221} \\
& - 6 k_2^2 k_1^3 k_{221} B_3 - 180 k_2 B_3 k_{211} k_1 k_{221} \\
& + 84 k_2 k_1^2 k_{21} k_{221} B_3 + 144 k_2 k_1 k_{21}^2 k_{221} \\
& - 12 k_{21} B_3^2 k_1^2 k_{221} - 24 k_{21}^2 k_{221} B_3 k_1 + 36 B_3^2 k_{211} k_{221} k_1 \\
& + 144 B_3 k_{211} k_{221} k_{21} + 96 k_{21}^3 k_{221} + 6 k_2^2 B_3^2 k_{211} k_1^2 \\
& - 36 k_2^2 k_1^2 k_{21}^2 B_3 + 144 k_2^2 B_3 k_{211} k_1 k_{21} \\
& - 48 k_2 B_3^2 k_{211} k_{21} k_1 + 12 k_2 k_{21}^2 B_3^2 k_1^2 + 108 k_2 k_1^2 k_{221}^2 \\
& - 72 k_2^2 B_3 k_{211}^2)
\end{aligned}$$

> B[2];

$$\begin{aligned}
& - (-8 k_2^2 k_1 k_{21} + k_2^3 k_1^2 - 6 k_2 B_3 k_{211} + 6 k_2^2 k_{211} + 4 k_2 k_{21}^2 \\
& + 6 k_2 k_{21} B_3 k_1 + 12 k_2 k_1 k_{221} - k_2^2 k_1^2 B_3 - 6 k_{221} B_3 k_1 \\
& - 12 k_{221} k_{21}) / (8 k_2^3 k_{21} k_1 - 16 k_2^2 k_{21}^2 - 14 k_2^2 k_{21} B_3 k_1 \\
& - k_2^4 k_1^2 + 2 k_2^3 k_1^2 B_3 - 12 k_2^2 k_1 k_{221} + 48 k_2 k_{21} k_{221} \\
& + 18 k_2 k_{221} B_3 k_1 - 36 k_{221}^2 + 6 B_3^2 k_1 k_2 k_{21} - B_3^2 k_1^2 k_2^2 \\
& - 6 B_3^2 k_1 k_{221} + 12 k_{21}^2 B_3 k_2 - 24 k_{21} B_3 k_{221} - 6 k_2 B_3^2 k_{211} \\
& + 6 k_2^2 B_3 k_{211})
\end{aligned}$$

> A[3];

$$\begin{aligned}
& - (-4 k_2 k_{21} + k_2^2 k_1 + 6 k_{221}) / (6 B_3 k_{211} - 6 k_2 k_{211} + 2 k_2 k_1 k_{21} \\
& + 4 k_{21}^2 - 2 B_3 k_1 k_{21} - 6 k_1 k_{221})
\end{aligned}$$

> B[3];

B_3

Base de Gröbner pour des polynômes de degré 2

```

> p1 := a0 + a1*t + a2*t^2 ;
> p2 := b0 + b1*t + b2*t^2 ;
>
> define( u1, forall(t,u1(t)=p1)):
> define( u2, forall(t,u2(t)=p2)):
>
> x1 := integrate( p1 ,t):
> x2 := integrate( p2 ,t):
> x21 := normal( integrate( x2*p1 ,t) ):
> x211 := normal( integrate( x21*p1 ,t) ):
> x221 := normal( integrate( 1/2*x2^2*p1 ,t) ):
>
> t := 1:
>
> f1 := k1 - x1 ;
> f2 := k2 - x2 ;
> f21 := k21 - x21 ;
> f211 := k211 - x211 ;
> f221 := k221 - x221 ;
>
>
p1 := a0 + a1 t + a2 t^2

```

$$p2 := b0 + b1 t + b2 t^2$$

$$f1 := k1 - a0 - \frac{1}{2} a1 - \frac{1}{3} a2$$

$$f2 := k2 - b0 - \frac{1}{2} b1 - \frac{1}{3} b2$$

$$f21 := k21 - \frac{1}{18} b2 a2 - \frac{1}{10} b1 a2 - \frac{1}{15} b2 a1 - \frac{1}{4} b0 a2 - \frac{1}{8} b1 a1 - \frac{1}{12} b2 a0 \\ - \frac{1}{3} b0 a1 - \frac{1}{6} b1 a0 - \frac{1}{2} b0 a0$$

$$f211 := k211 - \frac{1}{162} b2 a2^2 - \frac{1}{80} b1 a2^2 - \frac{11}{720} b2 a2 a1 - \frac{5}{252} b2 a2 a0 \\ - \frac{9}{280} a2 b1 a1 - \frac{1}{28} b0 a2^2 - \frac{1}{105} b2 a1^2 - \frac{7}{72} a2 b0 a1 \\ - \frac{2}{45} a2 b1 a0 - \frac{1}{40} a1 b2 a0 - \frac{1}{48} b1 a1^2 - \frac{3}{20} b0 a0 a2 - \frac{1}{15} b0 a1^2 \\ - \frac{7}{120} a1 b1 a0 - \frac{1}{60} b2 a0^2 - \frac{5}{24} b0 a0 a1 - \frac{1}{24} b1 a0^2 - \frac{1}{6} b0 a0^2$$

$$\begin{aligned}
 f_{221} := & k_{221} - \frac{1}{162} b_2^2 a_2 - \frac{1}{48} b_1 b_2 a_2 - \frac{1}{144} b_2^2 a_1 - \frac{1}{21} a_2 b_0 b_2 \\
 & - \frac{1}{56} a_2 b_1^2 - \frac{1}{42} b_1 b_2 a_1 - \frac{1}{126} b_2^2 a_0 - \frac{1}{12} b_0 b_1 a_2 - \frac{1}{18} a_1 b_0 b_2 \\
 & - \frac{1}{48} a_1 b_1^2 - \frac{1}{36} b_1 b_2 a_0 - \frac{1}{10} b_0^2 a_2 - \frac{1}{10} b_0 b_1 a_1 - \frac{1}{15} a_0 b_0 b_2 \\
 & - \frac{1}{40} a_0 b_1^2 - \frac{1}{8} b_0^2 a_1 - \frac{1}{8} b_0 b_1 a_0 - \frac{1}{6} b_0^2 a_0
 \end{aligned}$$

```

> z0 := f1 + a0 : a0 := z0 ;
> z0 := f2 + b0 : b0 := z0 ;
>
>

```

$$a_0 := k_1 - \frac{1}{2} a_1 - \frac{1}{3} a_2$$

$$b_0 := k_2 - \frac{1}{2} b_1 - \frac{1}{3} b_2$$

```

> with (grobner,gbasis) :
>
> equations := [ f21, f211, f221 ] :
> variables := [ a1, a2, b1, b2];
>
> base := gbasis(equations,variables):
>

```

```
variables := [a1,a2,b1,b2]
```

La base de Gröbner comporte 6 polynômes :

```

> for e in base do
>   print('-----');
>   e;
> od;
>
>

```

$$\begin{aligned}
& -2k_2k_1b_2a_2 + k_2^2a_2^2 + (-405k_2^2k_1 + 810k_2k_{21})a_1 \\
& + (-510k_2^2k_1 + 1440k_2k_{21} - 1260k_{221})a_2 \\
& + (405k_2k_1^2 - 810k_1k_{21})b_1 \\
& + (300k_2k_1^2 - 1260k_{211} - 180k_1k_{21})b_2 + k_1^2b_2^2 \\
& + 37800k_2k_{211} + 37800k_1k_{221} - 1800k_2^2k_1^2 \\
& - 5400k_1k_2k_{21} - 32400k_{21}^2
\end{aligned}$$

$$\begin{aligned}
& (774k_2^3k_1^3 + 4536k_2^2k_1k_{211} - 4572k_{21}k_2^2k_1^2 + 4536k_2k_1^2k_{221}) \\
& b_2a_1 + (382k_2^3k_1^3 + 2184k_2^2k_1k_{211} - 2220k_{21}k_2^2k_1^2 \\
& + 2184k_2k_1^2k_{221})b_2a_2 + (-47628k_2^4k_1^3 \\
& + 197316k_2^3k_1^2k_{21} + 192780k_2^2k_1^2k_{221} \\
& - 283500k_2^3k_{211}k_1 - 521640k_2^2k_{211}k_{21} \\
& + 2857680k_{221}k_2k_{211} + 521640k_1k_2^2k_{21}^2 \\
& - 3379320k_2k_1k_{221}k_{21} + 2857680k_1k_{221}^2)a_1 + (\\
& -61992k_2^4k_1^3 + 395640k_2^3k_1^2k_{21} - 140616k_2^2k_1^2k_{221} \\
& - 369432k_2^3k_{211}k_1 + 151200k_2^2k_{211}k_{21} \\
& + 1375920k_{221}k_2k_{211} - 151200k_1k_2^2k_{21}^2 \\
& - 1224720k_2k_1k_{221}k_{21} + 1375920k_1k_{221}^2)a_2 + (\\
& 132678k_2^3k_1^4 - 1195236k_2^2k_1^3k_{21} + 1281420k_2^2k_1^2k_{211} \\
& - 5284440k_2k_{211}k_1k_{21} + 2857680k_2k_{211}^2 \\
& + 2404080k_1^2k_2k_{21}^2 + 782460k_1^3k_{221}k_2 \\
& - 2381400k_1^2k_{221}k_{21} + 2857680k_1k_{221}k_{211})b_1 + (\\
& 106116k_2^3k_1^4 - 901584k_2^2k_1^3k_{21} + 876456k_2^2k_1^2k_{211} \\
& - 2993760k_2k_{211}k_1k_{21} + 1375920k_2k_{211}^2 \\
& + 1594080k_1^2k_2k_{21}^2 + 624960k_1^3k_{221}k_2 \\
& - 1572480k_1^2k_{221}k_{21} + 1375920k_1k_{221}k_{211})b_2 + (\\
& 189k_2^4k_1^2 + 1134k_2^3k_{211} - 1134k_1k_2^3k_{21} + 1134k_2^2k_1k_{221} \\
&)a_1^2 + (\\
& -396k_1^4k_2^2 - 2268k_1^2k_2k_{211} + 2304k_1^3k_2k_{21} - 2268k_1^3k_{221} \\
&)b_1b_2 + 4936680k_2^3k_1^2k_{211} + 4830840k_2^2k_1^3k_{221} \\
& - 419580k_2^4k_1^4 + 1580040k_2^3k_1^3k_{21} - 635040k_2^2k_1^2k_{21}^2 \\
& + 45087840k_2^2k_{211}^2 + 88905600k_2k_{211}k_1k_{221} \\
& - 50621760k_2^2k_{211}k_1k_{21} - 38102400k_2k_{211}k_{21}^2 \\
& - 49986720k_1^2k_2k_{21}k_{221} + 38102400k_1k_2k_{21}^3 \\
& + 44452800k_1^2k_{221}^2 - 38102400k_1k_{221}k_{21}^2 + (\\
& -200k_1^4k_2^2 - 1092k_1^2k_2k_{211} + 1128k_1^3k_2k_{21} - 1092k_1^3k_{221} \\
&)b_2^2
\end{aligned}$$

$$\begin{aligned}
& -2k_2k_1b_2a_1 + (330k_2^2k_1 - 180k_2k_{21} - 1260k_{221})a_1 \\
& + (420k_2^2k_1 - 780k_2k_{21})a_2 \\
& + (-540k_2k_1^2 - 1260k_{211} + 1440k_1k_{21})b_1 \\
& + (-420k_2k_1^2 + 780k_1k_{21})b_2 + k_2^2a_2a_1 + k_1^2b_1b_2 \\
& - 37800k_2k_{211} - 37800k_1k_{221} + 1980k_2^2k_1^2 \\
& + 5040k_1k_2k_{21} + 32400k_{21}^2
\end{aligned}$$

$$\begin{aligned}
& (18k_2^3k_1 - 36k_2^2k_{21})b_2a_1 + (18k_2^3k_1 - 36k_2^2k_{21})b_2a_2 \\
& + (756k_2^3k_{21} - 378k_2^4k_1)a_1 \\
& + (1008k_2^3k_{21} - 420k_2^4k_1 - 504k_2^2k_{221})a_2 + (\\
& -22680k_2k_1k_{221} - 1512k_2^3k_1^2 + 14364k_1k_2^2k_{21} \\
& - 22680k_2k_{21}^2 + 45360k_{221}k_{21})b_1 + (3024k_2^2k_{211} \\
& - 19152k_2k_1k_{221} + 12168k_1k_2^2k_{21} - 1236k_2^3k_1^2 \\
& - 23760k_2k_{21}^2 + 45360k_{221}k_{21})b_2 + \\
& (360k_2^2k_1^2 - 2232k_1k_2k_{21} + 2268k_2k_{211} + 2268k_1k_{221})b_1 \\
& b_2 + \\
& (189k_2^2k_1^2 + 1134k_2k_{211} - 1134k_1k_2k_{21} + 1134k_1k_{221}) \\
& b_1^2 + \\
& (164k_2^2k_1^2 - 1056k_1k_2k_{21} + 1092k_2k_{211} + 1092k_1k_{221}) \\
& b_2^2 - 52920k_2^3k_{211} + 52920k_2^2k_1k_{221} + 1260k_2^4k_1^2 \\
& - 22680k_1k_2^3k_{21} + 181440k_2^2k_{21}^2 - 635040k_{221}k_2k_{21} \\
& + 635040k_{221}^2
\end{aligned}$$

$$\begin{aligned}
& b_1a_2 - b_2a_1 + 30a_1k_2 + 30a_2k_2 - 30b_1k_1 - 30b_2k_1 - 360k_2 \\
& + 180k_2k_1
\end{aligned}$$

$$\begin{aligned}
& (189 k_2^2 k_1^2 + 1134 k_2 k_{211} - 1134 k_1 k_2 k_{21} + 1134 k_1 k_{221}) b_1 a_1 + \\
& (396 k_2^2 k_1^2 + 2268 k_2 k_{211} - 2304 k_1 k_2 k_{21} + 2268 k_1 k_{221}) b_2 \\
& a_1 + \\
& (200 k_2^2 k_1^2 + 1092 k_2 k_{211} - 1128 k_1 k_2 k_{21} + 1092 k_1 k_{221}) b_2 \\
& a_2 + (-6048 k_2^3 k_1^2 - 34020 k_2^2 k_{211} + 34776 k_1 k_2^2 k_{21} \\
& - 34020 k_2 k_1 k_{221}) a_1 + (-5796 k_2^3 k_1^2 - 32256 k_2^2 k_{211} \\
& + 33264 k_1 k_2^2 k_{21} - 32760 k_2 k_1 k_{221}) a_2 + (7938 k_2^2 k_1^3 \\
& + 56700 k_1 k_2 k_{211} - 49896 k_1^2 k_2 k_{21} + 22680 k_1 k_{21}^2 \\
& + 34020 k_1^2 k_{221} - 45360 k_{211} k_{21}) b_1 + (7920 k_2^2 k_1^3 \\
& + 57960 k_1 k_2 k_{211} - 50328 k_1^2 k_2 k_{21} + 21600 k_1 k_{21}^2 \\
& + 35784 k_1^2 k_{221} - 45360 k_{211} k_{21}) b_2 \\
& + (-18 k_1^3 k_2 + 36 k_1^2 k_{21}) b_1 b_2 - 41580 k_2^3 k_1^3 \\
& - 309960 k_2^2 k_1 k_{211} + 355320 k_{21} k_2^2 k_1^2 - 309960 k_2 k_1^2 k_{221} \\
& + 725760 k_{21} k_2 k_{211} - 544320 k_1 k_2 k_{21}^2 + 725760 k_{21} k_1 k_{221} \\
& - 635040 k_{221} k_{211} + (-18 k_1^3 k_2 + 36 k_1^2 k_{21}) b_2^2
\end{aligned}$$

6.2.2 Méthode d'identification

Le système possède 5 équations. En choisissant des entrées constantes par morceaux ou polynomiales le degré total de chacune par rapport aux inconnues (les aires de chaque palier et les coefficients des polynômes respectivement) est le suivant : degré 1 pour la première et la deuxième, degré 2 pour la troisième, la quatrième et cinquième sont de degré 3.

Dans tous les cas il faut au moins 6 inconnues pour résoudre le système, c'est à dire une de plus que le nombre d'équations, pour une solution générique (pour un point cible pris n'importe où à l'extérieur de certaines sous variétés). On dispose donc pour résoudre de 1 à 3 paramètres libres.

Entrées constantes par morceaux

En choisissant bien l'ordre et en ajoutant des conditions sur les paramètres, le système devient *linéaire*. On trouve ainsi une solution en au plus 4 paliers c'est à dire 8 paramètres (3 pour une solution générique). En procédant comme suit :

1. de la première équation ($\xi_1(T) = q_1$) on extrait A_1 linéairement
2. de la deuxième équation ($\xi_2(T) = q_1$) on extrait B_1 linéairement

3. de la troisième équation ($\xi_{21}(T) = q_{21}$) on extrait B_2 linéairement
4. de la quatrième équation ($\xi_{211}(T) = q_{211}$) on extrait B_3 linéairement
5. la dernière équation ($\xi_{221}(T) = q_{221}$) est rationnelle sur A_2, A_3 et A_4 , polynomiale de degré 2 sur B_4 . En posant ($A_3 + A_4 = 0$) le coefficient de B_4^2 s'annule. On peut alors résoudre linéairement.

on obtient la solution générale suivante :

> A[1];

$$k1 - A_2$$

> A[2];

$$A_2$$

> A[3];

$$A_3$$

> A[4];

$$-A_3$$

> B[1];

$$\frac{k2 A_3 A_2 + 2 k21 A_2 - 6 k211 - 2 k21 A_3}{k1 (k1 + A_3)}$$

> B[2];

$$\begin{aligned} & (k2 A_3 A_2^2 + 2 k21 A_2^2 + k2 A_3 k1 A_2 + 2 k21 k1 A_2 + k2 A_3^2 A_2 \\ & - 6 A_2 k211 + k2 A_3 k1^2 + 2 k21 k1^2 + k2 A_3^2 k1 - 6 k211 k1 \\ & - 6 k211 A_3 - 2 k21 A_3^2) / (k1 (A_3 + A_2) (k1 + A_3)) \end{aligned}$$

> B[3];

$$\begin{aligned}
& (72k211^2 k1 A_3 - 18k2 A_3^2 k1^2 k211 - 18k2 A_3^3 k211 k1 - k2^2 A_3 A_2^2 k1^3 \\
& + k2^2 A_3^4 A_2 k1 + 6k2 A_3^3 k21 k1^2 - 6k221 k1 A_2^2 A_3^2 \\
& + 4k21^2 A_2 A_3^2 k1 + 8k21 k1 A_3^3 k2 A_2 - 12k211 k1 k21 A_2^2 \\
& - 6k211 k1 k2 A_3 A_2^2 - 24k211 k1 k2 A_3^2 A_2 + 6k21 k1^3 A_3^2 k2 \\
& + 8k21 k1 A_3^2 k2 A_2^2 - 48k21 k1^2 A_3 k211 + 8k21^2 k1 A_3 A_2^2 \\
& + 16k21 A_2 A_3^2 k2 k1^2 - 36k21 A_2 A_3 k211 k1 + 16k21^2 A_2 A_3 k1^2 \\
& + 8k21^2 k1^3 A_3 + 8k21^2 k1^2 A_2^2 + 6k21 k1^2 k2 A_3 A_2^2 \\
& - k2^2 A_3^3 k1^2 A_2 - k2^2 A_3^2 k1^2 A_2^2 + 36k211^2 k1 A_2 \\
& - 24k211 k1^2 k21 A_2 - 6k211 k1^2 k2 A_3 A_2 + k2^2 A_3^3 A_2^2 k1 \\
& - 2k2^2 A_3^2 k1^3 A_2 + 4k21^2 k1^3 A_2 + 2k21 k1^3 k2 A_3 A_2 \\
& - 4k21^2 A_2^2 A_3^2 + 24k211 A_3^3 k21 + 36k211^2 A_3 A_2 \\
& + 36k211^2 A_3^2 + 4k21^2 A_3^4 + 12k21 A_2 A_3^2 k211 \\
& - 2k21 A_2 A_3^4 k2 - 2k21 A_2^2 A_3^3 k2 - 12k21 A_2^2 A_3 k211 \\
& - 6k211 A_3^2 k2 A_2^2 - 6k211 A_3^3 k2 A_2 - 6k221 k1^3 A_3^2 \\
& - 12k221 k1^2 A_3^3 - 6k221 k1^3 A_2^2 - 6k221 k1 A_3^4 \\
& - 12k221 k1^3 A_2 A_3 - 24k221 k1^2 A_2 A_3^2 - 12k221 k1^2 A_2^2 A_3 \\
& - 12k221 k1 A_2 A_3^3) / (k1 A_3 \\
& (k2 A_2 k1 - 2k21 A_2 - 2k21 k1 + 6k211) (k1 + A_3) (A_3 + A_2))
\end{aligned}$$

> B[4];

$$\begin{aligned}
& - \left(-2k2 A_3^2 k21 A_2 + k2^2 A_3^2 A_2 k1 + 10k2 A_3 A_2 k21 k1 \right. \\
& + 6k2 A_3 k1^2 k21 - 2k2^2 A_3 k1^2 A_2 - 6k2 A_3 A_2 k211 \\
& - 18k2 A_3 k1 k211 - 4k21^2 A_3 A_2 - 4k21^2 A_3 k1 + 8k21^2 A_2 k1 \\
& + 24k21 A_3 k211 - 12k21 A_2 k211 - 24k211 k21 k1 + 4k21^2 A_3^2 \\
& + 36k211^2 + 4k21^2 k1^2 - 6k221 k1 A_2 A_3 - 6k221 k1^2 A_3 \\
& \left. - 6k221 k1^2 A_2 - 6k221 k1 A_3^2 \right) / (k1 A_3 \\
& (k2 A_2 k1 - 2k21 A_2 - 2k21 k1 + 6k211))
\end{aligned}$$

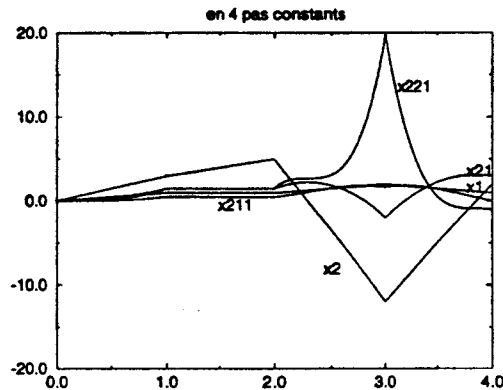


FIG. 6.1 - Solution en 4 morceaux constants, arrivée en (1,2,3,0,-1)

Entrées polynomiales

Pour des entrées polynomiales de degré 3 l'étude du système d'équation correspondant amène à une discussion suivant les valeurs de la cible: $k_l = 0$ ou $k_l \neq 0$. Pour tous les cas singuliers sauf un, on obtient directement la solution, c'est à dire qu'on a pu rendre le système linéaire par rapport aux inconnues (pas par rapport aux paramètres). Le cas faisant exception demande la résolution d'une équation quadratique qu'on réussit à résoudre formellement dans tous les cas.

Dans le cas générique ($k_1 \neq 0$), le système se ramène à une équation du 5^e degré, on trouve donc toujours une solution réelle.

Exemple :

Le cas generique: $k_1 \neq 0$

```
> p1 := a0 + a1*t + a2*t^2 + a3*t^3 ; define( u1, forall(t, u1(t)=p1) ):
> p2 := b0 + b1*t + b2*t^2 + b3*t^3 ;define( u2, forall(t, u2(t)=p2) ):
>
> x1 := integrate( p1 ,t):
> x2 := integrate( p2 ,t):
> x21 := normal( integrate( x2*p1 ,t) ):
> x211 := normal( integrate( x21*p1 ,t) ):
> x221 := normal( integrate( 1/2*x2^2*p1 ,t) ):
>
> t := 1:
>
> a1 := 0:
> a3 := 0:
>
>
```

$$p1 := a0 + a1 t + a2 t^2 + a3 t^3$$

$$p2 := b0 + b1 t + b2 t^2 + b3 t^3$$

```
> f1 := k1 - x1 ;
>
```

$$f1 := k1 - a0 - \frac{1}{3} a2$$

```
> f2 := k2 - x2 ;
>
```

$$f2 := k2 - b0 - \frac{1}{2} b1 - \frac{1}{3} b2 - \frac{1}{4} b3$$

```
> f21 := k21 - x21 ;
>
```

$$f21 := k21 - \frac{1}{28} b3 a2 - \frac{1}{18} b2 a2 - \frac{1}{10} b1 a2 - \frac{1}{20} b3 a0 - \frac{1}{4} b0 a2 \\ - \frac{1}{12} b2 a0 - \frac{1}{6} b1 a0 - \frac{1}{2} b0 a0$$

```
> f211 := k211 - x211 ;
>
```

$$f211 := k211 - \frac{1}{80} b1 a2^2 - \frac{1}{6} b0 a0^2 - \frac{1}{280} b3 a2^2 - \frac{1}{162} b2 a2^2 - \frac{1}{28} b0 a2^2 \\ - \frac{3}{280} a2 b3 a0 - \frac{5}{252} a2 b2 a0 - \frac{1}{120} b3 a0^2 - \frac{1}{60} b2 a0^2 - \frac{1}{24} b1 a0^2 \\ - \frac{2}{45} a2 b1 a0 - \frac{3}{20} b0 a0 a2$$

```
> f221 := k221 - x221 ;
>
```

$$f221 := k221 - \frac{1}{6} b0^2 a0 - \frac{1}{352} b3^2 a2 - \frac{1}{162} a2 b2^2 - \frac{1}{288} b3^2 a0 \\ - \frac{1}{56} a2 b1^2 - \frac{1}{120} b2 b3 a2 - \frac{1}{72} a2 b1 b3 - \frac{1}{32} a2 b0 b3 \\ - \frac{1}{48} a2 b1 b2 - \frac{1}{96} b2 b3 a0 - \frac{1}{21} a2 b0 b2 - \frac{1}{56} a0 b1 b3 \\ - \frac{1}{126} a0 b2^2 - \frac{1}{10} b0^2 a2 - \frac{1}{40} a0 b1^2 - \frac{1}{12} b0 b1 a2 - \frac{1}{24} a0 b0 b3 \\ - \frac{1}{36} a0 b1 b2 - \frac{1}{15} a0 b0 b2 - \frac{1}{8} b0 b1 a0$$

Les 2 premières équations f1 et f2 sont respectivement de degré 1 en (a0) et (b0), on les élimine :

```
> z0 := f1 + a0 : a0 := z0 ;
> z0 := f2 + b0 : b0 := z0 ;
>
>
```

$$a0 := k1 - \frac{1}{3} a2$$

$$b0 := k2 - \frac{1}{2} b1 - \frac{1}{3} b2 - \frac{1}{4} b3$$

f21 est de degré 1 en (b1, b2, b3), on peut fixer b1 par exemple :

```
> dom21 := factor( quo( f21 , b1, b1, 'res21' ) ):
> b1 := normal( - res21/dom21 );
>
>
```

$$b1 := \frac{3}{14}(1680 k21 + 3 b3 a2 + 126 b3 k1 - 140 a2 k2 + 140 b2 k1 - 840 k2 k1) / (a2 - 30 k1)$$

f211 est de degré 1 en (b2, b3), on peut fixer b2 par exemple:

```
> dom211 := factor( quo( f211 , b2, b2, 'res211' ) ):
> b2 := normal( - res211/dom211 );
```

$$b2 := \frac{1}{28}(211680 k211 a2 - 6350400 k211 k1 - 15120 a2^2 k21 + a2^3 b3 + 364 a2^3 k2 - 630 a2^2 b3 k1 + 8568 a2^2 k2 k1 + 211680 a2 k21 k1 - 378 a2 b3 k1^2 - 158760 a2 k2 k1^2 - 529200 k2 k1^3 + 3175200 k21 k1^2 + 26460 b3 k1^3) / (k1 (13 a2^2 + 36 a2 k1 - 630 k1^2))$$

$$656 b3^2 k1$$

f221 est alors de degré 2 en (b3), son numérateur numer(f221) de degré 5 en a2 : on résoud donc numériquement l'équation de degré 5 en a2

```
> degree( numer(normal(f221)), a2 );
>
>
5

> degree( numer(normal(f221)), b3 );
2
```

Exemple de résolution numérique :

```
> b3 := 1:
> k1 := 1:   k2 := 2 :   k21 := 3 :   k211 := 0 :   k221 := -1 :
>
```

```
> racines := op(1,solve( f221, a2 ));
>
>
```

$$\begin{aligned} \text{racines} := & 656 _Z^5 - 34888053795 _Z^4 + 365764409640 _Z^3 \\ & + 10960178274360 _Z^2 + 7702916765040 _Z \\ & - 310171206892932 \end{aligned}$$

```
> reelles := fsolve( {racines}, {_Z} );
>
```

$$\begin{aligned} \text{reelles} := & \{ _Z = -9.829951085 \}, \{ _Z = -8.020033235 \}, \\ & \{ _Z = 4.789760551 \}, \{ _Z = 23.54417595 \}, \\ & \{ _Z = .5318299835 \cdot 10^8 \} \end{aligned}$$

```
> for sol in reelles do
>   print('-----'):
>   a2 := 'a2' :
>   eq := subs(_Z=a2,sol):
>   assign(eq):
>   u := u1(x);
>   v := u2(x);
> od;
>
```

$$a2 := a2$$

$$eq := \{ a2 = -9.829951085 \}$$

$$u := 4.276650361 - 9.829951085 x^2$$

$$v := 29.21006357 - 220.6802927 x + 248.6402485 x^2 + x^3$$

$$a2 := a2$$

$$eq := \{ a2 = -8.020033235 \}$$

$$u := 3.673344411 - 8.020033235 x^2$$

$$v := -80.60803510 + 1237.150506 x - 1608.651654 x^2 + x^3$$

$$a2 := a2$$

$$eq := \{ a2 = 4.789760551 \}$$

$$u := -.596586850 + 4.789760551 x^2$$

$$v := -542.2326574 + 2497.308268 x - 2114.014430 x^2 + x^3$$

$$a2 := a2$$

$$eq := \{ a2 = 23.54417595 \}$$

$$u := -6.848058649 + 23.54417595 x^2$$

Methode IDENTIFICATION

Methode : IDENTIFICATION

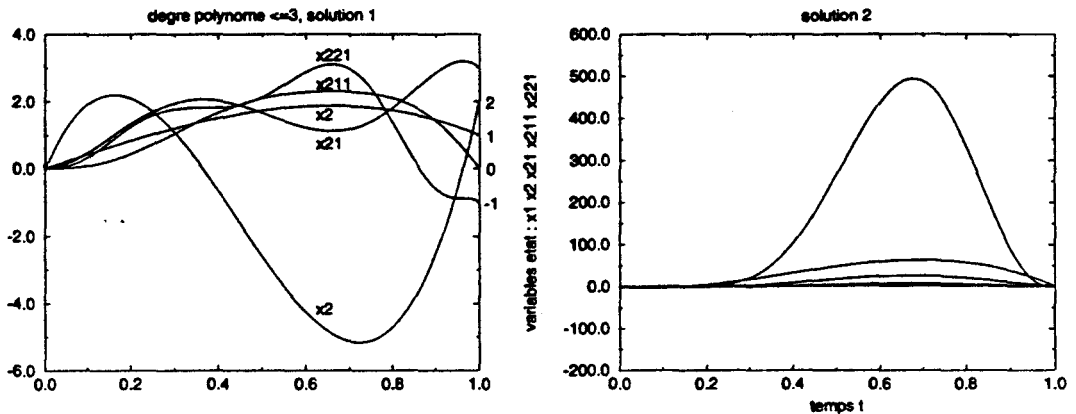


FIG. 6.2 - Solution polynomiale de degré 3, arrivée en (1, 2, 3, 0, -1

$$v := 42.13284405 - 111.1703876 x + 45.60704924 x^2 + x^3$$

$$a2 := a2$$

$$eq := \{ a2 = .5318299835 \cdot 10^8 \}$$

$$u := -.1772766511 \cdot 10^8 + .5318299835 \cdot 10^8 x^2$$

$$v := -.3550400497 \cdot 10^8 + .7252413519 x + .1065120191 \cdot 10^9 x^2 + x^3$$

Remarques : l'étude des résultants des dénominateurs des solutions avec l'équation de degré 5 montre que l'on peut annuler cette dernière sans annuler les contraintes (tout va bien).

6.2.3 Solution, simulations, pour la pièce qui roule

Une solution générique de $g_u = g_q$ pour une entrée u en 3 morceaux constants peut être calculée en utilisant les bases de Gröbner.

Pour les valeurs singulières de g_q d'autres méthodes et plus d'inconnues sont nécessaires. Par exemple pour une cible de coordonnées exponentielles $g_q = (q_1 = 0, q_2 = 0, q_{12} = 0, q_{112} = \phi, q_{122})$, le problème n'a pas de solution en 3 paliers mais en 4, i.e. pour 8 paramètres au lieu de 6. Dans le problème de la pièce de monnaie, de telles

coordonnées apparaissent lorsqu'on veut atteindre le point $q = (0, 0, 0, \phi)$ en partant de $p = (0, 0, 0, 0)$, i.e. lorsqu'on veut ajuster uniquement l'angle x_4 à ϕ sans changer les autres composantes ce qui supposerait de la faire rouler en glissant sans avancer. Pour des entrées polynomiales de degré 3 on obtient la solution paramétrée suivante :

$$a_1 = \frac{2}{11} a_4 t^3 + \frac{1}{6} a_3 t^2 + 105840 \frac{\phi}{(a_3 b_4 - a_4 b_3) t^8}$$

$$a_2 = -\frac{19}{22} a_4 t^2 - a_3 t - 211680 \frac{\phi}{(a_3 b_4 - a_4 b_3) t^9}$$

$$b_1 = \frac{6}{11} \frac{a_4 b_4}{(3 a_4 t + 2 a_3)} t^4 + \frac{1}{154} \frac{(55 a_3 b_4 + 78 a_4 b_3)}{(3 a_4 t + 2 a_3)} t^3 \\ + \frac{1}{3} \frac{a_3 b_3}{(3 a_4 t + 2 a_3)} t^2 + \frac{105840 (3 b_4 t + 2 b_3)}{(a_3 b_4 - a_4 b_3) (3 a_4 t + 2 a_3) t^8} \phi$$

$$b_2 = -\frac{57}{22} \frac{a_4 b_4}{(3 a_4 t + 2 a_3)} t^3 - \frac{4}{77} \frac{(33 a_3 b_4 + 58 a_4 b_3)}{(3 a_4 t + 2 a_3)} t^2 \\ - 2 \frac{a_3 b_3}{(3 a_4 t + 2 a_3)} t - \frac{211680 (3 b_4 t + 2 b_3)}{(a_3 b_4 - a_4 b_3) (3 a_4 t + 2 a_3) t^9} \phi$$

avec les contraintes :

$$a_3 b_4 \neq a_4 b_3 \quad \text{and} \quad a_3 \neq -\frac{3}{2} a_4 t$$

les variables a_3, a_4, b_3 et b_4 apparaissent donc comme des paramètres. En faisant varier ces paramètres on obtient différentes trajectoires admissibles du système qui mènent toutes (Σ_S) exactement à la cible.

Par exemple en fixant 3 des 4 paramètres : a_3, a_4, b_3 et le temps t :

$$\begin{cases} a_3 = -1 \\ a_4 = 1 \end{cases} \quad \text{and} \quad \begin{cases} b_3 = -\frac{3}{2} b_4 t \\ t = 4 \end{cases}$$

on garde b_4 comme paramètre libre. Les graphiques suivants représentent les simu-

lations du comportement du système pour les entrées correspondant à différentes valeurs de b_4 et pour une cible unique $(0, 0, 0, \pi)$:

$$b_4 = \frac{1}{100}, \frac{1}{10}, \frac{1}{7}, \frac{1}{5}, \frac{3}{4}, 1$$

elles font apparaître la pièce vue de dessus comme dans la figure 6.3, où l'angle (x_4) est représenté par une flèche.

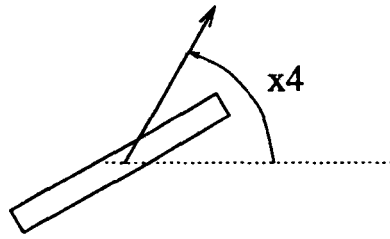
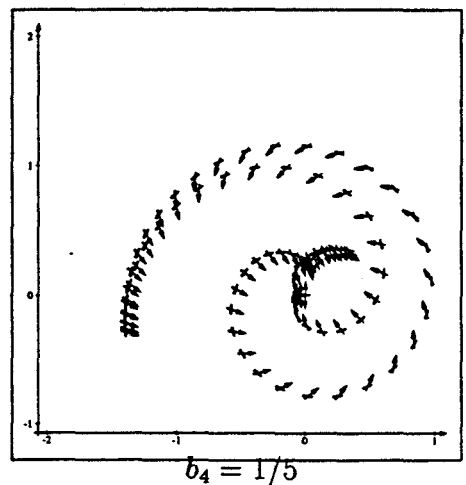
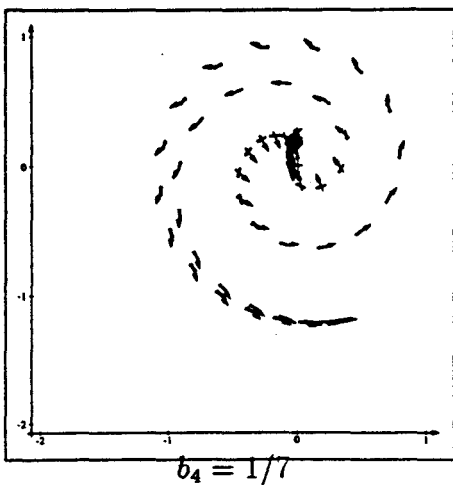
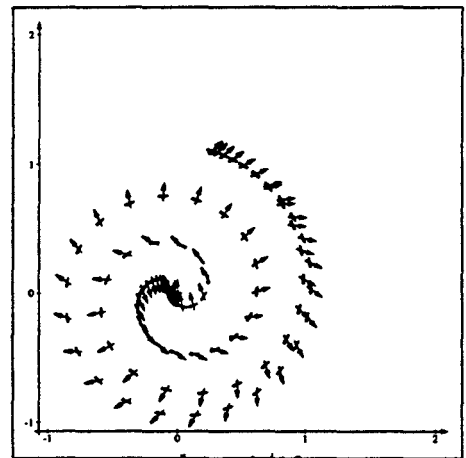
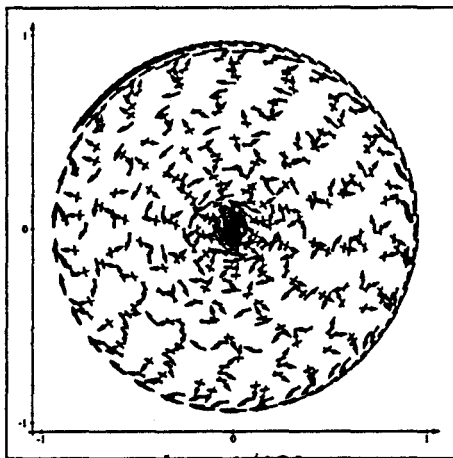
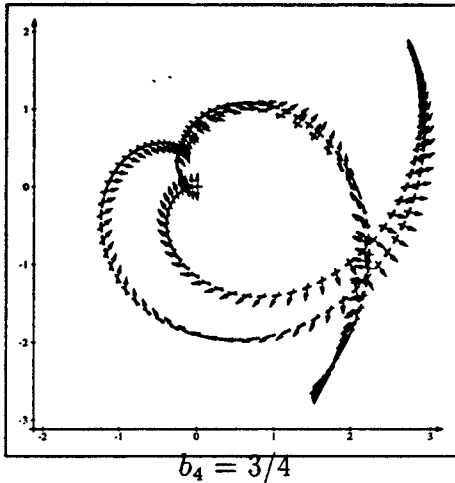
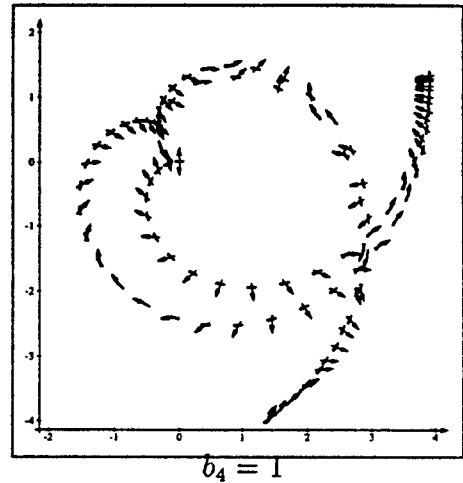


FIG. 6.3 - pièce de monnaie vue de dessus



 $b_4 = 3/4$  $b_4 = 1$

6.3 Systèmes avec dérive

Un système dynamique analytique affine en commande *avec dérive* est de la forme :

$$(\Sigma) \quad \frac{d}{dt}x = X_0(x) + u_1 X_1(x) + u_2 X_2(x) + \dots + u_m X_m(x)$$

lorsque le vecteur d'entrée est nul le système évolue librement suivant le champ X_0 de dérive.

Cette situation peut être modélisée en prenant une entrée constante fictive $u_0 = 1$. L'autre entrée u_1 reste quelconque. On peut donc chercher à résoudre avec des entrées constantes par morceaux, comme dans le cas d'un système sans dérive, les 5 équations algébriques (6.1.1), en faisant les modifications suivantes : $A_i = T_i - T_{i-1} > 0$ pour tout i . Aux équations précédentes se sont donc ajoutées des contraintes d'inégalité. La résolution dans ces conditions s'avère trop difficile encore, on peut l'éviter cependant en posant $A_i = \Delta$ pour tout i , c'est à dire en supposant les intervalles de temps constants.

Le système d'équations algébriques d'un système nilpotent à l'ordre 4 avec dérive est alors donné par :

$$\begin{aligned}
\xi_1 = T &= N\Delta \\
\xi_2 &= \sum_{i=1}^N B_i \\
\xi_{12} &= \left(\frac{1}{2} \sum_{i=1}^N B_i + \sum_{1 \leq i \leq N} B_i \right) \Delta \\
\xi_{112} &= \left(\frac{1}{3!} \sum_{i=1}^N B_i + \sum_{1 \leq i < N} (N-i)B_i + \sum_{1 \leq i < N-1} \binom{N-i}{2} B_i \right) \Delta^2 \\
\xi_{122} &= \left(\frac{1}{3!} \sum_{i=1}^N B_i^2 + \frac{1}{2} \sum_{1 \leq j < i \leq N} B_i B_j \right. \\
&\quad \left. + \frac{1}{2} \sum_{1 \leq i \leq N} (N-i)B_i^2 + \sum_{1 \leq j < i \leq N} (N-i)B_i B_j \right) \Delta
\end{aligned} \tag{6.1}$$

On commence alors la résolution comme pour le système sans dérive (section 6.2.2) :

- Δ est tirée de la première équation
- B_1 de la deuxième
- B_2 de la troisième
- B_3 de la quatrième

En posant :

$$\chi_0 = \xi_0 = T, \quad \chi_2, \quad \chi_{10} = \frac{\xi_{10}}{T}, \quad \chi_{100} = \frac{\xi_{100}}{T^2}, \quad \chi_{110} = \frac{\xi_{110}}{T},$$

on montre que la cinquième équation est de la forme :

$$Q_N(B_i, \chi_i) + R_N(\chi_i)$$

où Q_N est une forme quadratique définie positive en les B_i restants ($i = 4 \dots N$) et R_N ne dépend que des χ_i . Dès lors une condition suffisante pour que le système avec dérive soit commandable par une entrée constante par morceaux en N paliers de même durée Δ est :

$$R_N(\chi_i) \leq 0$$

Par ailleurs, le calcul de ce reste pour différentes valeurs de N montre qu'il peut lui-même se mettre sous la forme : d'une forme quadratique définie positive plus un reste, le même quel que soit N :

$$R_N(\chi_i) = \cdot \chi_1^2 + \cdot (\chi_{100} + \cdot \chi_1)^2 + \cdot (\chi_{10} + \cdot \chi_{100} + \cdot \chi_0)^2 - 6\chi_{110}$$

où les coefficients (.) sont calculés formellement.

On s'intéresse alors à la convergence de ces coefficients en fonction de N . À titre d'exemple les expressions suivantes donnent R_N pour différentes valeurs du nombre de paliers avec une approximation flottante des coefficients :

$$\begin{aligned}
 R_4 &= 0.2165 \chi_1^2 + 11.49 (\chi_{100} - 0.00074 \chi_1)^2 \\
 &\quad + 19.5 (\chi_{10} - 1.641 \chi_{100} - 0.073 \chi_1)^2 - 6\chi_{110} \\
 R_5 &= 0.1732 \chi_1^2 + 21.88 (\chi_{100} - 0.00016 \chi_1)^2 \\
 &\quad + 17.14 (\chi_{10} - 1.604 \chi_{100} - 0.0576 \chi_1)^2 - 6\chi_{110} \\
 R_{50} &= 0.01732 \chi_1^2 + 9.159 (\chi_{100} - 3 \cdot 21^{-31} \chi_1)^2 \\
 &\quad + 12.36 (\chi_{10} - 1.509 \chi_{100} - 0.00577 \chi_1)^2 - 6\chi_{110} \\
 R_{90} &= 0.0096 \chi_1^2 + 9.087 (\chi_{100} - 2 \cdot 21^{-54} \chi_1)^2 \\
 &\quad + 12.196 (\chi_{10} - 1.505 \chi_{100} - 0.0032 \chi_1)^2 - 6\chi_{110} \\
 R_{100} &= 0.0087 \chi_1^2 + 9.079 (\chi_{100} - 2 \cdot 37^{-61} \chi_1)^2 \\
 &\quad + 12.176 (\chi_{10} - 1.5044 \chi_{100} - 0.00288 \chi_1)^2 - 6\chi_{110}
 \end{aligned}$$

6.4 Systèmes chaînés

On pose :

$$\begin{aligned}
 u_1 &= a_0 + a_1 t + a_2 t^2 + \dots + a_m t^m \\
 u_2 &= b_0 + b_1 t + b_2 t^2 + \dots + b_n t^n
 \end{aligned}$$

où les (a_i) et les (b_i) sont les inconnues du problème. Trouver leur expression en fonction des coordonnées de la cible résoud la planification de trajectoire symboliquement : l'entrée est complètement déterminée par la donnée des coefficients).

6.4.1 Système chaîné de 4 équations

Rappelons ces équations :

$$\begin{cases} \dot{\xi}_1 &= u_1 \\ \dot{\xi}_2 &= u_2 \\ \dot{\xi}_{21} &= u_1 \xi_2 \\ \dot{\xi}_{211} &= u_1 \xi_{21} \end{cases}$$

Minimiser le nombre de paramètres, le degré des polynômes

Lorsqu'on cherche à minimiser le nombre de paramètres et le nombre d'équations on obtient les résultats suivants : en éliminant simplement chacune des variables suc-

cessivement, 5 variables sont nécessaires et suffisantes pour atteindre tout l'espace d'état.

Les solutions polynomiales à 5 paramètres de degré minimal sont de la forme :

$$\begin{aligned} u_1 &= a_0 + a_3 t^3 \\ u_2 &= b_0 + b_1 t + b_2 t^2 \end{aligned}$$

ou

$$\begin{aligned} u_1 &= a_0 + a_2 t^2 \\ u_2 &= b_0 + b_1 t + b_3 t^3 \end{aligned}$$

c'est à dire qu'il faut qu'une au moins des 2 entrées soit de degré 3 et l'autre de degré 2.

Les solutions génériques à 4 paramètres, polynomiales ou en 2 paliers constants, font intervenir une au moins des deux contraintes :

$$q_1 \neq 0, \quad 2q_{21} \neq q_2 q_1$$

comme par exemple l'entrée suivante qui ne mène à une solution que pour $q_1 \neq 0$:

$$\begin{aligned} u_1 &= a_0 \\ u_2 &= b_0 + b_1 t + b_2 t^2 + b_3 t^3 \end{aligned}$$

On peut poser $T = 1$ dans toutes les équations, cela revient à faire le changement de variables :

$$\begin{cases} A_0 = a_0 T, \dots, A_i = a_i T^{i+1} \\ B_0 = b_0 T, \dots, B_i = b_i T^{i+1} \end{cases}$$

Dans le cas singulier ($q_1 = 0$) on montre que pour une entrée du type :

$$\begin{aligned} u_1 &= a_0 + a_k t^k \\ u_2 &= b_0 + b_1 t + \dots + b_r t^r \end{aligned}$$

certaines variables disparaissent des équations lors du processus d'élimination. Plus précisément on peut raisonnablement conjecturer que l'annulation des coefficients des variables obéit à la règle suivante :

$$\begin{aligned} \text{degré}(b_k) &= 0 && \text{dans } f_{21}, f_{211}, \dots \\ \text{degré}(b_{2k+1}) &= 0 && \text{dans } f_{21^{k+1}}, f_{21^{k+2}}, \dots \\ &\vdots && \\ \text{degré}(b_{ik}) &= 0 && \text{dans } f_{21^{(i-1)k+1}}, f_{21^{(i-1)k+2}}, \dots \end{aligned}$$

Ceci sera utilisé dans la méthode qui permet de résoudre 19 équations.

6.4.2 N équations

La méthode s'étend naturellement au cas général. Pour un système chaîné de $(r + 2)$ équations on choisit plutôt, dans le *cas singulier* ($q_1 = 0$), une commande du type :

$$\begin{aligned} u_1 &= a_0 + a_{r+1}t^{r+1} \\ u_2 &= b_0 + b_1t + \dots + b_rt^r \end{aligned}$$

et dans le *cas régulier*, une commande du type :

$$\begin{aligned} u_1 &= a_0 \\ u_2 &= b_0 + b_1t + \dots + b_rt^r \end{aligned}$$

6.5 Jusqu'à 19 équations

Les 5 équations algébriques s'avèrent difficiles à résoudre. La complexité des équations suivantes croît : le degré de celles-ci est égal à la longueur du mot de Širšov correspondant. Aussi est-il longtemps resté peu probable de pouvoir poursuivre beaucoup plus avant la résolution (pour des systèmes plus importants). Un examen attentif du comportement de ces équations a cependant fait apparaître des caractères remarquables qui ont permis de résoudre complètement jusqu'à 19 équations et ce *linéairement* pour des cas particuliers, on en déduit alors un schéma de résolution *symbolique* pour toutes les cibles.

6.5.1 Remarques préliminaires

Élimination dans le sous système chaîné

Parmi les équations on distingue tout d'abord la partie chaînée des autres équations. Elle forme un sous système pour lequel une résolution complète a été proposée.

Dans le cas où la cible vérifie $q_1 = 0$ et l'entrée :

$$\begin{cases} u_1 = a_0 + a_1t \\ u_2 = b_0 + b_1t + \dots + b_rt^r \end{cases}$$

les termes en $b_1, b_3, \dots, b_{2k+1}$ disparaissent des équations de la partie chaînée.

Élimination dans les équations restantes

a) 19 équations dans un cas singulier.

Pour la partie non chaînée, le plus grand nombre d'équations ayant pu être pris en compte par la méthode ici exposée est 8. Ces équations supplémentaires sont liées aux mots suivants :

$$221, 2211, 221^3, 221^4, 21211, 2221, 22211, 22121$$

On considère pour cela une cible particulière dont certaines composantes sont nulles; ceci permettant de simplifier les équations algébriques. On suppose l'entrée u de type :

$$\begin{cases} u_1 = a_0 + a_1 t \\ u_2 = b_0 + b_1 t + \dots + b_{18} t^{18} \end{cases}$$

Pour une chaîne allant jusqu'à 21^9 les équations ($\xi(T) = g_u = M$) sont les suivantes :

$$\begin{pmatrix} \xi_1(T_M) \\ \xi_2(T_M) \\ \xi_{21}(T_M) \\ \xi_{211}(T_M) \\ \vdots \\ \xi_{21^8}(T_M) \\ \xi_{21^9}(T_M) \\ \xi_{221}(T_M) \\ \xi_{2211}(T_M) \\ \xi_{221^3}(T_M) \\ \xi_{221^4}(T_M) \\ \xi_{21211}(T_M) \\ \xi_{2221}(T_M) \\ \xi_{22211}(T_M) \\ \xi_{22121}(T_M) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ M_{21^9} \neq 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ M_{221^4} \neq 0 \\ M_{21211} \neq 0 \\ M_{2221} \\ M_{22211} \\ M_{22121} \end{pmatrix}$$

Au cours de l'élimination des b_i dans les équations de la partie chaînée et dans ces conditions sur la cible, les $(b_{2i+1}), (i = 0, \dots, 8)$ disparaissent des équations restantes en commençant par les termes de plus haut degré (en les b_{2i+1}), comme dans le cas chaîné. À la fin de l'élimination dans la partie chaînée, les caractéristiques du système des équations non chaînées sont les suivantes :

- les b_i y apparaissent au plus au degré 2 (pour un degré dans le système initial de 3 au plus)
- $f_{221}, f_{2211}, f_{21^3}, f_{21^4}$ forment un système linéaire triangulaire en $b_9, b_{11}, b_{13}, b_{15}$ (les équations différentielles correspondantes forment la fin d'une chaîne aboutissant à x_2)

– $f_{2221}, f_{22211}, f_{22121}$ sont linéaires en b_1, b_3, b_5 (c'est le groupe des équations dont les mots de codage comportent trois fois la lettre "2")

– f_{21211} est linéaire en tous les b_i qu'elle comporte: $b_{11}, b_{13}, b_{15}, b_{17}$

on résoud symboliquement ce système par élimination successive de $b_{15}, b_{13}, b_{11}, b_9, b_{17}, b_5, b_3, b_1$ dans $f_{221}, f_{2211}, f_{21^3}, f_{21^4}, f_{21211}, f_{2221}, f_{22211}, f_{22121}$ (respectivement).

b) 16 équations dans un cas singulier.

Par ailleurs, pour un système comprenant les 5 équations supplémentaires — au lieu des 8 précédentes — liées au mots :

$$221, 2211 \ 221^3, 221^4, 21211$$

et pour une partie chaînée identique, on trouve une solution symbolique au problème suivant : trouver une commande $v(t)$

$$\begin{cases} v_1 = a_0 + a_1 t \\ v_2 = b_0 + b_1 t + \dots + b_{18} t^{18} \end{cases}$$

menant le système de l'origine à N :

$$\begin{pmatrix} \xi_1(T_N) \\ \xi_2(T_N) \\ \xi_{21}(T_N) \\ \xi_{211}(T_N) \\ \vdots \\ \xi_{21^8}(T_N) \\ \xi_{21^9}(T_N) \\ \xi_{221}(T_N) \\ \xi_{2211}(T_N) \\ \xi_{221^3}(T_N) \\ \xi_{221^4}(T_N) \\ \xi_{21211}(T_N) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ N_{21^9} \neq 0 \\ N_{221} \\ N_{2211} \\ N_{221^3} \\ N_{221^4} \\ N_{21211} \end{pmatrix}$$

Si le système possède en fait les 19 équations du système précédent, 3 variables d'état ont évolué librement :

$$\begin{aligned} \xi_{2221}(T_N) &= \mathcal{E}_v(S_{2221})(T_N) \\ \xi_{22211}(T_N) &= \mathcal{E}_v(S_{22211})(T_N) \\ \xi_{22121}(T_N) &= \mathcal{E}_v(S_{22121})(T_N) \end{aligned}$$

6.5.2 Une méthode en 3 mouvements

Description

En mettant bout à bout le le mouvement du a) (pour les 19 équations) et celui du b) (pour les 16 équations), grâce aux formules de la loi du groupe nilpotent libre (cf. annexe B), on arrive en :

$$g_{u\#v} = g_u \circ g_v = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ M_{21^9} + N_{21^9} \\ N_{221} \\ N_{2211} \\ N_{221^3} \\ M_{221^4} + N_{221^4} \\ M_{21211} + N_{21211} \\ M_{2221} + \mathcal{E}_v(S_{2221})(T_N) \\ M_{22211} + \mathcal{E}_v(S_{22211})(T_N) \\ M_{22121} + \mathcal{E}_v(S_{22121})(T_N) \end{pmatrix}$$

où les M_l, N_l doivent répondre aux seules contraintes :

$$\begin{aligned} M_{21^9} &\neq 0 & N_{21^9} &\neq 0 \\ M_{221^4} &\neq 0 \\ M_{21211} &\neq 0 \end{aligned}$$

On voit donc que toutes les composantes non chaînées du système sont commandables. Un dernier mouvement doit donc permettre d'ajuster les composantes de la partie chaînée en $Q = (Q_1, Q_2, Q_{21}, \dots, Q_{21^9})$. En utilisant la méthode spécifique aux systèmes chaînés on peut calculer symboliquement une commande w :

$$\begin{cases} w_1 = a_0 \\ w_2 = b_0 + b_1 t + \dots + b_9 t^9 \end{cases}$$

dans le cas régulier et dans le cas singulier :

$$\begin{cases} w_1 = a_0 + a_{10} t^{10} \\ w_2 = b_0 + b_1 t + \dots + b_9 t^9 \end{cases}$$

que l'on concatène avec les deux précédentes (loi du groupe nilpotent libre) pour arriver finalement en :

$$g_{u\#v\#w} = g_u \circ g_v \circ g_w = \begin{pmatrix} Q_1 \\ Q_2 \\ Q_{21} \\ Q_{211} \\ \vdots \\ Q_{21^6} \\ M_{21^9} + N_{21^9} + Q_{21^9} \\ N_{221} + \mathcal{E}_w(S_{2221})(T_Q) \\ N_{2211} + \mathcal{E}_w(S_{2221})(T_Q) \\ N_{221^3} + \mathcal{E}_w(S_{2221})(T_Q) \\ M_{221^4} + N_{221^4} + \mathcal{E}_w(S_{2221})(T_Q) \\ M_{21211} + N_{21211} + \mathcal{E}_w(S_{2221})(T_Q) \\ M_{2221} + \mathcal{E}_v(S_{2221})(T_N) + \mathcal{E}_w(S_{2221})(T_Q) \\ M_{22211} + \mathcal{E}_v(S_{22211})(T_N) + \mathcal{E}_w(S_{22211})(T_Q) \\ M_{22121} + \mathcal{E}_v(S_{22121})(T_N) + \mathcal{E}_w(S_{22121})(T_Q) \end{pmatrix}$$

Les contraintes n'ont pas changé, on peut donc dans tous les cas identifier ces coordonnées avec celles de la cible ($g_{u\#v\#w} = g_q$). On en déduit alors successivement Q , w , les $\mathcal{E}_w(S_i)$, on connaît alors N , ce qui permet de calculer v , donc les $\mathcal{E}_v(S_i)$, M est alors déterminé et u s'en déduit.

Tableau récapitulatif

Les traits **—** signifient que la variable en question a évolué librement au cours de l'étape concernée : on n'en a pas tenu compte lors de la résolution.

ξ	0	\xrightarrow{u} M	\xrightarrow{v} N	\xrightarrow{w} Q _{rég}	\xrightarrow{w} Q _{sing}
ξ_1	0	0	0	q_1	0
ξ_2	0	0	0	q_2	q_2
ξ_{21}	0	0	0	q_{21}	q_{21}
ξ_{211}	0	0	0	q_{211}	q_{211}
ξ_{21^3}	0	0	0	q_{21^3}	q_{21^3}
ξ_{21^4}	0	0	0	q_{21^4}	q_{21^4}
\vdots	\vdots	\vdots	\vdots		\vdots
ξ_{21^9}	0	$g_{21^9} \neq 0$	$h_{21^9} \neq 0$	q_{21^9}	q_{21^9}
ξ_{221}	0	0	h_{221}	—	—
ξ_{2211}	0	0	h_{2211}	—	—
ξ_{221^3}	0	0	h_{221^3}	—	—
ξ_{221^4}	0	$g_{221^4} \neq 0$	h_{221^4}	—	—
ξ_{21211}	0	$g_{21211} \neq 0$	h_{21211}	—	—
ξ_{2^31}	0	g_{2^31}	—	—	—
ξ_{2^311}	0	g_{2^311}	—	—	—
ξ_{22121}	0	g_{22121}	—	—	—

TAB. 6.1 - Planification de trajectoire des 19 équations

Chapitre 7

Identification de la sortie plate d'un sous système

7.1 Méthode d'interpolation

Les systèmes chaînés sont plats. La sortie linéarisante d'un système chaîné à 2 entrées est la suivante :

$$\xi_1, \xi_{21^n}$$

Ceci guide la mise au point de la méthode exposée ici.

Prenons dans un premier temps les équations d'un système chaîné à 4 états :

$$\begin{cases} \dot{\xi}_1 &= u_1 \\ \dot{\xi}_2 &= u_2 \\ \dot{\xi}_{21} &= u_1 \xi_2 \\ \dot{\xi}_{211} &= u_1 \xi_{21} \end{cases}$$

Sous la condition de non singularité $\xi_1 \neq 0$ on peut considérer les ξ_i et u_2 (on perd ainsi l'indépendance de u_2) comme fonctions de ξ_1 qui devient ainsi le paramètre. On posera donc :

$$\begin{cases} \xi_{211}(t) = X_{211}(\xi_1) \\ \xi_{21}(t) = X_{21}(\xi_1) \\ \xi_2(t) = X_2(\xi_1) \end{cases}$$

d'où il vient :

$$\begin{cases} \dot{\xi}_{211}(t) = \frac{d}{d\xi_1} X_{211} \dot{\xi}_1 \\ \dot{\xi}_{21}(t) = \frac{d}{d\xi_1} X_{21} \dot{\xi}_1 \\ \dot{\xi}_2(t) = \frac{d}{d\xi_1} X_2 \dot{\xi}_1 \end{cases}$$

les contraintes dynamiques du départ peuvent donc se réécrire :

$$\begin{cases} \frac{d}{d\xi_1} X_{211} = X_{21} \\ \frac{d}{d\xi_1} X_{21} = X_2 \\ \frac{d}{d\xi_1} X_2 \xi_1 = u_2 \end{cases}$$

et se résoudre par une simple interpolation polynomiale. On cherche alors pour $X(\xi_1) = (X_{211}(\xi_1), X_{21}(\xi_1), X_2(\xi_1))$ un polynôme $P(\xi_1)$ vérifiant 3 conditions au départ

$$X(0) = \xi(0) = (0, 0, 0)$$

et 3 autres à l'arrivée

$$\begin{aligned} X(\xi_1(T)) = X(q_1) &= \xi(T) \\ &= (q_2, q_{21}, q_{211}) \end{aligned}$$

qui sont résumées dans le tableau suivant :

	$t = 0$	\dots	$t = T$
$P = X_{211}$	0	\dots	q_{211}
$\frac{d}{d\xi_1} P = X_{21}$	0	\dots	q_{21}
$\frac{d^2}{d\xi_1^2} P = X_2$	0	\dots	q_2

toujours possible :

- si $q_1 \neq 0$, avec f de degré 5
- si $q_1 = 0$, en faisant 2 interpolations successives

Les entrées sont alors données par :

$$u_1 = \dot{\xi}_1, \quad u_2 = \left(\frac{d^3}{d\xi_1^3} f(\xi_1) \right) \dot{\xi}_1$$

ainsi en posant : $f(\xi_1) = a_0 + a_1 \xi_1 + \dots + a_5 \xi_1^5$, on trouve

$$\text{solution : } \left\{ \begin{aligned} a_0 &= 0, a_1 = 0, a_2 = 0, \\ a_3 &= \frac{1}{2} \frac{20 q_{211} - 8 q_1 q_{21} + q_1^2 q_2}{q_1^3}, \\ a_4 &= - \frac{15 q_{211} - 7 q_1 q_{21} + q_1^2 q_2}{q_1^4}, \\ a_5 &= \frac{1}{2} \frac{12 q_{211} - 6 q_1 q_{21} + q_1^2 q_2}{q_1^5} \end{aligned} \right\}$$

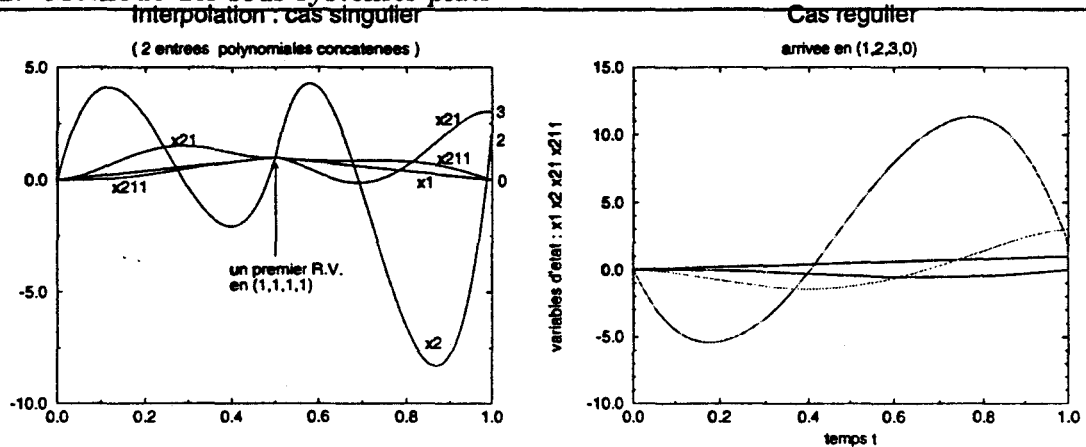


FIG. 7.1 - Méthode d'interpolation

Remarque :

- toutes les fonctions sont alors polynomiales
- la méthode revient à considérer un système avec dérive sur Y_1

Exemple : arrivée en $(1, 2, 3, 0)$ pour le cas régulier et arrivée en $(0, 2, 3, 0)$ pour le cas singulier (figure 7.1).

7.2 Méthode des sous systèmes plats

Le principe consiste à commencer la résolution pour le sous système plat (voir section 4.6) puis à considérer celui de la dernière équation. A l'ordre 3 les équations sont les suivantes :

$$\left\{ \begin{array}{l} \dot{\xi}_1 = u_1 \\ \dot{\xi}_2 = u_2 \\ \dot{\xi}_{21} = u_1 \xi_2 \\ \dot{\xi}_{211} = u_1 \xi_{21} \\ \dot{\xi}_{221} = \frac{1}{2!} u_1 \xi_2^2 \end{array} \right.$$

où toutes les fonctions dépendent du temps. La méthode d'identification des entrées consistera à prendre par exemple une entrée polynomiale en le temps, à intégrer les équations du système nilpotent libre et à identifier les coefficients en résolvant le système d'équations algébriques ainsi obtenu. Pour le système nilpotent libre à l'ordre 3 ces équations algébriques :

- 2 équations sont de degré total 1

- 1 de degré 2

- 2 de degré 3

Or les 4 premières équations de ce système forment un système plat de sortie plate (ξ_1, ξ_{211}) . On peut donc fixer la sortie plate du sous système plat formé des quatre premières équations (au lieu de l'entrée), par exemple en posant :

$$\xi_1(\alpha(t)) = \alpha(t) \quad , \quad \xi_{211} = f(\alpha(t))$$

les équations deviennent :

$$\begin{aligned} \xi_1(\alpha) &= \alpha \\ \xi_{211}(\alpha) &= f(\alpha) \\ \xi_{21}(\alpha) &= f'(\alpha) \\ \xi_2(\alpha) &= f''(\alpha) \\ \xi_{221}(\alpha) &= \frac{1}{2!} \int_0^\alpha f''(\tau)^2 d\tau \end{aligned}$$

Système que l'on peut tenter de résoudre de différentes manières. Celle retenue ici consiste à retrouver les entrées u_1 et u_2 en assignant une trajectoire polynomiale à (ξ_1, ξ_{211})

$$\begin{cases} s(t) &= a_1 t + a_0 \\ \xi_{211}(s) &= b_6 s^6 + b_5 s^5 + b_4 s^4 + b_3 s^3 \end{cases}$$

les 5 équations sont alors polynomiales :

- les quatre premières sont linéaires par rapport aux coefficients (les a_i et les b_i)
- la dernière de degré total 2

Cette méthode ne permet pas toutefois d'atteindre tout l'espace d'état car il est clair que :

$$\dot{\xi}_1(t) \geq 0 \quad \forall t \geq 0 \quad \Rightarrow \quad \xi_{221}(t) \geq 0 \quad \forall t \geq 0$$

Ainsi par exemple, en partant de l'origine, on ne pourra atteindre un point d'arrivée q tel que ses composantes (q_1, q_{221}) vérifient $q_1 > 0$ et $q_{221} < 0$ puisque u_1 est constante. On peut éviter cette difficulté en passant par un point intermédiaire. On décompose ainsi le mouvement en 2 trajectoires pour lesquelles :

$$\dot{\xi}_1 \dot{\xi}_{221} > 0$$

L'étude des équations qui se ramène à l'étude du signe d'un discriminant, montre qu'elles ont alors une solution, quel que soit le point d'arrivée : on peut atteindre tout l'espace d'état.

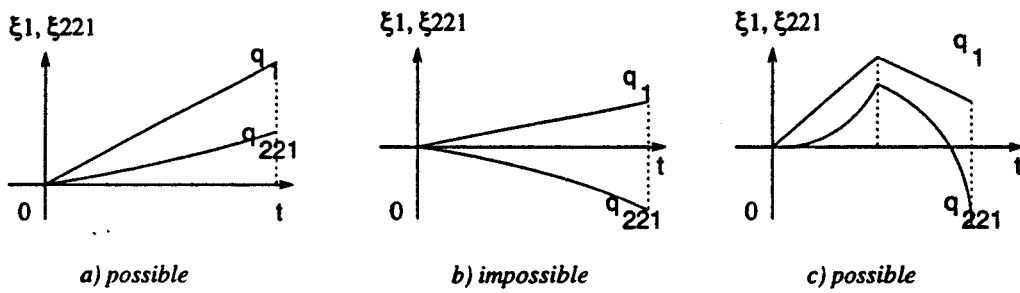


FIG. 7.2 - $q_1 q_{221} > 0$ possible en 1 morceau, $q_1 q_{221} < 0$ possible en 2 morceaux

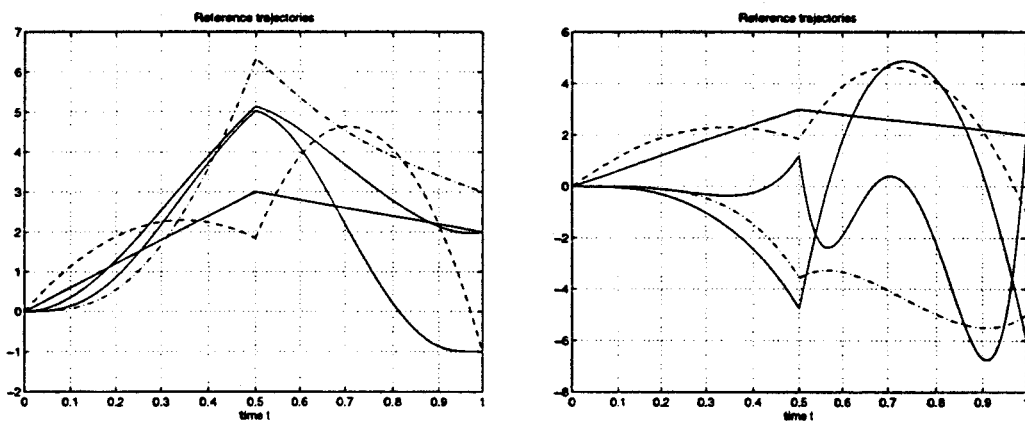


FIG. 7.3 - $\xi(t)$ (forme canonique) et $x(t)$ (approximation du "plate-ball")

Exemple de la main articulée

On a vu comment le système "plate-ball" approximé pouvait par changement de coordonnées polynomial se ramener au système nilpotent libre (4.6.1). Par ailleurs Si l'on fixe les contraintes suivantes pour la planification de trajectoire: un état initial $x(0) = (0, 0, 0, 0, 0)$, (resp. $\xi(0) = (0, 0, 0, 0, 0)$).

et un état final $x(1) = (2, -1, -6, -5, 2)$ (resp. $\xi(1) = (2, -1, 2, 3, -1)$).

On doit alors fixer un point intermédiaire, par exemple $(t = \frac{1}{2}, \alpha = 2)$, puis l'on cherche 2 fonctions polynomiales $f_1(\alpha)$ et $f_2(\alpha)$ de degré 4 (en α). La résolution des équations permet d'obtenir les commandes suivantes :

$$\begin{cases} u_1(t) = 6; u_2(t) = -38.08t + 13.21 \text{ pour } 0 \leq t < 0.5 \\ u_1(t) = -2; u_2(t) = -130.75t + 92.37 \text{ pour } 0.5 \leq t \leq 1 \end{cases}$$

La figure 7.3 donne les comportement des variables d'état du système correspondants.

Chapitre 8

Identification dans le groupe de Lie

On présente ici une méthode *linéaire* de résolution. Elle permet de trouver un produit de 6 exponentielles menant le système libre nilpotent à l'ordre 4 à toute cible. L'expression des résultats est particulièrement simple et assure au système de ne pas s'écarter loin de la cible avant d'y parvenir.

8.1 Intégrer un champ de vecteur à la fois

Lorsqu'on suit un champ de vecteur unique l'opérateur de transport est simplement le flot associé à ce champ. Par exemple si l'on suit une courbe intégrale du champ X_1 (resp. X_2) avec la loi de commande u_1 (resp. u_2) pendant T , u_1 est alors la vitesse de parcours de cette trajectoire et l'opérateur de transport est donné par :

$$\mathcal{H} = e^{\left(\int_0^T u_1 dt\right) X_1}$$

en posant $A_1 = \int_0^T u_1$ l'expression de \mathcal{H} devient :

$$\mathcal{H} = e^{A_1 X_1}$$

En alternant les champs de vecteurs suivant lesquels on se déplace, l'opérateur de transport est donc de la forme :

$$\mathcal{H} = e^{A_1 X_1} e^{B_2 X_2} \dots e^{A_{N-1} X_1} e^{B_N X_2}$$

Pour le système dynamique cela revient à prendre un vecteur d'entrée de la forme $u = (u_1, 0)$ ou $u = (0, u_2)$ (une des deux entrées nulle) dont la composante non nulle vérifie la condition sur l'intégrale.

8.2 Cas particuliers de la loi produit

Dans les 4 situations suivantes la loi du groupe libre nilpotent à l'ordre 4 est très simple :

$$\xi = p \circ g$$

Cas 1 : Notons la valeur initiale de l'état $\xi(0)$ et supposons la égale à $(p_1, p_2, p_{21}, p_{211}, p_{221})^T$. Si l'on suit le flot de X_1 partant de ce point pendant A_1 le point auquel on aboutit est calculé par le produit $\xi = p \circ g = \xi(0) \circ e^{A_1 X_1}$:

$$\begin{cases} \xi_1 &= A_1 + p_1 \\ \xi_2 &= p_2 \\ \xi_{21} &= A_1 p_2 + p_{21} \\ \xi_{211} &= A_1 (\frac{1}{2} A_1 p_2 + p_{21}) + p_{211} \\ \xi_{221} &= \frac{1}{2} A_1 p_2^2 + p_{221} \end{cases}$$

comme on le voit ici toutes les grandeurs sauf ξ_2 sont modifiées.

cas 2 (1 bis) : de plus, si la valeur initiale de ξ_2 est nulle ($p_2 = 0$) alors le système devient :

$$\begin{cases} \xi_1 &= A_1 + p_1 \\ \xi_2 &= 0 \\ \xi_{21} &= p_{21} \quad (\text{n'évolue pas}) \\ \xi_{211} &= A_1 p_{21} + p_{211} \\ \xi_{221} &= p_{221} \quad (\text{n'évolue pas}) \end{cases}$$

où seules les variables (ξ_1) et (ξ_{211}) évoluent.

cas 3 (1 ter) : Un autre cas particulier est celui où l'on part de conditions initiales nulles sur toutes les grandeurs sauf ξ_1 éventuellement $\xi(0) = (p_1, 0, 0, 0, 0)^T$. Dans ce cas de figure seul ξ_1 est modifiée :

$$\begin{cases} \xi_1 &= A_1 + p_1 \\ \xi_2 &= 0 \\ \xi_{21} &= 0 \\ \xi_{211} &= 0 \\ \xi_{221} &= 0 \end{cases}$$

cas 4 : En agissant sur la deuxième entrée seulement, i.e. en suivant une courbe intégrale du champ X_2 le système devient :

$$\begin{cases} \xi_1 &= p_1 \\ \xi_2 &= B_1 \\ \xi_{12} &= p_{21} \\ \xi_{112} &= p_{211} \\ \xi_{122} &= p_{221} \end{cases}$$

où seule la variable (ξ_2) évolue.

8.3 Algorithme.

On va donc utiliser des commandes constantes sur une entrée et nulles sur l'autre (cf. paragraphe précédent) en alternant le choix de l'entrée. On va régler successivement ξ_{21} et ξ_{221} dans le même temps puis ξ_{211} et ξ_1 , enfin ξ_2 , car cette grandeur peut être ajustée à n'importe quel moment. A noter que ξ_1 est en fait ajustée par anticipation (...). Une alternative consisterait à changer l'ordre: ajuster ξ_{211} avant ξ_{21} et ξ_{221} .

Le point de départ est l'élément neutre du groupe nilpotent libre.

1. $u = (a_1(t), 0)$, on pose $A_1 = \int_0^{T_1} a_1(t) dt$. Il s'agit du cas 3, le déplacement est donc le suivant :

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} A_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

ceci nous donne toute latitude pour régler ξ_1

2. $u = (0, b_2(t))$, on pose $B_2 = \int_0^{T_2-T_1} b_2(t) dt$, c'est le cas 4 :

$$\begin{pmatrix} A_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} A_1 \\ B_2 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

ceci afin de pouvoir au pas suivant modifier ξ_{21} , ξ_{211} et ξ_{221}

3. $u = (a_3(t), 0)$, on pose $A_3 = \int_0^{T_3-T_2} a_3(t) dt$. On applique les formules du cas 1 :

$$\begin{pmatrix} A_1 \\ B_2 \\ 0 \\ 0 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} A_1 + A_3 \\ B_2 \\ B_2 A_3 \\ \frac{1}{2} B_2 A_3^2 \\ \frac{1}{2} B_2^2 A_3 \end{pmatrix}$$

on ajuste ainsi ξ_{12} et ξ_{122} en résolvant :

$$\begin{aligned} B_2 A_3 &= q_{21} \\ B_2^2 A_3 &= 2q_{221} \end{aligned}$$

résolution toujours possible modulo la contrainte :

$$q_{21} q_{221} \neq 0$$

4. $u = (0, b_4(t))$, on pose $B_4 = \int_0^{T_4-T_3} b_4(t) dt$ (cas 4) :

$$\begin{pmatrix} A_1 + A_3 \\ B_2 \\ q_{21} \\ \frac{1}{2} B_2 A_3^2 \\ q_{221} \end{pmatrix} \longrightarrow \begin{pmatrix} A_1 + A_3 \\ B_2 + B_4 \\ q_{21} \\ \frac{1}{2} B_2 A_3^2 \\ q_{221} \end{pmatrix}$$

en prenant :

$$B_2 + B_4 = 0$$

on se ramène ainsi au cas 2 afin d'ajuster la composante ξ_{211}

5. $u = (a_5(t), 0)$, on pose $A_5 = \int_0^{T_5-T_4} a_5(t) dt$:

$$\begin{pmatrix} A_1 + A_3 \\ 0 \\ q_{21} \\ \frac{1}{2} B_2 A_3^2 \\ q_{221} \end{pmatrix} \longrightarrow \begin{pmatrix} A_1 + A_3 + A_5 \\ 0 \\ q_{21} \\ \frac{1}{2} B_2 A_3^2 + A_5 q_{21} \\ q_{221} \end{pmatrix}$$

pour acheminer la composante ξ_{211} du système à sa valeur cible q_{211} on pose alors :

$$A_5 = \frac{q_{211} - \frac{1}{2} B_2 A_3^2}{q_{21}}$$

où l'on retrouve la contrainte $q_{21} \neq 0$. Restent donc les deux premières composantes à positionner. Or la première composante, ξ_1 peut être ajustée au cours de cette même étape. Il suffit pour cela d'avoir bien anticiper lors du premier échelon, en prenant :

$$A_1 = q_1 - (A_3 + A_5)$$

le cas 4 permet finalement d'ajuster la deuxième composante lors d'un dernier mouvement.

6. $u = (0, b_6(t))$, on pose $B_6 = \int_0^{T_6-T_5} b_6(t) dt$:

$$\begin{pmatrix} q_1 \\ 0 \\ q_{21} \\ q_{211} \\ q_{221} \end{pmatrix} \longrightarrow \begin{pmatrix} q_1 \\ B_6 \\ q_{21} \\ q_{211} \\ q_{221} \end{pmatrix}$$

en posant simplement :

$$B_6 = q_2$$

La cible est donc atteinte génériquement en 6 mouvements.

8.3.1 Solution générique

Le bilan du schéma de résolution est le suivant :

Δ_i	$\int_{\Delta_i} u$	
	A_i	B_i
$[0, T_1[$	$q_1 - \frac{q_{211}}{q_{21}} - \frac{q_{21}^2}{4q_{221}}$	0
$[T_1, T_2[$	0	$\frac{2q_{221}}{q_{21}}$
$[T_2, T_3[$	$\frac{q_{21}^2}{2q_{221}}$	0
$[T_3, T_4[$	0	$-\frac{2q_{221}}{q_{21}}$
$[T_4, T_5[$	$\frac{q_{211}}{q_{21}} - \frac{q_{21}^2}{4q_{221}}$	0
$[T_5, T_6[$	0	q_2

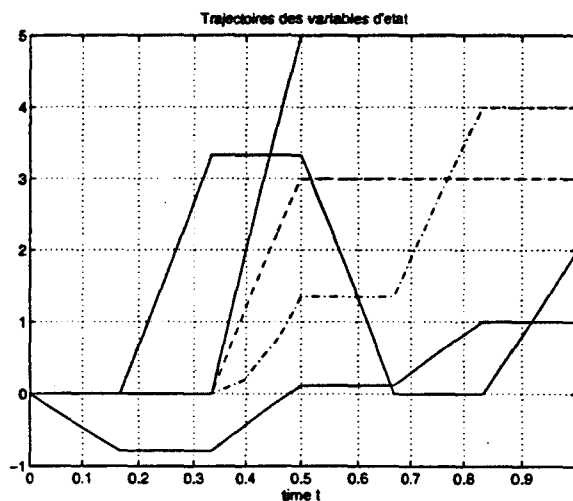


FIG. 8.1 - Cas régulier, arrivée en (1, 2, 3, 4, 5)

8.3.2 Solutions particulières

Dans les deux cas singuliers ($q_{21} = 0, q_{221} \neq 0$) et ($q_{21} \neq 0, q_{221} = 0$) on peut raisonner de la même façon mais la solution nécessite 2 exponentielles supplémentaires afin de ramener une composante à l'origine sans ramener les deux.

Le cas ($q_{21} = 0$ et $q_{221} = 0$) se résoud avec le même enchaînement de 8 exponentielles comme un cas particulier plus simple; il est en effet plus facile de ramener les deux composantes à 0 plutôt qu'une seule en utilisant cette méthode alternant les entrées. La méthode ne permet pas de trouver une solution linéaire en moins de 8 exponentielles.

Cas ($q_{21} = 0, q_{221} \neq 0$):

Δ_i	$\int_{\Delta_i} u$	
	A_i	B_i
$[0, T_1[$	$q_1 + \frac{q_{221}}{B_2 B_6} - \frac{q_{211}}{2q_{221}} (B_2 - B_6)$	0
$[T_1, T_2[$	0	B_2
$[T_2, T_3[$	$\frac{2q_{221}}{B_2(B_2 - B_6)}$	0
$[T_3, T_4[$	0	$-B_2$
$[T_4, T_5[$	$\frac{q_{221}}{B_2 B_6} + \frac{q_{211}}{2q_{221}} (B_2 - B_6)$	0
$[T_5, T_6[$	0	B_6
$[T_6, T_7[$	$-\frac{2q_{221}}{B_6(B_2 - B_6)}$	0
$[T_7, T_8[$	0	$q_2 - B_6$

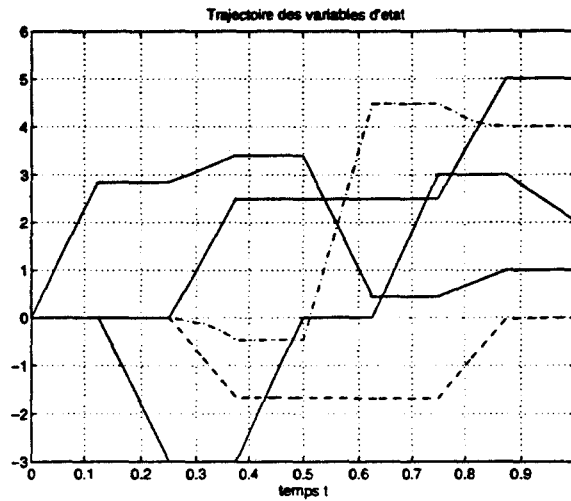
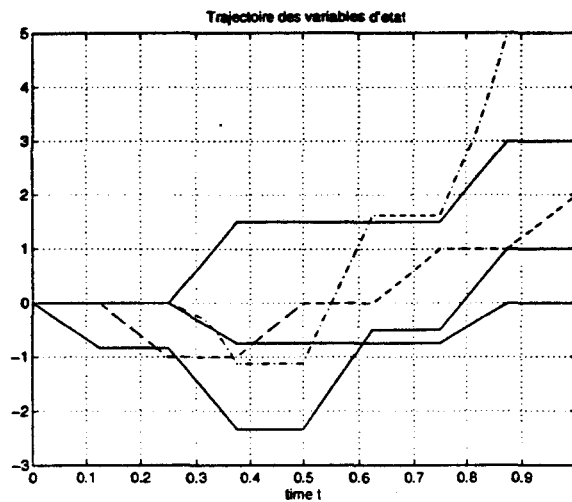


FIG. 8.2 - Cas singulier, arrivée en $(1, 2, \xi_{21} = 0, 4, 5)$

Cas ($q_{21} \neq 0, q_{221} = 0$):

Δ_i	$\int_{\Delta_i} u$	
	A_i	B_i
$[0, T_1[$	$q_1 - \frac{q_{211}}{q_{21}} \left(1 - \frac{B_2}{B_6}\right) - \frac{q_{21}}{2B_2} \left(1 + \frac{B_2}{B_6} + \frac{B_2^2}{B_6^2}\right)$	0
$[T_1, T_2[$	0	B_2
$[T_2, T_3[$	$\frac{q_{21}}{B_2} \left(1 - \frac{B_2}{B_6}\right)^{-1}$	0
$[T_3, T_4[$	0	$-B_2$
$[T_4, T_5[$	$\frac{q_{211}}{q_{21}} \left(1 - \frac{B_2}{B_6}\right) - \frac{q_{21}}{2B_2} \left(1 + \frac{B_2}{B_6} - \frac{B_2^2}{B_6^2}\right)$	0
$[T_5, T_6[$	0	B_6
$[T_6, T_7[$	$\frac{q_{21}}{B_6} \left(1 - \frac{B_6}{B_2}\right)^{-1}$	0
$[T_7, T_8[$	0	$q_2 - B_6$

FIG. 8.3 - Cas singulier, arrivée en $(1, 2, 3, 5, \xi_{221} = 0)$

Cas $(q_{21} = 0, q_{221} = 0)$:

Δ_i	$\int_{\Delta_i} u$	
	A_i	B_i
$[0, T_1[$	$q_1 - \frac{q_{211}}{B_2 A_3}$	0
$[T_1, T_2[$	0	B_2
$[T_2, T_3[$	A_3	0
$[T_3, T_4[$	0	$-B_2$
$[T_4, T_5[$	$\frac{q_{211}}{B_2 A_3}$	0
$[T_5, T_6[$	0	B_2
$[T_6, T_7[$	$-A_3$	0
$[T_7, T_8[$	0	$q_2 - B_2$

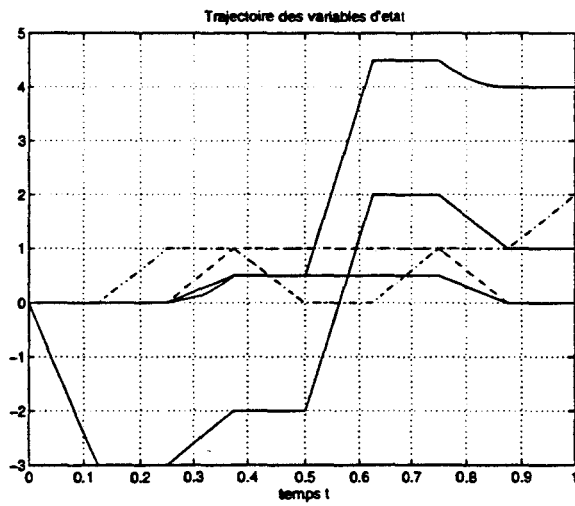


FIG. 8.4 - Cas singulier, arrivée en $(1, 2, \xi_{21} = 0, 4, \xi_{221} = 0)$

Chapitre 9

Conclusion

La planification de trajectoire des systèmes nilotents peut être ramenée à celle des systèmes nilpotents *libres*. Dans ce dernier cas, trois approches ont été dégagées que nous rappelons ici dans l'ordre d'apparition du mémoire.

L'identification de l'entrée consiste à chercher une entrée polynomiale ou constante par morceaux, à coefficients indéterminés, puis à calculer les coordonnées exponentielles du point de l'espace des configurations où elle mène le système et finalement à identifier ces coordonnées avec celles de la cible en résolvant le système d'équations algébriques ainsi obtenu. Pour le premier système nilpotent libre intéressant car non plat $\Sigma NL(4)$, qui possède 5 équations, cette méthode nous a permis d'étudier complètement des solutions minimales en terme du degré des polynômes ou du nombre d'inconnues nécessaires (les coefficients des entrées, les valeurs des entrées constantes). Par la même méthode, la résolution des équations a pu être poussée jusqu'à 19 équations ceci incluant les 14 équations de $\Sigma NL(6)$. De telles solutions s'avèrent cependant peu stables d'un point de vue numérique et provoquent en général des détours importants pour le système qui s'éloigne beaucoup avant d'atteindre la cible.

L'identification de la sortie plate d'un sous système plat consiste à traiter dans un premier temps les équations de la partie plate –ce qui peut se faire de façon linéaire– puis à essayer de traiter les équations restantes. Cette approche a permis de faire le lien avec la platitude et de mieux cerner les difficultés (entendre par là qu'elle met en évidence la partie non plate du système). Les résultats obtenus pour $\Sigma NL(4)$ sont cependant comparables à ceux obtenus avec des entrées polynomiales par la méthode d'identification, les travaux entrepris dans ce cadre ayant amené à une stratégie de résolution assez similaire. Il apparaît en effet que ces deux méthodes reviennent au même dans le cas d'un système chaîné.

L'identification dans le groupe de Lie est la dernière approche que nous avons étudiée. Elle consiste à trouver directement un élément du groupe de Lie nilpotent libre (et non plus une commande) menant le système à la cible, en composant alter-

nativement des exponentielles d'un —à la fois— des champs de vecteurs initiaux, à la manière des intégrateurs symplectiques :

$$\exp(A_1 Y_1) \exp(B_2 Y_2) \exp(A_3 Y_1) \exp(B_4 Y_2) \dots \exp(A_{N-1} Y_1) \exp(B_N Y_2)$$

Les solutions ainsi obtenues sont composées de 6 exponentielles pour le cas générique, 8 pour les cas singuliers. Elles sont aussi les plus simples formellement et l'on est assuré par construction d'un bon comportement dynamique pour le système (pas de détours incontrôlés). Par ailleurs, les inconnues (A_i et B_j) représentent le temps durant lequel on suit le champ de vecteur correspondant, seule importe l'aire algébrique de la fonction d'entrée, on peut donc envisager des raccords C^1 ou plus entre chaque étape.

Les perspectives de ce travail sont tout d'abord la *généralisation de la méthode des exponentielles alternées à des systèmes plus importants* quant au nombre des équations. On souhaitera également trouver des méthodes pour rendre les commandes existantes des grands systèmes plus stables (éviter les détours).

Dans les domaines plus théoriques de nombreuses questions restent en suspens. On ne sait pas par exemple *décider si un système est nilpotentisable par bouclage*. Le lien entre nos travaux et ceux de Hermès [16, 17], Bellaïche, Laumond, Risler [1], sur les *approximations nilpotentes*, reste également à faire. Enfin on souhaiterait également étendre ces résultats au *cas des systèmes dont l'algèbre de Lie de commande est de présentation finie*.

Annexe A

Exemple de système plat

A.1 Définition

Un système est dit plat si les conditions suivantes sont vérifiées :

1. il existe un ensemble fini de variables $y = (y_1, \dots, y_p)$ différentiellement indépendantes (qui ne sont liées, entre elles, par aucune équation algébrico-différentielle) : elles forment une sortie plate ou linéarisante
2. toute variable dynamique du système est solution d'une équation algébrique en les y_i et leur dérivées

A.2 Exemple : la pièce est plate

En effet une sortie linéarisante pour ce système a pu être trouvée ([39]), elle est donnée par les coordonnées du point (I) : x_I et y_I (voir figure

$$\left\{ \begin{array}{l} \frac{1}{\phi} = \frac{\dot{x}_I \ddot{y}_I - \dot{y}_I \ddot{x}_I}{\sqrt{\dot{x}_I^2 + \dot{y}_I^2}} \\ \tan \theta = \frac{-\dot{x}_I}{\dot{y}_I} \\ x = x_I + \phi \cos \theta \\ y = y_I + \phi \sin \theta \\ u_0 = \dot{\theta} \\ u_1 = \dot{\phi} \end{array} \right.$$

A.3 La planification de trajectoire :

on choisit une trajectoire pour (I) vérifiant les conditions sur la courbure et la tangente en $I(0)$ et $I(T)$,

les autres variables du système –dont l'entrée– se déduisent algébriquement (c'est une propriété des systèmes plats) de $I(t)$, $\dot{I}(t)$ et $\ddot{I}(t)$ (cf équations ci-dessus).

Remarques : les conditions sur la tangente et la courbure interdisent les trajectoires rectilignes (pour celles-ci le point (I) ne bouge pas), on note également que si le choix de la trajectoire de la sortie linéarisante (ici le point I) est libre aux conditions de raccordement près, trajectoire réelle du système ne peut être choisie.

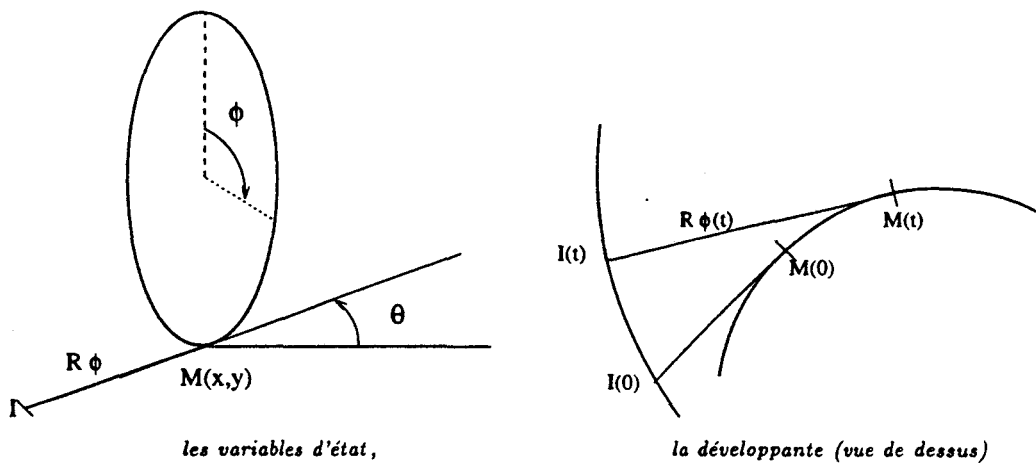


FIG. A.1 - la pièce et la développante de sa trajectoire

Annexe B

Programmes Axiom

B.1 Les S_l

-- Address comments and questions to the

--

-- L I F L Laboratoire d'informatique fondamentale de LILLE
 -- Bat M3 U.F.R d' I.E.E.A
 -- 59655 VILLENEUVE D'ASCQ Cedex
 -- tel: 20-43-44-92

--

-- resp. : G. JACOB
 -- prog. : C. GUYON & M. PETITOT

--

-- last modif.: Dim 14 Fev 1993

+++++

++

++

++ Lyndon Words constructed with an arbitrary alphabet

++ 16 may 1991 -- Release 1.0 --

++

++ The 3 following properties shall be used to construct Lyndon Words

++ 1) the letters of the ordered alphabet they are build with
 ++ are Lyndon Words

++ 2) if a,b are Lyndon Words verifying: $a < b$
 ++ then $a*b$ is again a Lyndon Word, where "*" holds for
 ++ concatenation (local function "concat")

++ 3) if a,b,l,m are Lyndon Words verifying
 ++ $a = l*m$, i.e. $[l,m]$ is its standard factorisation
 ++ then $a*b$ is a Lyndon Word of standard factorisation $[a,b]$
 ++ iff $a < b$ (2) and $b < m$

++ Other properties:

++ i) a Lyndon Words is smaller than every right factor of it
 ++ ii) every word on the same alphabet shall be written uniquely as
 ++ a decreasing product of Lyndon Words

++ iii) the order ($<$) used here is the same as for a dictionary
 ++ (lexicographical order)

++ (the lexicographic by length" order is also implemented :
 ++ the function name is "lexicoByLength")

++

+++++

)abbrev domain LWORDPBW LyndonWordPBW

LyndonWordPBW(VarSet:OrderedSet):Public == Private where

```

OFMON ==> OrderedFreeMonoid( VarSet )
MAGM  ==> Magma( VarSet )
PI    ==> PositiveInteger
NNI   ==> NonNegativeInteger
I     ==> Integer
FACT_FORM ==> Record( first:VarSet, rest:List Record( gen:$, exp:NNI ) )

```

```

Public == Join(OrderedSet, RetractableTo VarSet) with

```

```

left :      $ -> $
right :     $ -> $
factorise : OFMON -> List $           -- prop ii)
length :    $ -> PI
length :    ($,VarSet) -> NNI         -- other kind of length
lexicoByLength :
            ($,$) -> Boolean          -- another order on lwords

```

```

-- words sometimes are Lyndon
Lyndon:      OFMON -> $
Lyndon?:    OFMON -> Boolean
LyndonIfCan: OFMON -> Union($,"failed")

```

```

-- generating Lyndon Words of finite size
LyndonWordsList : (List VarSet,PI) -> List $
LyndonWordsByLength : (List VarSet,PI) -> Vector List $

```

```

-- Lyndon's sometimes are Retractable
retractable? : $ -> Boolean

```

```

-- to access the implementation
aFactoredForm: $ -> FACT_FORM
prettyFactForm:$ -> OutputForm

```

```

-- coercions
convertMagma:  $ -> MAGM
coerce:       $ -> OFMON
coerce:       $ -> OutputForm
coerce:       $ -> MAGM

```

Private == add

```

-- Representation.....
-----
-- iD in this Representation : extract the first letter
-- factor the rest which isn't a Lyndon Word anymore (prop.ii)
-- This may be usefull to Control Computations
-----
LFMON ==> OrderedFreeMonoid($)    -- words on an infinite
                                   -- alphabet : Lyndon Words
Rep:= Record( first : VarSet, rest : LFMON )

-- Local Signatures.....
-----
concat:      ($,$) -> $
standardise: (List $, List $) -> List $
letterList:  OFMON -> List $
concatFrom:  (List $, List $) -> List $

-- Definitions.....
-----
-- another order : a is smaller than b if
--   a is shorter
-- or a is the same length but smaller according
-- to the dictionary
-----
lexicoByLength( a, b ) ==
  length a = length b => a < b
  length a < length b

-----
-- the following just "appends" the lists in the vector
-- produced by the "LyndonWordsByLength" function
-----
LyndonWordsList( v1, n ) ==
  ll : Vector List $ := LyndonWordsByLength(v1,n)
  sort( [ : ll.i for i in 1..n ] )
-----

```

```

-- the following generates a vector of dimension (n+1)
-- where the ith component is the list of all valid
-- Lyndon Words of size i using the "VarSet" alphabet
-- (the first component, base.0, is not used)
-----

LyndonWordsByLength( vl, n ) ==
  null vl => error "empty list not an alphabet"
  base: Vector List $ := empty()

  -- words of size equal to 1 (prop.1)
  base.1 := [ w::$ for w in sort( vl ) ]

  -- words of size equal to ll
  for ll in 2..n:I repeat
    base.ll:=
      sort([ :concatFrom( base.i, reverse base.(ll-i) )
            for i in 1..(ll-1) ] )
  return base

-----

-- L1 and L2 are supposed to be sorted lists of $
-- respectively increasing and decreasing as respect to "<"
-- the following generates (in a list), all Lyndon Words l1*l2
-- l1 in L1, l2 in L2 according to properties 2) and 3)
-----

concatFrom( L1, L2 ) ==
  null L1 or null L2 => empty()
  not L1.first < L2.first => empty()  -- prop.2) & (L1,L2):sorted
  l:$ := L1.first
  nP3:Boolean := ( retractable? l => false;
    nthFactor( l.rest, size l.rest ) < L2.first) -- tests prop.3)
  [: (if nP3 then []::List($)
    else [ concat( L1.first, L2.first )]$List($)),
    :concatFrom( [ L1.first ], rest L2 ),  -- along with the
    :concatFrom( rest L1 , L2 ) ]        -- other elements

-----

-- the next 3 functions are used to compute the unique
-- factorisation of any word (on the same alphabet),
-- in a decreasing product of Lyndon Words (prop.ii)
-----

factorise (w) ==
  w = 1::OFMON => []

```

```

standardise( reverse letterList w, [] )

-----
-- words sometimes are Lyndon
-----

LyndonIfCan( w )==
  f:List($):= factorise w
  ( (#f = 1) => first f ; "failed" )

Lyndon? w ==
  #( factorise w ) = 1

Lyndon w ==
  f:List($):= factorise w
  #( f ) = 1 => first f
  error "LWORDPBW.Lyndon : not a Lyndon Word"

-----
-- turn the word into a List of letters
-----

letterList( w ) ==
  [ :[ r.gen :: $ for i in 1..r.exp ]
    for r in factors w ]

-----
-- Reuteunauer's algo :
-- given m a reversed list of $ if one encounters an
-- "inversion" beginning from left (right if not reversed)
-- i.e. l.i < l.(i+1) they become one, type $ (l.i*l.(i+1),
-- where "concat"~="*"),
-----

standardise( m, l ) ==
  null m => l
  null l => standardise( rest m, [ m.first ]$List($ ) )
  not m.first < l.first => standardise( rest m, [ m.first , :l ] )
  standardise( [ concat( m.first, l.first ), : rest m ], rest l )

-----
-- the following provides a mean to access the implementation
-----

aFactoredForm a ==
  [ a.first, factors.a.rest ]

```



```

-----
-- displaying LyndonWords with "(.,.)"
-- according to the standard factorisation (l,r)
-----
convertMagma( a:$ ) ==
  retractable? a => retract( a )::MAGM
  l:$:= left a
  r:$:= right a
  convertMagma( l ) *$MAGM convertMagma( r )

-----
-- to really see what the representation is
-----
prettyFactForm a ==
  retractable? a => retract( a )::OutputForm
  f:OutputForm:= a.first ::OutputForm
  s:List OutputForm:=[]
  for w in factors a.rest repeat
    out:OutputForm:= prettyFactForm w.gen
    if w.exp > 1 then out:= super( out, outputForm w.exp )
    s:= cons( out, s )
  s := reverse s
  paren hconcat [ f , :s ]

-----
concat( a, b ) ==
  [ a.first, a.rest * b::LFMON ]$Rep

-----
-- standardFactorisationOf a = [left a, right a]
-----
left a:$ ==
  r: LFMON
  if retractable? a
    then r := a.rest
    else r := rquo( a.rest,
                    factors( a.rest ).last.gen )::LFMON
  [ a.first, r ]

right a ==
  retractable? a =>
    error("LWORDPBW.right : illegal use : parameter is letter" )

```

```

factors( a.rest ).last.gen

-----
-- number of occurrences of a letter (z) in a Lyndon
-----
length( a:$, z:VarSet ) ==
  if a.first = z then
    cpt:NNI:=1
  else
    cpt:NNI:=0
  retractable? a => cpt
  cpt + +/[ length( b.gen, z ) * b.exp
    for b in factors a.rest ]

-----
-- the following returns the number of letters
-----
length a ==
  retractable? a => 1
  ( 1 + +/[ length( r.gen ) * r.exp
    for r in factors a.rest ] )::PI

-----
-- letters are Lyndon and Lyndon sometimes letters
-----
retractIfCan a ==
  retractable? a => retract a
  "failed"
retract a == a.first
retractable? a == a.rest = 1

-----
-----
a < b ==
  a.first = b.first => a.rest <$LFMON b.rest
  a.first <$VarSet b.first

-----
-- coercions .....
-----
coerce(v:VarSet): $ == [ v, 1 ]$Rep

coerce(x:$): OFMON ==

```

```

w:OFMON:= */[( r.gen::OFMON )** r.exp
  for r in factors x.rest ]
x.first *$OFMON w
coerce(x:$): OutputForm == prettyFactForm x
coerce(x:$): MAGM == convertMagma x

```

B.2 Evaluation des $S_l: \mathcal{E}(S_l)$

```
-- Adress comment and questions to the
```

```
--
```

```
-- L I F L   Laboratoire d'informatique fondamentale de LILLE
--           Bat M3   U.F.R d' I.E.E.A
--           59655 VILLENEUVE D'ASCQ Cedex
--           tel: 20-43-69-40
--
```

```
-- resp.      : G. JACOB
-- prog.      : C. GUYON & M. PETITOT
--
```

```
-- last modif.: 12 Mars 1995
--
```

```

+++++
++
++ This Package is aimed at the Evaluation Transform
++ of certain polynomial.
++ The Evaluation Transform operation decodes some coding word
++ and then computes some iterated integral
++
+++++

```

```
)abbrev package EVSL EvalS1
```

```

EvalS1( VarSet, K, Input, integrate ): Public == Private where
  VarSet      : OrderedSet
  K           : Field
  Input       : Algebra(K)
  integrate   : Input -> Input

  RN          ==> Fraction Integer
  Lyndon      ==> LyndonWordPBW( VarSet )
  Code        ==> Record( cod: VarSet, fct: Input )
  LCode       ==> List( Code )

```

```
Public == with
```

```
    makeInput : ( VarSet, Input ) -> Code
```

```
    evSl      : ( Input, Lyndon, LCode ) -> Input
```

```
    evSl      :      ( Lyndon, LCode ) -> Input
```

```
Private == add
```

```
    ICF ==> IntegerCombinatoricFunctions
```

```
-- Local Signatures.....
```

```
    decode : ( VarSet, LCode ) -> Input
```

```
-- Definitions.....
```

```
-- finding in l the function associated to a letter a
```

```
    decode( a:VarSet, l:LCode ): Input ==
```

```
        inp : Input
```

```
        if a = l.first.cod
```

```
            then inp := l.first.fct
```

```
            else inp := decode( a, l.rest )
```

```
-- making a code given a letter and a function
```

```
    makeInput( a, inp ) == [a, inp ]$Code
```

```
-- computes the iterated integral associated to
-- the element S(rl) of dual PBWL where rl is the mirror
-- image of the Lyndon word given as parameter
-- This version allows use of kernel function (fn)
-- as first integration operator
```

```
    evSl( fn, lw, cl ) ==
```

```
        retractable? lw => integrate( fn * decode( retract lw, cl ) )
```

```
        innerEval : Input := reduce("*",
```

```
            [ evSl( (1/ factorial( lwi.exp ))::RN::K *fn, lwi.gen, cl
              )**lwi.exp
```

```
                for lwi in aFactoredForm( lw ).rest ])  
                evS1( innerEval, aFactoredForm( lw ).first::Lyndon, cl )  
-----  
-- computation without kernel (equivalent to kernel=id)  
-----  
evS1( lw, cl ) ==  
    evS1( 1::Input, lw, cl )
```

Annexe C

Calculs dans le groupe nilpotent libre

Loi du groupe nilpotent libre a l'ordre 6 (tous les crochets de longueur supérieure ou égale à six sont nuls):

$$f = g * h \quad , \text{avec} \begin{cases} g = (g_1, \dots, g_{21}, \dots, g_2)^T \\ h = (h_1, \dots, h_{21}, \dots, h_2)^T \end{cases}$$

Les coordonnées exponentielles de f sont :

$$\begin{aligned}
 f_{[1]} &= h_{[1]} + g_{[1]} \\
 f_{[2 \ 1^4]} &= h_{[2 \ 1^4]} + \frac{1}{24} g_{[2]} h_{[1]}^4 + \frac{1}{6} g_{[2 \ 1]} h_{[1]}^3 + \frac{1}{2} g_{[2 \ 1^2]} h_{[1]}^2 + g_{[2 \ 1^3]} h_{[1]} + g_{[2 \ 1^4]} \\
 f_{[2 \ 1^3]} &= h_{[2 \ 1^3]} + \frac{1}{6} g_{[2]} h_{[1]}^3 + \frac{1}{2} g_{[2 \ 1]} h_{[1]}^2 + g_{[2 \ 1^2]} h_{[1]} + g_{[2 \ 1^3]} \\
 f_{[2 \ 1^2]} &= h_{[2 \ 1^2]} + \frac{1}{2} g_{[2]} h_{[1]}^2 + g_{[2 \ 1]} h_{[1]} + g_{[2 \ 1^2]} \\
 f_{[2 \ 1 \ 2 \ 1^2]} &= h_{[2 \ 1 \ 2 \ 1^2]} + (g_{[2]} h_{[1]} + g_{[2 \ 1]}) h_{[2 \ 1^2]} + \frac{1}{4} g_{[2]^2} h_{[1]}^3 \\
 &\quad + \left(\frac{1}{2} g_{[2^2 \ 1]} + \frac{1}{2} g_{[2]} g_{[2 \ 1]} \right) h_{[1]}^2 + \left(g_{[2^2 \ 1^2]} + \frac{1}{2} g_{[2 \ 1]^2} \right) h_{[1]} + g_{[2 \ 1 \ 2 \ 1^2]} \\
 f_{[2 \ 1]} &= h_{[2 \ 1]} + g_{[2]} h_{[1]} + g_{[2 \ 1]} \\
 f_{[2^2 \ 1^3]} &= h_{[2^2 \ 1^3]} + g_{[2]} h_{[2 \ 1^3]} + \frac{1}{12} g_{[2]^2} h_{[1]}^3 + \frac{1}{2} g_{[2^2 \ 1]} h_{[1]}^2 + g_{[2^2 \ 1^2]} h_{[1]} + g_{[2^2 \ 1^3]} \\
 f_{[2^2 \ 1^2]} &= h_{[2^2 \ 1^2]} + g_{[2]} h_{[2 \ 1^2]} + \frac{1}{4} g_{[2]^2} h_{[1]}^2 + g_{[2^2 \ 1]} h_{[1]} + g_{[2^2 \ 1^2]} \\
 f_{[2^2 \ 1 \ 2 \ 1]} &= h_{[2^2 \ 1 \ 2 \ 1]} + \frac{1}{2} g_{[2]} h_{[2 \ 1]^2} + \left(\frac{1}{2} g_{[2]^2} h_{[1]} + g_{[2^2 \ 1]} \right) h_{[2 \ 1]} + \frac{1}{6} g_{[2]^3} h_{[1]}^2 \\
 &\quad + \left(-g_{[2^3 \ 1]} + g_{[2]} g_{[2^2 \ 1]} \right) h_{[1]} + g_{[2^2 \ 1 \ 2 \ 1]} \\
 f_{[2^2 \ 1]} &= h_{[2^2 \ 1]} + g_{[2]} h_{[2 \ 1]} + \frac{1}{2} g_{[2]^2} h_{[1]} + g_{[2^2 \ 1]} \\
 f_{[2^3 \ 1^2]} &= h_{[2^3 \ 1^2]} + g_{[2]} h_{[2^2 \ 1^2]} + \frac{1}{2} g_{[2]^2} h_{[2 \ 1^2]} + \frac{1}{12} g_{[2]^3} h_{[1]}^2 + g_{[2^3 \ 1]} h_{[1]} + g_{[2^3 \ 1^2]} \\
 f_{[2^3 \ 1]} &: h_{[2^3 \ 1]} + g_{[2]} h_{[2^2 \ 1]} + \frac{1}{2} g_{[2]^2} h_{[2 \ 1]} + \frac{1}{6} g_{[2]^3} h_{[1]} + g_{[2^3 \ 1]} \\
 f_{[2^4 \ 1]} &= h_{[2^4 \ 1]} + g_{[2]} h_{[2^3 \ 1]} + \frac{1}{2} g_{[2]^2} h_{[2^2 \ 1]} + \frac{1}{6} g_{[2]^3} h_{[2 \ 1]} + \frac{1}{24} g_{[2]^4} h_{[1]} + g_{[2^4 \ 1]} \\
 f_{[2]} &: h_{[2]} + g_{[2]}
 \end{aligned}$$

Les composantes du produit dans l'ordre lexicographique par longueur

$$f_{[2]} : h_{[2]} + g_{[2]}$$

$$f_{[1]} = h_{[1]} + g_{[1]}$$

$$f_{[2 \ 1]} = h_{[2 \ 1]} + g_{[2]} h_{[1]} + g_{[2 \ 1]}$$

$$f_{[2^2 \ 1]} = h_{[2^2 \ 1]} + g_{[2]} h_{[2 \ 1]} + \frac{1}{2} g_{[2]}^2 h_{[1]} + g_{[2^2 \ 1]}$$

$$f_{[2 \ 1^2]} = h_{[2 \ 1^2]} + \frac{1}{2} g_{[2]} h_{[1]}^2 + g_{[2 \ 1]} h_{[1]} + g_{[2 \ 1^2]}$$

$$f_{[2^3 \ 1]} : h_{[2^3 \ 1]} + g_{[2]} h_{[2^2 \ 1]} + \frac{1}{2} g_{[2]}^2 h_{[2 \ 1]} + \frac{1}{6} g_{[2]}^3 h_{[1]} + g_{[2^3 \ 1]}$$

$$f_{[2^2 \ 1^2]} = h_{[2^2 \ 1^2]} + g_{[2]} h_{[2 \ 1^2]} + \frac{1}{4} g_{[2]}^2 h_{[1]}^2 + g_{[2^2 \ 1]} h_{[1]} + g_{[2^2 \ 1^2]}$$

$$f_{[2 \ 1^3]} = h_{[2 \ 1^3]} + \frac{1}{6} g_{[2]} h_{[1]}^3 + \frac{1}{2} g_{[2 \ 1]} h_{[1]}^2 + g_{[2 \ 1^2]} h_{[1]} + g_{[2 \ 1^3]}$$

$$f_{[2^4 \ 1]} = h_{[2^4 \ 1]} + g_{[2]} h_{[2^3 \ 1]} + \frac{1}{2} g_{[2]}^2 h_{[2^2 \ 1]} + \frac{1}{6} g_{[2]}^3 h_{[2 \ 1]} + \frac{1}{24} g_{[2]}^4 h_{[1]} + g_{[2^4 \ 1]}$$

$$f_{[2^3 \ 1^2]} = h_{[2^3 \ 1^2]} + g_{[2]} h_{[2^2 \ 1^2]} + \frac{1}{2} g_{[2]}^2 h_{[2 \ 1^2]} + \frac{1}{12} g_{[2]}^3 h_{[1]}^2 + g_{[2^3 \ 1]} h_{[1]} + g_{[2^3 \ 1^2]}$$

$$f_{[2^2 \ 1 \ 2 \ 1]} = h_{[2^2 \ 1 \ 2 \ 1]} + \frac{1}{2} g_{[2]} h_{[2 \ 1]^2} + \left(\frac{1}{2} g_{[2]}^2 h_{[1]} + g_{[2^2 \ 1]} \right) h_{[2 \ 1]} + \frac{1}{6} g_{[2]}^3 h_{[1]}^2 \\ + \left(-g_{[2^3 \ 1]} + g_{[2]} g_{[2^2 \ 1]} \right) h_{[1]} + g_{[2^2 \ 1 \ 2 \ 1]}$$

$$f_{[2^2 \ 1^3]} = h_{[2^2 \ 1^3]} + g_{[2]} h_{[2 \ 1^3]} + \frac{1}{12} g_{[2]}^2 h_{[1]}^3 + \frac{1}{2} g_{[2^2 \ 1]} h_{[1]}^2 + g_{[2^2 \ 1^2]} h_{[1]} + g_{[2^2 \ 1^3]}$$

$$f_{[2 \ 1 \ 2 \ 1^2]} = h_{[2 \ 1 \ 2 \ 1^2]} + \left(g_{[2]} h_{[1]} + g_{[2 \ 1]} \right) h_{[2 \ 1^2]} + \frac{1}{4} g_{[2]}^2 h_{[1]}^3 \\ + \left(\frac{1}{2} g_{[2^2 \ 1]} + \frac{1}{2} g_{[2]} g_{[2 \ 1]} \right) h_{[1]}^2 + \left(g_{[2^2 \ 1^2]} + \frac{1}{2} g_{[2 \ 1]^2} \right) h_{[1]} + g_{[2 \ 1 \ 2 \ 1^2]}$$

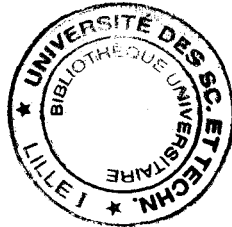
$$f_{[2 \ 1^4]} = h_{[2 \ 1^4]} + \frac{1}{24} g_{[2]} h_{[1]}^4 + \frac{1}{6} g_{[2 \ 1]} h_{[1]}^3 + \frac{1}{2} g_{[2 \ 1^2]} h_{[1]}^2 + g_{[2 \ 1^3]} h_{[1]} + g_{[2 \ 1^4]}$$

Bibliographie

- [1] M. Chyba. A. Bellaïche, J-P. Laumond. Canonical nilpotent approximation of control systems: Application to nonholonomic motion planning. In *Proc of C.D.C. 32th*, pages 2694–2699, San Antonio, Texas, December 1993.
- [2] M. Bigotte. *Implantation des Algèbres de Lie sous Maple*. Mémoire de d.e.a., Université Lille I, juin 1995.
- [3] J. E. Campbell. On a law of combination of operators (second paper). *Proceedings of the London Mathematical Society*, 1(29):14–32, 1898.
- [4] W.L. Chow. *Über Systeme von linearen partiellen Differentialgleichungen erster Ordnung*, volume 117, pages 98–105. *Math. Ann.*, 1939.
- [5] B. d'Andréa Novel. *Commande non linéaire des robots*. Traité des Nouvelles Technologies (Série Automatique). HERMES, 1988.
- [6] E.B. Dynkin. On the representation by means of commutators of the series $\log(e^x e^y)$ for noncommutative x and y . *Matematicheskii Sbornik*, 25(67):155–162, 1949.
- [7] E.B. Dynkin. Normed Lie algebras and analytic groups. *Uspekhi Matematicheskikh Nauk*, 5(1 (35)):135–186, 1950. (In Russian). English translation in: Translation series 1, Vol.9, Lie groups. American Mathematical Society (1962), 470–534.
- [8] M. Fliess. Fonctionnelles causales non linéaires et indéterminées non commutatives. *Bull. Soc. Math. France*, 109:3–40, 1981.
- [9] M. Fliess. Automatique et corps différentiels. *Forum Mathematicum*, 1:227–238, 1989.
- [10] M. Fliess and S.T. Glad. An algebraic approach to linear and nonlinear control. In H.L. Trentelman and J.C. Willems, editors, *Perspective in the Theory and its Applications*, volume 13 of *Essays on Control*, pages 223–267. Birkhäuser, Boston, 1993.

- [11] M. Fliess, J. Lévine, P. Martin, and P. Rouchon. Flatness and defect of non-linear systems: Introductory theory and examples. *International Journal of Control*, 61:1327–1361, 1995.
- [12] R. Gardner and W. Shadwick. The GS algorithm for exact linearization to Brunovsky normal form. *IEEE Transactions on automatic control*, 37(2):224–230, 1992.
- [13] C. Guyon and G. Jacob. Systèmes nilpotents et commande polynomiale exacte. Publication du LIFL IT-278, Université Lille I, 1995.
- [14] P. Hall. A contribution to theory of groups of prime-power order. *Proc. London of Math. Soc.*, 36:29–95, 1933.
- [15] F. Hausdorff. Die symbolische Exponentialformel in der Gruppentheorie. *Leipziger Berichte*, 58:19–48, 1906.
- [16] H. Hermès. Distributions and the Lie algebras their bases can generate. *A.M.S.*, 106(2):555–565, 1989.
- [17] H. Hermès. *Nilpotent and High-Order Approximations of Vector Fields Systems*, volume 33, pages 238–264. S.I.A.M., 1991.
- [18] H. Hermès, A. Lundell, and D. Sullivan. Nilpotent bases for distributions and control systems. *Journal of Differential Equations*, (55):385–400, 1984. (Ac. Press Inc.).
- [19] T. Huillet, A. Monin, and G. Salut. Représentations exponentielles des systèmes analytiques généraux. *RAIRO APII*, 24:37–59, 1990.
- [20] A. Isidori. *Nonlinear Control Systems: an Introduction*. Lecture Notes in Control and Information Sciences. Springer-Verlag, 1988.
- [21] G. Jacob. Lyndon discretization and exact motion planning. In *Proc. of ECC'91*, pages 1507–1512, 1991.
- [22] G. Jacob. Algebraic computation tools and the dynamic motion planning problem. In *Proc. of ECC'93*, pages 1919–1924, Groningen, The Netherlands, June 1993.
- [23] I. Kaplansky. *An Introduction to Differential Algebra*. Hermann.
- [24] M. Kawski. Nilpotent Lie algebras of vectorfields. *J. reine angew. Math.*, 388:1–17, 1988.
- [25] M. Kawski. Combinatorics of realization of nilpotent control systems. In *Proc. IFAC-Symposium NOLCOS'92*, pages 639–644, Bordeaux, France, 1992.

- [26] A.A. Klyachko. Lie elements in the tensor algebra. *Sibirskii Matematicheskii Zhurnal*, 15:1296–1304, 1974. (translation).
- [27] D. Krob, G. Duchamp, B. Leclerc, and J.-Y. . Thibon. *Déformations de projecteurs de Lie*, volume 319 of *1*, pages 909–914. C.R.A.S., 1994.
- [28] D. Krob, B. Leclerc, J.-C. Novelli, and J.-Y. . Thibon. Campbell-Baker-Hausdorff type computations using noncommutative symmetric functions. Publication du LITP, Université Paris VI, 1995.
- [29] G. Lafferrière and H.J. Sussmann. Motion planning for controllable systems without drift: a preliminary report. Center for system and control, Rutgers University, June 1990.
- [30] J.-P. Laumond. Controllability of a multibody mobile robot. In *I.C.A.R.*, Groningen, The Netherlands, June 1991.
- [31] J.-P. Laumond, P. Jacobs, M. Taïx, and R.W. Murray. A motion planner for nonholonomic mobile robots. In *IEEE Transactions on Robotics and Automation*, 1993.
- [32] R.C. Lyndon. On Burnside problem i. *Trans. Amer. Math. Soc.*, 77:202–215, 1979.
- [33] G. Melançon and C. Reutenauer. Lyndon words, free algebras and shuffles. Publication du LITP Paris et UQAM Montreal 87–63, Université Paris VII et Université du Québec, 1987.
- [34] V. Hoang Ngoc Minh. *Contribution au développement d'outils informatiques pour résoudre des problèmes d'automatique non linéaire*. Thèse de doctorat, Université Lille I, juin 1990.
- [35] R. Murray. Nilpotent bases for a class of non-integrable distributions with applications to trajectory generation for nonholonomic systems. Memorandum no. ucb erl m91/45, California Institute of Technology, 1992.
- [36] R. Murray and S. Sastry. Nonholonomic motion planning steering using sinusoids. Memorandum no. ucb erl m91/45, University of California at Berkeley, Electronics Research Laboratory, 1991.
- [37] P. J. Olver. *Applications of Lie Groups to Differential Equations*. Graduate Texts in Mathematics (second edition). Springer-Verlag, 1993.
- [38] N. Oussous and M. Petitot. The motion planning with Lie algebra (combinatoric tools). In *Proc. of ECC'93*, pages 2103–2106, Groningen, The Netherlands, June 1993.



- [39] N.E. Oussous and M. Petitot. Computer algebra for the motion planning problem. In *MTNS 93*, Regensburg, Germany, august 1993. Mathematical Theory of Networks and Systems.
- [40] R. Ree. Lie elements and an algebra associated with shuffles. *Annals of Mathematics*, 68:210–220, 1958.
- [41] C. Reutenauer. *Free Lie Algebras*. Math. Soc. Monog. 7 (new serie) Clarendon Press. Oxford Sciences Publications, 1993.
- [42] S. Sastry and Z. Li. Robot motion planning with nonholonomic constraints. In *28th CDC*, Tampa, Florida, december 1989.
- [43] M.-P. Schützenberger. Sur une propriété combinatoire des algèbres de lie libres pouvant être utilisée dans un problème de mathématiques appliquées. Séminaire p. dubreil, Faculté des Sciences, Paris, 1958.
- [44] A.A. Širšov . Bases of free Lie algebras. *Algebra i Logika*, 1:14–19, 1962.
- [45] P. Souères and J.-P. Laumond. Shortest paths synthesis for a car-like robot. In *European Control Conference, Proc ECC 93*, pages 570–577, Groningen, The Netherlands, 1993.
- [46] H.J. Sussmann. A product expansion for the Chen series. In *In Theory and Applications of Non-Linear Control Systems*, pages 323–335. C. Byrnes and A. Lindquists editors, North-Holland, 1986.
- [47] H.J. Sussmann. Two new methods for motion planning for controllable systems without drift. In *Proc. of ECC'91*, Grenoble, France, July 2–5 1991.
- [48] D. Tilbury, R. Murray, and S. Sastry. Trajectory generation for the n-trailer problem using goursat normal form. Memorandum no. ucb erl m93/12, University of California, Berkeley, 1993.
- [49] G. Viennot. *Algèbres de Lie libres et monoïdes libres*, volume 691 of *Lecture Notes In Mathematics*. Springer-Verlag, 1978.
- [50] F. Wever. über Invarianten in Liesche Ringe. *Mathematische Annalen*, 120:563–580, 1949.
- [51] E. Witt. über freie Liesche Ringe und ihre Unterringe. *Mathematische Zeitschrift*, 64:195–216, 1956.