69189

50376

Numéro d'ordre : 558

ANNEE : 1990

1990
138

USTL FLANDRES ARTOIS — LIFL U.A. 369 du C.N.R.S.

LABORATOIRE D'INFORMATIQUE FONDAMENTALE DE LILLE

50376
1990
138

THESE

présentée à

L'UNIVERSITE DES SCIENCES ET TECHNIQUES
DE LILLE FLANDRES ARTOIS

pour obtenir le titre de

DOCTEUR en INFORMATIQUE

BIBLIOTHEQUE INTERUNIVERSITAIRE
SECTION DE SCIENCES
LILLE

CONTRIBUTION AU DEVELOPPEMENT D'OUTILS INFORMATIQUES POUR RESOUDRE DES PROBLEMES D'AUTOMATIQUE NON LINEAIRE

par

HOANG NGOC MINH

Thèse soutenue le 22 Juin 1990, devant la commission d'examen

Membres du Jury:

Président	: Max DAUCHET
Rapporteurs	: Jean BERSTEL
	: Françoise LAMNABHI-LAGARRIGUE
Membres	: Gérard JACOB
	: Claude MOOG
	: Daniel RAKOTOPARA

UNIVERSITE DES SCIENCES
ET TECHNIQUES DE LILLE
FLANDRES ARTOIS

DOYENS HONORAIRES DE L'ANCIENNE FACULTE DES SCIENCES

M.H. LEFEBVRE, M. PARREAU.

PROFESSEURS HONORAIRES DES ANCIENNES FACULTES DE DROIT ET SCIENCES ECONOMIQUES, DES SCIENCES ET DES LETTRES

MM. ARNOULT, BONTE, BROCHARD, CHAPPELON, CHAUDRON, CORDONNIER, DECUYPER, DEHEUVELS, DEHORS, DION, FAUVEL, FLEURY, GERMAIN, GLACET, GONTIER, KOURGANOFF, LAMOTTE, LASSERRE, LELONG, LHOMME, LIEBAERT, MARTINOT-LAGARDE, MAZET, MICHEL, PEREZ, ROIG, ROSEAU, ROUELLE, SCHILTZ, SAVARD, ZAMANSKI, Mes BEAUJEU, LELONG.

PROFESSEUR EMERITE

M. A. LEBRUN

ANCIENS PRESIDENTS DE L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE

MM. M. PAREAU, J. LOMBARD, M. MIGEON, J. CORTOIS.

PRESIDENT DE L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE FLANDRES ARTOIS

M. A. DUBRULLE.

PROFESSEURS - CLASSE EXCEPTIONNELLE

M. CONSTANT Eugène	Electronique
M. FOURET René	Physique du solide
M. GABILLARD Robert	Electronique
M. MONTREUIL Jean	Biochimie
M. PARREAU Michel	Analyse
M. TRIDOT Gabriel	Chimie Appliquée

PROFESSEURS - 1ère CLASSE

M. BACCHUS Pierre	Astronomie
M. BIAYS Pierre	Géographie
M. BILLARD Jean	Physique du Solide
M. BOILLY Bénoni	Biologie
M. BONNELLE Jean-Pierre	Chimie-Physique
M. BOSCQ Denis	Probabilitéss
M. BOUGHON Pierre	Algèbre
M. BOURIQUET Robert	Biologie Végétale
M. BREZINSKI Claude	Analyse Numérique

M. BRIDOUX Michel	Chimie-Physique
M. CELET Paul	Géologie Générale
M. CHAMLEY Hervé	Géotechnique
M. COEURE Gérard	Analyse
M. CORDONNIER Vincent	Informatique
M. DAUCHET Max	Informatique
M. DEBOURSE Jean-Pierre	Gestion des Entreprises
M. DHAINAUT André	Biologie Animale
M. DOUKHAN Jean-Claude	Physique du Solide
M. DYMENT Arthur	Mécanique
M. ESCAIG Bertrand	Physique du Solide
M. FAURE Robert	Mécanique
M. FOCT Jacques	Métallurgie
M. FRONTIER Serge	Ecologie Numérique
M. GRANELLE Jean-Jacques	Sciences Economiques
M. GRUSON Laurent	Algèbre
M. GUILLAUME Jean	Microbiologie
M. HECTOR Joseph	Géométrie
M. LABLACHE-COMBIER Alain	Chimie Organique
M. LACOSTE Louis	Biologie Végétale
M. LAVEINE Jean-Pierre	Paléontologie
M. LEHMANN Daniel	Géométrie
Mme LENOBLE Jacqueline	Physique Atomique et Moléculaire
M. LEROY Jean-Marie	Spectrochimie
M. LHOMME Jean	Chimie Organique Biologique
M. LOMBARD Jacques	Sociologie
M. LOUCHEUX Claude	Chimie Physique
M. LUCQUIN Michel	Chimie Physique
M. MACKE Bruno	Physique Moléculaire et Rayonnements Atmosph.
M. MIGEON Michel	E.U.D.I.L.
M. PAQUET Jacques	Géologie Générale
M. PETIT Francis	Chimie Organique
M. POUZET Pierre	Modélisation - calcul Scientifique
M. PROUVOST Jean	Minéralogie
M. RACZY Ladislas	Electronique
M. SALMER Georges	Electronique
M. SCHAMPS Joel	Spectroscopie Moléculaire
M. SEGUIER Guy	Electrotechnique
M. SIMON Michel	Sociologie
Melle SPIK Geneviève	Biochimie
M. STANKIEWICZ François	Sciences Economiques
M. TILLIEU Jacques	Physique Théorique
M. TOULOTTE Jean-Marc	Automatique
M. VIDAL Pierre	Automatique
M. ZEYTOUNIAN Radyadour	Mécanique

PROFESSEURS - 2ème CLASSE

M. ALLAMANDO Etienne	Composants Electroniques
M. ANDRIES Jean-Claude	Biologie des organismes
M. ANTOINE Philippe	Analyse
M. BART André	Biologie animale
M. BASSERY Louis	Génie des Procédés et Réactions Chimiques

Mme BATTIAU Yvonne	Géographie
M. BEGUIN Paul	Mécanique
M. BELLET Jean	Physique Atomique et Moléculaire
M. BERTRAND Hugues	Sciences Economiques et Sociales
M. BERZIN Robert	Analyse
M.BKOUCHE Rudolphe	Algèbre
M. BODARD Marcel	Biologie Végétale
M. BOIS Pierre	Mécanique
M. BOISSIER Daniel	Génie Civil
M. BOIVIN Jean-Claude	Spectroscopie
M. BOUQUELET Stéphane	Biologie Appliquée aux enzymes
M. BOUQUIN Henri	Gestion
M. BRASSELET Jean-Paul	Géométrie et Topologie
M. BRUYELLE Pierre	Géographie
M. CAPURON Alfred	Biologie Animale
M. CATTEAU Jean-Pierre	Chimie Organique
M. CAYATTE Jean-Louis	Sciences Economiques
M. CHAPOTON Alain	Electronique
M. CHARET Pierre	Biochimie Structurale
M. CHIVE Maurice	Composants Electroniques Optiques
M. COMYN Gérard	Informatique Théorique
M. COQUERY Jean-Marie	Psychophysiologie
M. CORIAT Benjamin	Sciences Economiques et Sociales
Mme CORSIN Paule	Paléontologie
M. CORTOIS Jean	Physique Nucléaire et Corpusculaire
M. COUTURIER Daniel	Chimie Organique
M. CRAMPON Norbert	Tectolique Géodynamique
M. CROSNIER Yves	Electronique
M. CURGY Jean-Jacques	Biologie
Melle DACHARRY Monique	Géographie
M. DEBRABANT Pierre	Géologie Appliquée
M. DEGAUQUE Pierre	Electronique
M. DEJAEGER Roger	Electrochimie et Cinétique
M. DELAHAYE Jean-Paul	Informatique
M. DELORME Pierre	Physioliogie Animale
M. DELORME Robert	Sciences Economiques
M. DEMUNTER Paul	Sociologie
M. DENEL Jacques	Informatique
M. DE PARIS Jean Claude	Analyse
M. DEPREZ Gilbert	Physique du Solide - Cristallographie
M. DERIEUX Jean-Claude	Microbiologie
Melle DESSAUX Odile	Spectroscopie de la réactivité Chimique
M. DEVRAINNE Pierre	Chimie Minérale
Mme DHAINAUT Nicole	Biologie Animale
M. DHAMELINCOURT Paul	Chimie Physique
M. DORMARD Serge	Sciences Economiques
M. DUBOIS Henri	Spectroscopie Hertzienne
M. DUBRULLE Alain	Spectroscopie Hertzienne
M. DUBUS Jean-Paul	Spectrométrie des Solides
M. DUPONT Christophe	Vie de la firme (I.A.E.)
Mme EVRARD Micheline	Génie des procédés et réactions chimiques
M. FAKIR Sabah	Algèbre
M. FAUQUAMBERGUE Renaud	Composants électroniques

M. FONTAINE Hubert	Dynamique des cristaux
M. FOUQUART Yves	Optique atmosphérique
M. FOURNET Bernard	Biochimie Sturcturale
M. GAMBLIN André	Géographie urbaine, industrielle et démog.
M. GLORIEUX Pierre	Physique moléculaire et rayonnements Atmos.
M. GOBLOT Rémi	Algèbre
M. GOSSELIN Gabriel	Sociologie
M. GOUDMAND Pierre	Chimie Physique
M. GOURIEROUX Christian	Probabilités et Statistiques
M. GREGORY Pierre	I.A.E.
M. GREMY Jean-Paul	Sociologie
M. GREVET Patrice	Sciences Economiques
M. GRIMBLOT Jean	Chimie Organique
M. GUILBAULT Pierre	Physiologie animale
M. HENRY Jean-Pierre	Génie Mécanique
M. HERMAN Maurice	Physique spatiale
M. HOUDART René	Physique atomique
M. JACOB Gérard	Informatique
M. JACOB Pierre	Probabilités et Statistiques
M. Jean Raymond	Biologie des populations végétales
M. JOFFRE Patrick	Vie de la firme (I.A.E.)
M. JOURNEL Gérard	Spectroscopie hertzienne
M. KREMBEL Jean	Biochimie
M. LANGRAND Claude	Probabilités et statistiques
M. LATTEUX Michel	Informatique
Mme LECLERCQ Ginette	Catalyse
M. LEFEBVRE Jacques	Physique
M. LEFEBVRE Christian	Pétrologie
Melle LEGRAND Denise	Algèbre
Melle LEGRAND Solange	Algèbre
M. LEGRAND Pierre	Chimie
Mme LEHMANN Josiane	Analyse
M. LEMAIRE Jean	Spectroscopie hertzienne
M. LE MAROIS Henri	Vie de la firme (I.A.E.)
M. LEROY Yves	Composants électroniques
M. LESENNE Jacques	Systèmes électroniques
M. LHENAFF René	Géographie
M. LOCQUENEUX Robert	Physique théorique
M. LOSFELD Joseph	Informatique
M. LOUAGE Francis	Electronique
M. MAHIEU Jean-Marie	Optique-Physique atomique
M. MAIZIERES Christian	Automatique
M. MAURISSON Patrick	Sciences Economiques et Sociales
M. MESMACQUE Gérard	Génie Mécanique
M. MESSELYN Jean	Physique atomique et moléculaire
M. MONTEL Marc	Physique du solide
M. MORCELLET Michel	Chimie Organique
M. MORTREUX André	Chimie Organique
Mme MOUNIER Yvonne	Physiologie des structures contractiles
Mme MOUYART-TASSIN Annie Françoise	Informatique
M. NICOLE Jacques	Spectrochimie
M. NOTELET Françis	Systèmes électroniques
M. PARSY Fernand	Mécanique

M. PECQUE Marcel	Chimie organique
M. PERROT Pierre	Chimie appliquée
M. STEEN Jean-Pierre	Informatique

REMERCIEMENTS

Je suis très heureux de remercier Monsieur le Professeur Max Dauchet de l'Université Lille I qui me fait l'honneur de présider ce jury. Je tiens à lui exprimer toute mon estime pour ses efforts afin que les chercheurs du L.I.F.L., dont il est directeur, travaillent dans de bonnes conditions.

C'est pour moi une très grande joie de remercier Monsieur le Professeur Gérard Jacob de l'Université Lille I. Il est à l'origine de ce travail et m'a constamment apporté amitié, encouragements, conseils et aide efficace. Qu'il trouve ici l'expression de la reconnaissance d'un élève envers son maître.

Je suis très honoré de la présence de Monsieur le Professeur Jean Berstel de l'Université Paris VI dans le jury en tant que rapporteur. Ses critiques constructives et ses conseils judicieux sur mes rapports scientifiques ont permis à ce mémoire de prendre forme.

Que Madame Françoise Lamnabhi-Lagarrigue, Chargée de recherche au C.N.R.S. au L.S.S. - E.S.E. à Gif sur Yvette, trouve ici le témoignage de ma très vive reconnaissance pour les multiples et judicieuses remarques qu'elle a apportées à ce travail et pour son acceptation d'en être rapporteur.

Je suis reconnaissant à Monsieur Claude Moog, Chargé de recherche au C.N.R.S. au L.A.N. - E.N.S.M. à Nantes, et à Monsieur Daniel Rakotopara, Responsable scientifique à Shell-Recherche à Grande Couronne, d'avoir bien voulu se pencher sur cette thèse et participer à ce jury.

Ce travail a été réalisé au sein de l'équipe Séries Non commutatives et Calcul Formel. Je veux dire à tous ses membres combien j'ai apprécié l'ambiance chaleureuse, les discussions fort fructueuses que nous avons eues pour nous mettre sur le "bon rail".

J'adresse mes remerciements à Monsieur Henri Glanc qui a mis sa grande compétence au service de l'impression de cette thèse.

Il m'est arrivé plusieurs fois de déranger les ingénieurs et les utilisateurs du réseau ETHERNET du L.I.F.L. en saturant, indépendamment de ma volonté, le serveur avec MACSYMA. Je tiens à les remercier de leur patience.

TABLE DES MATIERES

CHAPITRE 1

INTRODUCTION GENERALE

Le but de ce mémoire est de développer et étudier certains outils informatiques (séries formelles génératrices) et de les mettre en oeuvre pour obtenir puis implanter à l'aide du calcul formel (MACSYMA) des résultats fins en automatique non linéaire (calcul symbolique, équations différentielles non linéaires en régime forcé, découplage).

CADRE SCIENTIFIQUE

Pour mieux tracer le cadre scientifique dans lequel est effectué le présent travail, nous présentons d'abord les grandes lignes des techniques et résultats sur les séries formelles et la théorie des langages, le calcul formel et les systèmes dynamiques.

Séries formelles non commutatives

Les séries formelles en variables non commutatives ont été introduites pour la première fois par M.P. Schützenberger ([S2]) pour étudier les problèmes liés à l'informatique théorique, tels que la théorie des langages et la théorie des automates. Il généralise le théorème de S.C. Kleene ([K2]), qui établit l'équivalence des langages reconnaissables et rationnels, en démontrant l'équivalence de la reconnaissabilité et de la rationnalité des séries formelles ([S3]). Il montre ainsi le rôle fondamental joué, pour l'étude des séries formelles, par les repésentations matricielles du monoïde libre, notamment les décompositions en représentations irréductibles, obtenant ainsi des résultats fins sur la croissance des coefficients ([S4]).

Après M.P. Schützenberger, il faut citer les travaux de M. Fliess ([F3]), G. Jacob ([J1]) et C. Reutenauer ([R2]). Ceux-ci ont développé un ensemble d'outils fondamentaux, en lien avec la théorie des automates, et avec l'étude combinatoire du monoïde libre. En effet, M. Fliess, grâce aux matrices de Hankel, caractérise les séries rationnelles (une série est rationnelle si et seulement si le rang de sa matrice de Hankel est fini) et les séries à coefficients positifs. Avec les représentations matricielles, G. Jacob généralise la notion de boucle dans les automates finis (lemme de l'étoile), ce qui permet de résoudre des problèmes de décidabilité de la finitude de l'ensemble des coefficients des séries rationnelles. Tandis que C. Reutenauer caractérise les séries rationnelles par leur algèbre syntaxique (une série est rationnelle si et seulement si son algèbre syntaxique est de dimension finie) et il définit la notion de variété de séries formelles non commutatives au sens de S. Eilenberg ([E]). Ces travaux sont largement repris dans les livres de J. Berstel et C. Reutenauer ([BR2]), de W. Kuich et A. Salomaa ([KS]) et de A. Salomaa et M. Soittola ([SS]).

Comme l'ont montré Chomsky et M.P. Schützenberger ([CS]) les langages algébriques sont les supports des séries algébriques solutions d'un système d'équations algébriques propres à coefficients entiers positifs ou nuls. M. Nivat ([N2]), puis M. Fliess ([F3]), a montré que l'étude des séries algébriques dépend de façon fondamentale de l'étude des transductions rationnelles des séries formelles. En introduisant la notion de transduction régulée (pour établir la réciproque du théorème de Shamir) G. Jacob montre que l'image par transduction rationnelle régulée (resp. algébrique régulée) d'une série rationnelle (resp. algébrique) est une série rationnelle (resp. algébrique), ouvrant ainsi la voie à l'étude des "familles agréables de séries formelles". W. Kuich ([K7]) a approfondi cette méthode dans une approche par des automates à pile "cycle-free" pour étudier les séries algébriques.

Ainsi, les séries formelles en variables non commutatives bénéficient des acquis de la théorie des langages et de la théorie des automates. En retour, les techniques qu'elles développent et les résultats qu'elles obtiennent ont permis de nouvelles visions dans ces domaines.

Ces développements font de l'algèbre des séries formelles un outil privilégié pour l'étude syntaxique des algèbres d'opérateurs. En outre, l'algèbre des séries formelles se revèle être un outil particulièrement bien adapté pour l'implantation de calculs effectifs dans les systèmes de calcul formel modernes. Elle conduit à des méthodes élégantes d'implantation (M.P. Delest [D1], C. Hespel & G. Jacob [HJ2], G. Jacob & N.E. Oussous [JO2], P.V. Koseleff [K5]) qui utilisent des représentations non numériques et des manipulations algébriques de base. C'est d'ailleurs le développement de tels outils et de telles implantations qui constitue l'axe principal de recherche de l'équipe $\mathcal{S.N.C.F.}$ du $\mathcal{L.I.F.L.}$

Le calcul formel

Le calcul formel est apparu dans les années 60, en même temps que le langage LISP (le premier langage de l'Intelligence Artificielle) qui se propose comme un langage évolué pouvant être utilisé comme un langage machine. Ainsi les machines LISP construites consistent à manipuler des structures plus complexes que les variables numériques : les "expressions". A la même époque D. Knuth publie des livres traitant des algorithmes semi-numériques qui sont efficaces et rapides ([K3], [K4]). Ces possibilités de manipuler efficacement et rapidement des objets non numériques ouvrent la voie à un traitement plus systématique que le "calcul littéral" cher aux physiciens. En effet, les lettres d'un alphabet de codage désignent non plus essentiellement des nombres mais aussi des objets du type polynôme, fonction, expression logique. On est amené également à utiliser des lettres non commutatives voire partiellement commutatives pour symboliser les opérateurs qui ne commutent pas forcément entre eux notamment les matrices, les dérivations, les intégrations, les intructions d'un programme.

Une étude systématique sur les structures libres (magmoïdes, monoïdes, groupes, algèbres de Lie, [V1]) associées à un alphabet de codage conduit à une programmation basée sur l'analyse syntaxique des expressions, style de programmation parfaitement adapté aux systèmes de calcul formel.

Ainsi grâce aux techniques du calcul exact (sur les entiers et rationnels) et au développement du calcul sur les expressions algébriques (J.D. Lipson [L3]), le calcul formel permet d'assigner aux ordinateurs, dans le domaine du calcul scientifique, d'autres tâches que celle du calcul numérique, notamment intégration et dérivation formelles, intégration des équations différentielles linéaires, calcul matriciel (A.G. Akritas [A2], Buchberger et al. [BCL], Davenport, Y. Siret et E. Tournier [DST]). De nombreux systèmes de calcul formel sont apparus (REDUCE, MACSYMA, SCRATCHPAD, MATHEMATICA, MAPLE ...) qui peuvent être considérés comme des outils permettant de traiter des problèmes non triviaux, de tester et valider des conjectures. De nombreux algorithmes ont été implantés sur ces systèmes de calcul formel notamment sur les polynômes et séries formelles en variables commutatives à coefficients entiers ou rationnels (factorisation, division exacte, simplification, bases standard ...). Cependant, bien qu'actuellement peu développée, l'implantation d'algorithmes concernant les polynômes et séries formelles non commutatives peut se faire très naturellement dans ces systèmes de calcul formel. Une telle implantation peut contribuer d'une part, à la combinatoire énumérative qui utilise les séries formelles pour coder des graphes ou des arbres (R. Cori [C3]), pour dénombrer différentes structures combinatoires (M.P. Delest [D1], Ph. Flajolet [F1], D. Foata [F8]) et pour étudier la complexité de certains algorithmes (voir le logiciel $\Lambda\Upsilon\Omega$ de Ph. Flajolet, B. Salvy & P. Zimmermann [FSZ]), et d'autre part aux systèmes dynamiques non linéaires par l'intermédiaire des bases de l'algèbre de Lie libre (base de Hall, base Poincarré-Birkhoff-Witt, base Chen-Fox-Lyndon, [JO2], [K5]) en tant que sous-algèbre de l'algèbre des séries formelles non commutatives.

Les systèmes dynamiques

Les systèmes dynamiques ont permis de modéliser de nombreux problèmes physiques, économiques ou biomédicaux. Ils sont définis par une relation causale des entrées $\{a^z\}_{z\in Z}$ et une sortie y liées entre elles par un système d'équations différentielles faisant intervenir un vecteur d'état q évoluant avec le temps. Ces systèmes sont de la forme :

$$(NL) \quad \begin{cases} \dot{q}(t) = \sum_{z\in Z} a^z(t)A_z(q) \\ y(t) = h(q(t)), \end{cases}$$

où :

- Z est un alphabet fini de lettres $Z = \{z_0, z_1, \ldots, z_m\}$,
- q est un élément d'une variété analytique réelle Q de dimension finie N,

- pour toute lettre $z \in Z$, A_z est un champ de vecteurs analytique sur Q, c'est-à-dire un opérateur différentiel linéaire de la forme :

$$A_z = \sum_{k=1}^{N} A_z^k(q)\frac{\partial}{\partial q_k},$$

où les $\{A_z^k\}_{k\in[1..N]}$ sont des applications analytiques de la variété analytique réelle Q dans $\mathbb{R}$,

- pour toute lettre $z \in Z$, a^z est une application réelle continue sur $\mathbb{R}_+$, en particulier, a^{z_0} est l'application constante et égale à 1,

- l'observation h est une application analytique de la variété analytique réelle Q dans $\mathbb{R}^p$.

Pour une introduction aux concepts des systèmes dynamiques on poura consulter par exemple [B1], [B4], [K1], [L5].

Parmi les systèmes dynamiques, on a beaucoup étudié (en utilisant l'algèbre matricielle) la classe des systèmes linéaires c'est-à-dire du type (L) :

$$(L) \quad \begin{cases} \dot{x}(t) = Fx(t) + Gu(t) \\ y(t) = Hx(t), \end{cases}$$

où :

- le vecteur x appartient à un espace d'état réel de dimension n,

- le vecteur u est tel que chaque composante est une application continue par morceaux de $\mathbb{R}_+$ dans $\mathbb{R}$,

- F et G sont des matrices de dimensions respectives $n \times n$ et $n \times m$ et à coefficients dans $\mathbb{R}$,

- l'observation H est une matrice $p \times n$ à coefficients dans $\mathbb{R}$.

En parallèle avec les méthodes basées sur l'algèbre matricielle, les systèmes linéaires ont été systématiquement étudiés par des techniques géométriques données par W.M. Wonham et al. ([W], [MW] ...). Notons que dans le cadre linéaire, les champs de vecteurs sont calculés par :

$$A_{z_0} = \sum_{i=1}^{n}\left(\sum_{k=1}^{n} F_i^k x_k\right)\frac{\partial}{\partial x_i},$$

où pour tous entiers $i, k \in [1..n], F_i^k$ est le coefficient d'indice i, k de la matrice F,

$$\forall j \in [1..m], \quad A_{z_j} = \sum_{k=1}^{n} G_j^k \frac{\partial}{\partial x_k},$$

où pour tous entiers $j \in [1..m]$ et $k \in [1..n]$, G_j^k est le coefficient d'indice k du vecteur G_j.

Quand aux systèmes non linéaires, le langage adéquat passe par la géométrie différentielle dont les initiateurs sont R. Hermann ([H1]) et C. Lobry ([L4]). Depuis de nombreux auteurs (J.P. Gauthier [G1], A. Isidori [I], B. Jakubczyk [J3], A.J. Krener [K6], H.J. Sussmann [S5]) ont développé et approfondi cette méthode. Pour une lecture synthétique de ces travaux, on peut consulter le livre de J.P. Gauthier ([G1]) ou de A. Isidori ([I]).

On sait que l'algèbre des séries commutatives joue un rôle déterminant en géométrie (voir par ex. M.P. Malliavin [M1]). De même, l'algèbre des séries formelles non commutatives trouve aussi sa place naturelle en géométrie différentielle. Elle permet une étude plus "syntaxique" de la structure des systèmes. En effet, à l'observation h d'un système dynamique du type (NL) M. Fliess ([F5]) associe une série génératrice, qui est une série formelle en variables non commutatives, sur un alphabet fini Z, de la forme :

$$\sigma_{q(0)}h = \sum_{w \in Z^*} (A_w \circ h)_{|q(0)} w,$$

où $|q(0)$ indique l'évaluation en l'état initial $q(0)$ du système, et pour tout mot w du monoïde libre Z^*, A_w représente l'opérateur différentiel suivant :

$$A_w = \begin{cases} \text{l'identité} & \text{si} \quad w = \epsilon \\ A_z \circ A_v & \text{si} \quad w = zv. \end{cases}$$

La formule fondamentale de M. Fliess permet d'obtenir la sortie $y(t)$ relative aux entrées $\{a^z\}_{z \in Z}$ pendant l'intervalle de temps $[0, t]$, $t \geq 0$, de la manière suivante ([F5]):

$$y(t) = \sum_{w \in Z^*} (A_w \circ h)_{|q(0)} \int_0^t \delta_a w,$$

où $\int_0^t \delta_a w$ est l'intégrale itérée associée au mot w définie par récurrence sur la longueur de w comme suit (nous utilisons une notation symétrique de celle de M. Fliess, ces deux notations correspondent aussi à celle introduite par K.T. Chen [C1]) :

$$\int_0^t \delta_a w = \begin{cases} 1 & \text{si} \quad w = \epsilon, \\ \int_0^t \left(\int_0^\tau \delta_a v \right) a^z(\tau) d\tau & \text{si} \quad w = vz. \end{cases}$$

Cette formule n'est autre que la "formule de Peano-Baker" ([F5]) pour les systèmes bilinéaires (systèmes dont la série génératrice est rationnelle).

Ainsi l'introduction des séries formelles en variables non commutatives en théorie du contrôle depuis 1976 a renouvelé la résolution des problèmes posés en automatique des systèmes dynamiques non linéaires et a déjà permis à M. Fliess et à son école d'obtenir des résultats algébriques fondamentaux, parmi lesquels nous citons :

. Calcul symbolique : le codage de M. Fliess conduit naturellement M. Lamnabhi ([L2]) et F. Lamnabhi-Lagarrigue ([L1]) à développer un calcul symbolique non commutatif généralisant au domaine non linéaire le calcul de Heaviside (M. Fliess, M. Lamnabhi & F. Lamnabhi-Lagarrigue [FLL]). Ainsi, les notions de fonctions de transfert (séries génératrices à une variable) et de réponse impulsionelle, codant les signaux produits par les systèmes linéaires (ou multilinéaires), se généralisent aux séries génératrices à $m+1$ variables et aux séries de Volterra, codant les signaux produits par les systèmes (analytiques) non linéaires. Dans cette approche, ces auteurs donnent une justification théorique aux méthodes heuristiques en Physique qui utilisent le calcul de Heaviside dans le cadre non linéaire. Ils donnent aussi une nouvelle méthode ([FLL]) pour calculer itérativement une solution approchée des équations différentielles non linéaires en régime forcé. Pour cette méthode, les auteurs fournissent les approximants du type de Volterra qui sont les combinaisons linéaires de fractions rationnelles non commutatives de la forme

$$(c_0 z_0)^{*p_0} z_{i_1} (c_1 z_0)^{*p_1} z_{i_2} (c_2 z_0)^{*p_2} \ldots z_{i_{k-1}} (c_{k-1} z_0)^{*p_{k-1}} z_{i_k} (c_k z_0)^{*p_k},$$

où $p_0, \ldots, p_k$ sont des entiers positifs, $c_0, \ldots, c_k$ sont des nombres complexes, $z_{i_0}, \ldots, z_{i_k}$ sont des lettres d'un alphabet fini Z. Pour calculer le signal associé à ces approximants, ils ont utilisé la tranformation de Laplace-Borel inverse. Très récemment, G.Viennot et P. Leroux ont donné une interprétation purement combinatoire en utilisant des "arbres croissants", des "chemins de Motzkin" et des "histoires" ([VL]).

. Découplage des systèmes dynamiques : dans l'étude du rejet de perturbations pour les systèmes non linéaires D. Claude ([C3]) a utilisé conjointement les séries génératrices et l'algèbre des champs de vecteurs. Ainsi les algèbres engendrées par les modules, remplaçant les distributions invariantes de l'approche géométrique, conduisent à une classe de lois de bouclage plus large que celle obtenue par les distributions invariantes. L'existence de la plus grande algèbre de découplage a été mise en évidence. Il montre aussi que l'obtention d'une caractérisation algébrique du découplage nécessite l'étude des systèmes non initialisés, et conduit à coder les systèmes dynamiques par leurs séries génératrices non commutatives à coefficients dans $C^{\omega}(Q)$, anneau des fonctions analytiques sur une variété analytique réelle Q. En introduisant le concept d'immersion ([CFI]), qui préserve le comportement d'entrée-sortie, et un choix judicieux de lois de bouclage régulier assurant le découplage, D. Claude donne des conditions nécessaires et suffisantes pour qu'un système non linéaire puisse, par bouclage, être immergé dans un système linéaire dont la dimension dépend des nombres caractéristiques.

. Approximation des systèmes dynamiques non linéaires : partant du théorème de M. Fliess :

Toute fonctionnelle causale analytique peut être uniformément approchée par les fonctionnelles régulières,

C. Hespel G. Jacob approximent un système dynamique analytique de série génératrice S par un système régulier (système dont la série génératrice G est rationnelle) de rang minimal pour un ordre donné k (i.e. les sorties de ces systèmes ont le même développement de Taylor jusqu'à l'ordre k). L'algorithme est basé sur un calcul la série G ayant les mêmes coefficients que la série S, pour tous les mots de longueur inférieure ou égale à k. Ces mots sont codés digitalement, pour rendre plus aisé le calcul des matrices de Hankel de S et de G. Ils associent alors à G la solution du problème. Ce problème non trivial a conduit les auteurs ([HJ1], [HJ2], [HJ3]) à une meilleur approximation que celle de la linéarisation tangente. Dans ce travail les auteurs ont aussi développé des outils informatiques adéquats (les "$\mathbb{R}$-automates de champs de vecteurs") pour implanter à l'aide du calcul formel (MACSYMA) un algorithme calculant l'automate tronqué qui fournit un approximant bilinéaire (non nécessairement minimal) de S.

. Réalisation locale minimale : après avoir implanté un paquetage des outils informatiques et algébriques (bases de Lie, mots de Lyndon, produit de mélange, calcul résiduel ...) pour traiter des séries formelles non commutatives en MACSYMA, G. Jacob et N.E. Oussous ont traité un problème technique de l'automatique ([JO2]) : le problème du calcul de la réalisation locale minimale des séries génératrices finies (M. Fliess [F6], C. Reutenauer [R3]). Ils ont répondu à la conjecture de G. Jacob :

Pour tout polynôme non commutatif, on peut trouver une réalisation locale minimale complètement polynômiale.

Leur démonstration consiste en une implantation en utilisant des mots de Lyndon (G. Mélançon & C. Reutenauer [MR2]) pour la construction des coordonnées locales sur laquelle ils calculent l'observation et les champs de vecteurs.

Ces travaux ont été prolongés dernièrement par les techniques d'algèbre différentielle dues essentiellement à J.F. Ritt ([R4]). Ces techniques ont été introduites par M. Fliess ([F7]) pour étudier une classe de systèmes dynamiques définie par des équations implicites différentielles algébriques. Ces travaux permettent une nouvelle compréhension de notions telles que la commandabilité, l'inversibilité, la réalisation ... (S. Diop [D2], C. Moog [M2]). On pourra aussi consulter les deux livres [P1] et [P2] de J.F. Pommaret pour une étude récente approfondie des techniques algébrico-différentielles et leur applications notamment en automatique.

PRESENTATION DU CONTOUR DU MEMOIRE

Les techniques de calcul et d'implantation développées dans ce mémoire présentent une méthode nouvelle et différente pour la programmation par rapport au logiciel déjà présenté par divers auteurs. Les différences avec les techniques de ces auteurs sont abordées à l'introduction de chaque chapitre.

Nos résultats sont déjà présentés succintement dans ([H2], [HJ4], [HJ5], [HJ6], [HJ7], [HJ8], [HJO]). Nous allons présenter notre contribution en six chapitres :

. Dans le premier chapitre, nous rappelons des résulats concernant l'algèbre des séries formelles en variables commutatives et non commutatives. Nous abordons aussi des questions de convergence de ces séries. Nous rappelons également des éléments de l'algèbre de Lie libre en tant que sous algèbre des séries formelles, et nous montrons certains résultats généraux pour étudier les champs de vecteurs sur les variétés analytiques.

. Dans le deuxième chapitre, nous développons les *bases d'un calcul symbolique* pour les systèmes dynamiques non linéaires. Nous en déduisons un théorème de correspondance entre certains produits de convolution de signaux et le produit de Cauchy des séries génératrices (codant les signaux produits par les systèmes dynamiques non linéaires). Si les séries génératrices sont considérées comme un codage symbolique du comportement entrée/sortie des systèmes dynamiques non linéaires, la transformation d'Evaluation, que nous introduisons pour ce calcul symbolique, permet en retour de déduire simplement le comportement temporel de ces systèmes à partir de leur description symbolique. Notre résultat le plus général est le calcul de l'Evaluation des combinaisons des séries de la forme :

$$G_0 z_{i_1} G_1 z_{i_2} G_2 \ldots z_{i_{k-1}} G_{k-1} z_{i_k} G_k,$$

où les séries $G_0, \ldots, G_k$ sont échangeables et $z_{i_0}, \ldots, z_{i_k}$ sont des lettres. La transformation d'Evaluation peut être considérée comme une généralisation de la transformation de Laplace inverse pour les systèmes dynamiques non linéaires. Elle est aussi une extension en plusieurs variables non commutatives du passage de la série génératrice ordinaire à la série génératrice exponentielle, bien connu en combinatoire énumérative étudié notamment par D. Foata et Ph. Flajolet.

. Dans le troisième chapitre, nous étudions une *dualité entre les séries de Chen et les séries de Fliess.* Nous donnons une condition suffisante pour qu'une série de Fliess soit "évaluable" au sens du deuxième chapitre. Nous redémontrons un résultat concernant la concaténation des séries de Chen en utilisant un lemme de translation temporelle. Nous donnons une graduation des séries de Chen basée sur l'égalité $(z_0 + z)^* = z_0^*(zz_0^*)^*$ (comme l'avait fait M. Fliess pour les séries génératrices en "réarrangeant les termes"). Cette graduation nous permet d'obtenir simplement les développements de Taylor des noyaux de Volterra.

. Dans le quatrième chapitre, nous montrons que la transformation d'Evaluation permet aussi d'étudier les *équations différentielles non linéaires* *en régime forcé* du type :

$$y^{(n)}(t) + \alpha_1 y^{(n-1)}(t) + \ldots + \alpha_n y(t) + \sum_{i=2}^{d} \beta_i y^i(t) = u(t),$$

où u est l'entrée, y est la sortie, les $\alpha_1, \ldots, \alpha_n$ et les $\beta_2, \ldots, \beta_d$ sont des constantes, et les conditions initiales sont nulles. Notre méthode consiste à transformer (via les séries génératrices et la transformation d'Evaluation) ces équations différentielles en des équations intégrales du type :

$$y(t) = f(t) + \sum_{i=2}^{d} \beta_i \int_0^t y^i(\tau) h(t-\tau) d\tau.$$

Nous proposons une méthode itérative simple et efficace pour calculer une solution approchée de ces équations intégrales. Nous donnons des exemples de calcul en utilisant MACSYMA.

. Dans le cinquième chapitre, en utilisant les séries de M. Fliess, et en utilisant des calculs dans l'algèbre de Lie libre, nous étudions le *découplage des systèmes dynamiques non linéaires* (linéaires en la commande) par bouclage analytique par retour d'état. Nous adaptons, en vue d'une implantation en MACSYMA, la technique développée par D. Claude, qui utilise les séries génératrices non commutatives à coefficients fonctions analytiques, dans le cas plus général du découplage d'un sous ensemble de sorties par rapport à un sous ensemble d'entrées. Nous retrouvons en cas particulier des résultats concernant les problèmes suivant :

- Rejet de Perturbations.
- Découplage k-bloc Diagonal.
- Découplage k-bloc Triangulaire.
- Systèmes Non Interactifs.

Cette technique se distingue des approches du type géométrie différentielle ou algèbre différentielle par le fait qu'elle s'oriente vers *le calcul effectif des lois de bouclage analytique.*

. Enfin, dans le sixième chapitre, nous indiquons nos choix de structures de données pour implanter nos algorithmes en MACSYMA, en particulier nos choix de *représentation des séries formelles* en variables non commutatives par les "arbres binaires". Nous donnons ensuite des *programmes écrits en* MACSYMA permettant de calculer les dérivations successives d'une fonction analytique de $C^\omega(Q)$, les nombres caractéristiques et la matrice de commande de chaque système. Nous terminons par une description de nos algorithmes récursifs, en particulier l'algorithme d'Evaluation utilisant les "λ−notations" de MACSYMA pour calculer le comportement temporel des systèmes dynamiques à partir des fractions rationnelles non commutatives de la forme :

$$(c_0 z_{j_0})^{*p_0} z_{i_1} (c_1 z_{j_1})^{*p_1} z_{i_2} (c_2 z_{j_2})^{*p_2} \ldots z_{i_{k-1}} (c_{k-1} z_0)^{*p_{k-1}} z_{i_k} (c_k z_{j_k})^{*p_k},$$

où $p_0, \ldots, p_k$ sont des entiers positifs, $c_0, \ldots, c_k$ sont des nombres complexes, $z_{j_0}, \ldots, z_{j_k}$ et $z_{i_0}, \ldots, z_{i_k}$ sont des lettres d'un alphabet de codage.

Les travaux et techniques que nous présentons ci après conduisent à penser que les outils de l'informatique théorique d'une part et l'utilisation adroite du calcul formel de l'autre peuvent jouer un rôle croissant en automatique non linéaire, aussi bien pour les travaux fondamentaux que pour les applications. Nous espérons que nos travaux ultérieurs pourront y contribuer.

CHAPITRE 1

CADRE GENERAL

1.0. Introduction

Dans ce chapitre, nous rappelons des résultats concernant l'algèbre des séries formelles en variables commutatives ([A1], [G2]) et non commutatives ([BR2]). Nous abordons aussi des questions de convergence de ces séries. Nous rappelons également des éléments de l'algèbre de Lie libre ([V1]) en tant que sous algèbre des séries formelles, et nous montrons certains résultats généraux pour étudier les champs de vecteurs sur les variétés analytiques ([G1], [I]).

1.1. Mots et langages sur un alphabet

Soit Z un ensemble fini non vide que l'on appelle **alphabet**. Un élément de Z est appelé **lettre**. Un **mot** w (sur Z) est une suite de lettres de Z :

$$w = z_{j_1} z_{j_2} \ldots z_{j_k}.$$

La longueur d'un mot w est notée $|w|$ (ici $|w| = k$). La suite vide, que l'on appelle aussi **mot vide**, sera noté ϵ, ($|\epsilon| = 0$). On note $|w|_z$ le nombre d'occurrences de la lettre z dans le mot w. On a clairement :

$$|w| = \sum_{z \in Z} |w|_z.$$

On note Z^* l'ensemble des mots sur Z. On note aussi Z^k l'ensemble des mots de longueur k, $k \geq 0$, on a clairement :

$$Z^* = \bigcup_{k \geq 0} Z^k.$$

On définit le produit des deux mots u et v de Z^* par la "juxtaposition" des deux suites u et v. Ainsi, si :

$$u = z_{j_1} z_{j_2} \ldots z_{j_k}, \qquad v = z_{i_1} z_{i_2} \ldots z_{i_l},$$

alors le produit de deux mots u et v est le mot w de Z^* :

$$w = z_{j_1} z_{j_2} \ldots z_{j_k} z_{i_1} z_{i_2} \ldots z_{i_l}.$$

Ce produit est associatif et non commutatif, et admet ϵ comme élément neutre. Z^* muni de ce produit n'est autre que **le monoïde libre engendré par l'alphabet** Z ([V1]).

Une partie $L \subset Z^*$ est appelée **langage** sur l'alphabet Z. Soient L_1, L_2 deux langages sur Z. Le produit, noté ".", de deux langages L_1 et L_2 est le langage $L_1.L_2$ défini (par extension de celui des mots) comme suit :

$$L_1.L_2 = \{w \in Z^*,\ w = w_1w_2,\ w_1 \in L_1,\ w_2 \in L_2\}.$$

Ce produit est associatif, et admet $\{\epsilon\}$ comme élément neutre. Lorsqu'il n'y a pas d'ambiguïté, on omettra le symbole ".".

L'ensemble $\mathcal{P}(Z^*)$ des parties de Z^*, muni de ce produit, est un monoïde. Soit $L \in \mathcal{P}(Z^*)$. On note L^k, $k \geq 1$, le langage sur Z défini par :

$$L^k = \{w = w_1w_2\dots w_k \in Z^* \mid \quad w_1, w_2, \dots, w_k \in L\}.$$

Et on pose $L^0 = \{\epsilon\}$. On définit également sur $\mathcal{P}(Z^*)$ l'opération **étoile** d'une partie L de $\mathcal{P}(Z^*)$ par $L^* = \bigcup_{k \geq 0} L^n$. Les trois opérations de produit, d'union et d'étoile sur $\mathcal{P}(Z^*)$ sont désignées sous le terme d'**opérations rationnelles**.

Soit Z_1 une partie de Z . On pose $Z_0 = Z \setminus Z_1$, d'après ([SS]) on a :

$$Z^* = \left(Z_1 \bigcup Z_0\right)^* = (Z_1^* Z_0)^* Z_1^* = \bigcup_{k \geq 0} (Z_1^* Z_0)^k Z_1^*.$$

En particulier, si $Z_1 = \{z_0\}$, alors :

$$Z^* = \left(\{z_0\} \bigcup Z_0\right)^* = (z_0^* Z_0)^* z_0^* = \bigcup_{k \geq 0} (z_0^* Z_0)^k z_0^*.$$

1.2. Séries formelles sur un alphabet fini ([BR2])

Soit A un anneau commutatif unitaire. On appelle **série formelle à coefficients dans A en les indéterminées associatives $z \in Z$ (non commutatives si** $\operatorname{card} Z \geq 2$), toute application :

$$\begin{aligned} S : Z^* &\longrightarrow A \\ w &\longmapsto < S|w >, \end{aligned}$$

et on note $A \ll Z \gg$ l'ensemble des séries formelles sur Z. Une série formelle S sera notée comme une somme infinie :

$$S = \sum_{w \in Z^*} < S|w > w.$$

Une série $S \in A \ll Z \gg$ est dite **propre** si et seulement si son terme constant, $< S|\epsilon >$, est nul.

Soit L un langage sur Z, on appelle **série caractéristique** de L, la série car(L) de $A \ll Z \gg$ définie par :

$$\forall w \in Z^*, \quad < \mathrm{car}(L)|w > = \begin{cases} 1 & \text{si} \quad w \in L \\ 0 & \text{si} \quad w \notin L \end{cases}$$

c'est-à-dire :

$$\mathrm{car}(L) = \sum_{w \in L} w.$$

Soit S une série de $A \ll Z \gg$. Le support de S est le langage sur Z défini par :

$$\mathrm{supp}(S) = \{w \in Z^* | < S|w > \neq 0\}.$$

Soit L un langage sur Z, on a :

$$\mathrm{supp}(\mathrm{car}(L)) = L.$$

Soit S une série formelle de $A \ll Z \gg$, on a :

$$\mathrm{car}(\mathrm{supp}(S)) = S \quad \Longleftrightarrow \quad \forall w \in Z^*, \ < S|w > \in \{0, 1\}.$$

Toute série formelle à coefficients dans $\{0, 1\}$ peut être identifiée à son support. Ainsi, pour tout langage L sur Z, on note encore L sa série caractéristique.

Soit S une série formelle de $A \ll Z \gg$. On définit l'**ordre** de S, que l'on note $\omega(S)$, par :

$$\omega(S) = \begin{cases} +\infty & \text{si} \quad S = 0, \\ \inf\{|w|, \quad w \in \mathrm{supp}(S)\} & \text{si} \quad S \neq 0. \end{cases}$$

Une série formelle P de support fini est appelée **polynôme. Le degré** de P est défini par :

$$\deg(P) = \begin{cases} -\infty & \text{si} \quad P = 0, \\ \max\{|w|, \quad w \in \mathrm{supp}(P)\} & \text{si} \quad P \neq 0. \end{cases}$$

On note $A < Z >$ l'ensemble des polynômes à coefficients dans A en les indéterminées associatives $z \in Z$ (non commutatives si card $Z \geq 2$). Chaque série formelle S de $A \ll Z \gg$ peut être considérée comme une forme linéaire sur $A < Z >$:

$$\forall P \in A < Z >, \quad < S|P > = \sum_{w \in Z^*} < P|w >< S|w > .$$

Ainsi $A \ll Z \gg$ est identifiée au dual de $A < Z >$.

1.3. Calcul de résiduels sur $A \ll Z \gg$

Définition 1.3.1. : *Soit S une série formelle de $A \ll Z \gg$. Soit u un mot de Z^*. Nous appelons* **résiduel à gauche** *(resp.* **droite***)* **de** S **par** u, *la série formelle $u \triangleleft S$ (resp. $S \triangleright u$) de $A \ll Z \gg$ définie par :*

$$\forall w \in Z^*, \quad < u \triangleleft S | w > = < S | wu > \quad (resp. \ < S \triangleright u | w > = < S | uw >).$$

Remarque : *Pour toutes lettres x, y de Z, pour tout mot w de Z^*, nous avons :*

$$x \triangleleft (wy) = \delta_x^y w,$$

où δ_x^y est le symbole de Kronecker usuel.

Nous pouvons vérifier aisément que pour toute série formelle S de $A \ll Z \gg$:

$$\begin{aligned} &\epsilon \triangleleft S = S \triangleright \epsilon = S, \\ \forall u \in Z^*, \quad &\mathrm{supp}(u \triangleleft S) = \{w \in Z^* | \quad wu \in \mathrm{supp}(S)\} \\ &\mathrm{supp}(S \triangleright u) = \{w \in Z^* | \quad uw \in \mathrm{supp}(S)\}, \end{aligned}$$

et :

$$\begin{aligned} \forall u, \ v \in Z^*, \quad & v \triangleleft (u \triangleleft S) = vu \triangleleft S \\ & (S \triangleright u) \triangleright v = S \triangleright uv \\ & (u \triangleleft S) \triangleright v = u \triangleleft (S \triangleright v). \end{aligned}$$

Nous avons ainsi défini deux actions de Z^* sur $A \ll Z \gg$. La première, notée par "$\triangleleft$", agit à gauche. Et la seconde, notée par "$\triangleright$", agit à droite. Chacune des deux actions est associative et commute avec l'autre.

Nous étendons ces deux actions à $A < Z >$ de la manière suivante :

Définition 1.3.2. : *Soit S une série formelle de $A \ll Z \gg$. Soit P un polynôme de $A < Z >$. Nous appelons* **résiduel à gauche** *(resp.* **droite***)* **de** S **par** P, *la série formelle $P \triangleleft S$ (resp. $S \triangleright P$) de $A \ll Z \gg$ définie par :*

$$\forall w \in Z^*, \quad < P \triangleleft S | w > = < S | wP > = \sum_{v \in Z^*} < S | wv >< P | v >$$

$$(resp. \ < S \triangleright P | w > = < S | Pw > = \sum_{v \in Z^*} < S | vw >< P | v >).$$

Et nous avons encore, pour toute série formelle S de $A \ll Z \gg$:

$$\begin{aligned} \forall P, \ Q \in A < Z >, \quad & Q \triangleleft (P \triangleleft S) = PQ \triangleleft S \\ & (S \triangleright P) \triangleright Q = S \triangleright PQ \\ & (P \triangleleft S) \triangleright Q = P \triangleleft (S \triangleright Q). \end{aligned}$$

1.4. Somme et produit des séries formelles

1.4.1. Somme des séries formelles

La **somme**, notée "+", de deux séries S et T de $A \ll Z \gg$ est la série $S+T$ de $A \ll Z \gg$ définie par :

$$\forall w \in Z^*, \quad < S+T|w > = < S|w > + < T|w > .$$

La somme est une opération associative et commutative.

On vérifie aisément que :

$$\forall S,\ T \in A \ll Z \gg, \quad \omega(S+T) \geq \inf\{\omega(S), \omega(T)\},$$
$$\forall P,\ Q \in A < Z >, \quad \deg(P+Q) \leq \max\{\deg(P), \deg(Q)\}.$$

1.4.2. Produit de Cauchy

Le **produit de Cauchy**, noté ".", de deux séries T et S de $A \ll Z \gg$ est la série $T.S$ de $A \ll Z \gg$ définie par :

$$\forall w \in Z^*, \quad < T.S|w > = \sum_{u,v \in Z^*, uv=w} < T|u >< S|v > .$$

Lorsqu'il n'y a pas d'ambiguïté, on omettra le symbole ".".

$A \ll Z \gg$, muni de la somme et le produit de Cauchy, est un anneau (non commutatif) :

. l'associativité est héritée de l'associativité dans Z^*,

. la distributivité se montre par simple vérification.

On identifie tout $a \in A$ à la la série $a\epsilon$. A s'identifie alors à un sous-anneau unitaire de $A \ll Z \gg$. Donc $A \ll Z \gg$ est une A_algèbre, et donc aussi un A_module (à gauche et à droite).

De même, pour le produit de Cauchy, $A < Z >$ est une sous-algèbre de $A \ll Z \gg$.

Pour toute série formelle propre en variables non commutatives S, la série formelle S^* est l'inverse à gauche et à droite de la série formelle $(1-S)$ pour le produit de Cauchy ([BR2]). En variables commutatives, S^* coïncide avec la série formelle $(1-S)^{-1}$, et on a :

$$S^{*p} = (1-S)^{-p}.$$

Soit z_0 une lettre de Z. On pose $Z_0 = Z \setminus \{z_0\}$. On note z_0^* (resp. Z^*, Z_0^*, Z^k, Z_0^k, $k \geq 0$) la série caractéristique du langage z_0^* (resp. Z^*, Z_0^*, Z^k, Z_0^k, $k \geq 0$), les séries z_0^*, Z^*, Z_0^* s'écrivent alors :

$$z_0^* = \sum_{k \geq 0} z_0^k, \quad Z^* = \sum_{k \geq 0} Z^k, \quad Z_0^* = \sum_{k \geq 0} Z_0^k.$$

Et d'après 1.1. on a :

$$Z^* = \sum_{k\geq 0} (z_0^* Z_0)^k z_0^*.$$

Par conséquent, soit S une série formelle de $A \ll Z \gg$, S peut s'écrire :

$$S = \sum_{w\in Z^*} < S|w > w = \sum_{k\geq 0} V_k(S),$$

où l'on a posé pour tout entier positif k :

$$V_k(S) = \sum_{w\in(z_0^* Z_0)^k z_0^*} < S|w > w.$$

1.4.3. Produit de mélange ([F3])

Le **produit de mélange de deux mots de** Z^* est le polynôme homogène de degré $|u| + |v|$, défini récursivement comme suit :

$$\textit{pour tout mot } u : \quad u ш \epsilon = \epsilon ш u = u$$
$$\textit{pour tous mots } u,v \textit{ et toutes lettres } x,y : uxшvy = [(ux)шv]y + [uш(vy)]x$$
$$\textit{ou encore } : xuшyv = y[(xu)шv] + x[uш(yv)]$$

Plus généralement, nous avons pour tous mots $v_1, \ldots, v_n$ de Z^*, et pour toutes lettres $x_1, \ldots, x_n$ de Z :

$$x_1v_1ш\ldots шx_nv_n = \sum_{i=1}^{n} x_i(x_1v_1ш\ldots шx_{i-1}v_{i-1}шv_iшx_{i+1}v_{i+1}ш\ldots шx_nv_n)$$
$$v_1x_1ш\ldots шv_nx_n = \sum_{i=1}^{n} (v_1x_1ш\ldots шv_{i-1}x_{i-1}шv_iшv_{i+1}x_{i+1}ш\ldots шv_nx_n)x_i.$$

Le polynôme $uшv$ est à coefficients entiers positifs et il est clair que ce produit est associatif et commutatif. Il s'en suit immédiatement les deux résultats suivants :

Lemme 1.4.3.1. : *Soit P un polynôme de $A < Z >$. On note* som(P) *la somme des coefficients de P :*

$$\mathrm{som}(P) = \sum_{w\in\mathrm{supp}(P)} < P|w > .$$

Soient u, v deux mots de Z^*. La somme des coefficients du polynôme $uшv$ est

$$\mathrm{som}(uшv) = \binom{|u| + |v|}{|u|} = \frac{(|u| + |v|)!}{|u|!\,|v|!}.$$

En particulier, pour toute lettre z de Z, on a :

$$\forall \lambda,\ \mu \geq 0, \quad z^{\lambda} ш z^{\mu} = \binom{\lambda+\mu}{\mu} z^{\lambda+\mu}.$$

<u>*Preuve*</u> :

. Le résultat est immédiat pour $u = \epsilon$ ou $v = \epsilon$.

. Supposons que le résultat soit vérifié pour tous mots u, v tels que $|uv| \leq n$.

. Soient u, v deux mots de Z^* tels que $|uv| = n+1$. On peut écrire $u = u_1 x$ et $v = v_1 y$ alors :

$$u ш v = u_1 x ш v_1 y = ((u_1 x) ш v_1) y + (u_1 ш (v_1 y)) x,$$

d'où :

$$\mathrm{som}(u ш v) = \mathrm{som}(((u_1 x) ш v_1) y) + \mathrm{som}((u_1 ш (v_1 y)) x).$$

Or d'après l'hypothèse de récurrence on a :

$$\mathrm{som}(((u_1 x) ш v_1) y) = \mathrm{som}((u_1 x) ш v_1) = \binom{|u|+|v|-1}{|u|},$$

$$\mathrm{som}((u_1 ш (v_1 y)) x) = \mathrm{som}(u_1 ш (v_1 y)) = \binom{|u|+|v|-1}{|u|-1},$$

par conséquent :

$$\mathrm{som}(u ш v) = \binom{|u|+|v|-1}{|u|} + \binom{|u|+|v|-1}{|u|-1} = \binom{|u|+|v|}{|u|} \quad \bullet$$

Lemme 1.4.3.2. : *Pour tout mot w de Z^*, pour toute lettre z de Z et pour tout entier positif n on a :*

$$(uz)^{ш n} = n[(uz)^{ш n-1} ш u]z,$$

où $(uz)^{ш n}$ désigne le polynôme $uz ш (uz)^{ш n-1}$ et $(uz)^{ш 0}$ est le mot vide.

<u>*Preuve*</u> :

Il suffit appliquer la définition et la propriété commutative du produit de mélange $\bullet$

Le produit de Hurwitz (ou **produit de mélange**) de deux séries formelles S et T de $A \ll Z \gg$, noté encore "ш", est la série formelle $S ш T$ de $A \ll Z \gg$ définie par :

$$S ш T = \sum_{u,v \in Z^*} < T|u >< S|v > u ш v.$$

Ce produit est associatif, commutatif, et distributif par rapport à la somme. Muni de la somme et du produit de mélange, $A \ll Z \gg$ est une A-algèbre commutative.

Pour toute série formelle S et pour tout entier positif n, la série formelle $S^{ш n}$ désigne la série $S ш S^{ш n-1}$ et $S^{ш 0}$ est la série 1.

Lemme 1.4.3.3. : *Les résiduels à gauche et à droite par une lettre de Z sont des dérivations pour le produit de mélange.*

Preuve : Il suffit de montrer que l'on a pour tous mots u, v de Z^* et pour toute lettre z de Z :

$$z \triangleleft (u \sqcup\!\sqcup v) = u \sqcup\!\sqcup (z \triangleleft v) + (z \triangleleft u) \sqcup\!\sqcup v.$$

. Le résultat est évident pour $u = \epsilon$ ou $v = \epsilon$.

. Nous supposons que le résultat soit vrai pour tous mots u, v de Z^* vérifiant $0 \leq |uv| \leq n$.

. Si $|uv| = n + 1$ alors nous pouvons écrire $u = u_1x$ et $v = v_1y$ avec x, $y \in Z$, u_1, $v_1 \in Z^*$ et $|u_1v| = |uv_1| = n$.

En utilisant la définition récursive du produit de mélange, nous avons pour toute lettre z :

$$\begin{aligned} z \triangleleft (u \sqcup\!\sqcup v) &= z \triangleleft [(u \sqcup\!\sqcup v_1)y + (u_1 \sqcup\!\sqcup v)x] \\ &= z \triangleleft [(u \sqcup\!\sqcup v_1)y] + z \triangleleft [(u_1 \sqcup\!\sqcup v)x] \\ &= \delta_z^y(u \sqcup\!\sqcup v_1) + \delta_z^x(u_1 \sqcup\!\sqcup v) \\ &= u \sqcup\!\sqcup (\delta_z^y v_1) + (\delta_z^x u_1) \sqcup\!\sqcup v \\ &= u \sqcup\!\sqcup (z \triangleleft v) + (z \triangleleft u) \sqcup\!\sqcup v. \end{aligned}$$

La preuve pour le résiduel à droite est similaire en utilisant la définition récursive symétrique •

Définition 1.4.3.1. : ([F3]) *Soit S une série formelle de $A \ll Z \gg$. La série S est* **échangeable** *si et seulement si deux mots ne différant que par l'ordre des lettres, ont même coefficient. On peut aussi écrire pour tous mots u, v de Z^* :*

$$(\forall z \in Z, \quad |u|_z = |v|_z) \quad \Rightarrow \quad < S|u > = < S|v > .$$

Il s'en suit que S est échangeable si et seulement si S peut s'écrire sous la forme :

$$S = \sum_{\alpha_0,\ldots,\alpha_m \geq 0} s_{\alpha_0,\ldots,\alpha_m} z_0^{\alpha_0} \sqcup\!\sqcup \ldots \sqcup\!\sqcup z_m^{\alpha_m}.$$

On peut montrer facilement que pour toutes séries sur une seule lettre

$$P_1 = \sum_{n_1 \geq 0} < P_1 | z_{i_1}^{n_1} > z_{i_1}^{n_1}, \ldots, P_k = \sum_{n_k \geq 0} < P_k | z_{i_k}^{n_k} > z_{i_k}^{n_k},$$

la série $P_1 ш \ldots ш P_k$ est échangeable car :

$$P_1 ш \ldots ш P_k = \sum_{n_1, \ldots, n_k \geq 0} (< P_1 | z_{i_1}^{n_1} > \ldots < P_k | z_{i_k}^{n_k} >) z_{i_1}^{n_1} ш \ldots ш z_{i_k}^{n_k}.$$

On vérifie facilement que la série $\left(\sum_{z \in Z} \alpha_z z\right)^*$ est échangeable car :

$$\left(\sum_{z \in Z} \alpha_z z\right)^* = (\alpha_{z_0} z_0)^* ш \ldots ш (\alpha_{z_m} z_m)^*.$$

1.4.4. Crochets de Lie

On définit **le crochet de Lie** de deux polynômes P, Q par :

$$[P, Q] = PQ - QP.$$

Il est anticommutatif et vérifie l'identité de Jacobi.

On note $Lie < Z >$ le plus petit sous A-module de $A \ll Z \gg$ qui contient des lettres de Z et qui est stable par crochet de Lie. On sait que $Lie < Z >$ est **l'algèbre de Lie libre sur** Z ([V1]). Un élément de $Lie < Z >$ est appelé **polynôme de Lie** sur Z. Tout polynôme de Lie est propre.

Une série S de $A \ll Z \gg$, est appelée **série de Lie** si elle s'écrit (de manière unique) :

$$S = \sum_{k \geq 1} P_k,$$

où pour tout $k \geq 1$, P_k est un polynôme de Lie homogène de degré k ([JO1]).

On définit **le crochet de Lie** de deux séries $S = \sum_{k \geq 1} P_k$, $T = \sum_{l \geq 1} Q_l$ par :

$$[S, T] = \sum_{k, l \geq 1} [P_k, Q_l],$$

où pour tout k, $l \geq 1$, P_k, Q_l sont des polynômes de Lie homogènes de degré k et l respectivement.

Pour ce crochet, l'ensemble $Lie \ll Z \gg$ des séries de Lie sur Z à coefficients dans A est une **algèbre de Lie**. Toute série de Lie est propre.

1.5. Topologie discrète sur $A \ll Z \gg$ et sur $Lie \ll Z \gg$

L'anneau A est supposé muni d'une topologie discrète. On définit une distance ultramétrique sur $A \ll Z \gg$ en posant, pour toutes séries formelles S et T de $A \ll Z \gg$:

$$d(S,T) = 2^{-\omega(S-T)}.$$

On obtient un le système fondamental de voisinages de zéro $\{\mathcal{O}_k\}_{k\geq 0}$ dans $A \ll Z \gg$ en posant :

$$\forall k \geq 0, \quad \mathcal{O}_k = \{S \in A \ll Z \gg \mid \ \omega(S) \geq k\}.$$

On a alors :

$$\forall k \geq 0, \quad \mathcal{O}_k = \{S \in A \ll Z \gg \mid \ \ \mathrm{supp}(S) \subset Z^k Z^*\},$$

par conséquent :

$$\bigcap_{k\geq 0} \mathcal{O}_k = \{0\}.$$

La topologie discrète de $Lie \ll Z \gg$ est la topologie induite par la topologie discrète de $A \ll Z \gg$. On a donc dans $Lie \ll Z \gg$ le système fondamental de voisinages de zéro :

$$\forall k \geq 0, \quad \mathcal{O}_k^L = \mathcal{O}_k \bigcap Lie \ll Z \gg .$$

$A \ll Z \gg$ (resp. $Lie \ll Z \gg$) est **le complété séparé** de $A < Z >$ (resp. $Lie < Z >$) pour ce système de voisinages $\{\mathcal{O}_k\}_{k\geq 0}$ (resp. $\{\mathcal{O}_k^L\}_{k\geq 0}$).

Soit $(S_k)_{k\geq 0}$ une suite de séries formelles de $A \ll Z \gg$. La **limite** dans $A \ll Z \gg$ de cette suite peut être calculée coefficient par coefficient. Plus précisément :

$$S = \lim_{k\to\infty} S_k \qquad \Longleftrightarrow \qquad \forall w \in Z^*, \quad < S|w > = \lim_{k\to\infty} < S_k|w > .$$

Donc la suite $(S_k)_{k\geq 0}$ converge si et seulement si, pour tout mot w de Z^*, la suite $(< S_k|w >)_{k\geq 0}$ est stationnaire.

Soit S une série formelle propre de $A \ll Z \gg$. **L'exponentielle** de S est la série

$$\exp(S) = \sum_{k\geq 0} \frac{S^k}{k!}.$$

Cette somme a un sens puisque S est propre, et que l'on a $\omega(S^k) \geq k$.

On appelle **exponentielle de Lie** toute série qui est l'exponentielle d'une série de Lie.

Soit S une série formelle de $A \ll Z \gg$ ($S = 1 + T$ avec T une série formelle propre de $A \ll Z \gg$). Le **logarithme** de S est la série formelle

$$\log(S) = \sum_{k\geq 1} \frac{(-1)^{k+1}}{k}(S-1)^k = \sum_{k\geq 1} \frac{(-1)^{k+1}}{k} T^k.$$

En fait, chacune des fonctions "exp" et "log" est l'inverse de l'autre. On peut vérifier aisément que les fonctions "exp" et "log" ainsi définies sont continues.

1.6. Séries majorantes

Nous supposons ici que A est une algèbre sur $\mathbb{R}$ et A est un espace vectoriel normé complet. Nous notons la norme sur A par $\|.\|$.

Définition 1.6.1.1. : *Soit ϕ une fonction positive réelle définie sur Z^*. Soit S une série formelle de $A \ll Z \gg$. S est dite* **ϕ−exponentiellement majorée** *si elle vérifie la condition suivante :*

$$\exists K \in \mathbb{R}_+, \quad \exists n \in \mathbb{N}, \quad \forall w \in Z^n Z^*, \quad \|< S|w >\| \leq K\frac{\phi(w)}{|w|!}.$$

Nous notons par $A^{\phi-em}\ll Z \gg$ l'ensemble de toutes les séries formelles de $A \ll Z \gg$ qui sont ϕ−exponentiellement majorée.

Lemme 1.6.1.1. : *Soit ϕ un morphisme de monoïde, de Z^* dans $\mathbb{R}_+$. Soit S une série formelle ϕ−exponentiellement majorée. Pour tout mot u de Z^*, la série $S \triangleright u$ est aussi ϕ−exponentiellement majorée.*

<u>*Preuve*</u> : En effet, puisque S est ϕ−exponentiellement majorée, alors :

$$\exists K \in \mathbb{R}_+, \quad \exists n \in \mathbb{N}, \quad \forall w \in Z^n Z^*, \quad \|< S|w >\| \leq K\frac{\phi(w)}{|w|!}.$$

Et d'autre part, nous avons :

$$\forall w \in Z^n Z^*, \quad < S \triangleright u|w > = < S|uw > .$$

Par conséquent :

$$\forall w \in Z^n Z^*, \quad \|< S \triangleright u|w >\| \leq K\frac{\phi(uw)}{|uw|!} \leq K\phi(u)\frac{\phi(w)}{|w|!},$$

et on pose dans ces conditions $K_1 = \phi(u)K$ •

Nous déduisons le corollaire suivant :

Corollaire 1.6.1.1. : *Soit ϕ un morphisme de monoïde, de Z^* dans $\mathbb{R}_+$. Soit S une série formelle ϕ−exponentiellement majorée. Pour tout polynôme P de $A < Z >$, la série $S \triangleright P$ est ϕ−exponentiellement majorée.*

<u>*Preuve*</u> : Comme précédemment, nous prenons :

$$K_1 = \max_{u \in \mathrm{supp}(P)} \phi(u)K \quad \bullet$$

Définition 1.6.1.2. : *Soit χ une fonction positive réelle définie sur Z^*. Soit S une série formelle de $A \ll Z \gg$. S vérifie la* **condition de $\chi-$croissance** *si elle vérifie :*

$$\exists K \in I\!R_+, \quad \exists n \in I\!N, \quad \forall w \in Z^n Z^*, \quad \| < S|w > \| \leq K\chi(w)\, |w|!.$$

Nous notons par $A^{\chi-cc}\ll Z \gg$ l'ensemble de toutes les séries formelles de $A \ll Z \gg$ qui vérifient la condition de $\chi-$croissance.

Définition 1.6.1.3. : *Soit Cl une classe de séries formelle. Soit S une série formelle de $A \ll Z \gg$. S est dite* **continue sur Cl** *si pour toute série formelle H de Cl, la série de terme générale $< H|w >< S|w >$ est normalement convergente. Dans ce cas, nous notons*

$$< S\|H > = \sum_{w \in Z^*} < H|w >< S|w > .$$

En particulier, soit ϕ une fonction positive réelle définie sur Z^*, S est dite **$\phi-$exponentiellement continue** *si elle est continue sur $A^{\phi-em}\ll Z \gg$.*

Nous notons par $A^{\phi-ec}\ll Z \gg$ l'ensemble de toutes les séries formelles qui sont $\phi-$exponentiellement continue.

Notons que nous avons, pour toute fonction positive réelle ϕ définie sur Z^*, l'inclusion suivante :

$$A < Z > \subset A^{\phi-ec}\ll Z \gg .$$

D'autre part, nous avons $A < Z > = A^{0-ec} \ll Z \gg$ (pour $\phi = 0$). Par conséquent, tout polynôme de $A < Z >$ est $0-$exponentiellement continu.

Lemme 1.6.1.2. : *Etant donné une série formelle S, si S est continue sur une classe Cl de séries formelles, alors :*

$$\forall u \in Z^*, \quad \forall H \in Cl, \quad < u \triangleleft S\|H > = < S\|Hu > .$$

<u>*Preuve*</u> : Puisque S est continue sur Cl alors pour toute série formelle H de Cl, nous avons :

$$\begin{aligned} < S\|Hu > &= \sum_{w \in Z^*} < S|w >< Hu|w > \\ &= \sum_{v \in Z^*} < S|vu >< Hu|vu > \\ &= \sum_{v \in Z^*} < u \triangleleft S|v >< H|v > \\ &= < u \triangleleft S\|H > \quad \bullet \end{aligned}$$

Lemme 1.6.1.3. : *Nous supposons ici que ϕ et χ sont deux morphismes de monoïde, de Z^* dans $\mathbb{R}_+$, vérifiant les conditions suivantes :*

$$\sum_{z\in Z} \chi(z)\phi(z) < 1.$$

Alors dans ce cas, pour toute série formelle F de $A^{\chi-cc}\ll Z \gg$, F est continue sur $A^{\phi-em}\ll Z \gg$.

Preuve : Si ϕ et χ vérifient la condition $\sum_{z\in Z} \chi(z)\phi(z) < 1$ alors la série

$$\Big(\sum_{z\in Z} \chi(z)\phi(z)\Big)^* = \sum_{w\in Z^*} \chi(w)\phi(w)$$

est normalement convergente. Etant donné une série formelle F vérifiant la condition de $\chi-$croissance, nous avons :

$$\exists K_1 \in \mathbb{R}_+, \quad \exists n_1 \in \mathbb{N}, \quad \forall w \in Z^{n_1}Z^*, \quad \|< F|w >\| \leq K_1\chi(w)\,|w|!.$$

Etant donné une série formelle C de $A^{\phi-em}\ll Z \gg$, nous avons :

$$\exists K_2 \in \mathbb{R}_+, \quad \exists n_2 \in \mathbb{N}, \quad \forall w \in Z^{n_2}Z^*, \quad \|< C|w >\| \leq K_2\frac{\phi(w)}{|w|!}.$$

Nous posons $n = \max\{n_1, n_2\}$, pour tout mot w de Z^nZ^*, nous avons les inégalitées suivantes :

$$\|< C|w >< F|w >\| \leq K_1K_2\chi(w)\phi(w).$$

La série de terme général $< C|w >< F|w >$ est donc normalement convergente. Par conséquent, F est continue sur $A^{\phi-em}\ll Z \gg$ •

1.7. Séries formelles en variables commutatives ([A1], [G2])

Soit $\mathbb{K}$ un corps ($\mathbb{R}$ ou $\mathbb{C}$) muni d'une norme notée $|.|$. Soient $\theta_1, \ldots, \theta_N$, des indéterminées commutatives sur $\mathbb{K}$. Nous notons $\Theta = \{\theta_1, \ldots, \theta_N\}$. L'algèbre des séries formelles (resp. polynômes) sur Θ à coefficients dans $\mathbb{K}$ est notée par $\mathbb{K}[\![\Theta]\!]$ (resp. $\mathbb{K}[\Theta]$). Un élément de $\mathbb{K}[\![\Theta]\!]$ est une somme infinie :

$$f = \sum_{i_1,\ldots,i_N \geq 0} f_{i_1,\ldots,i_N}\theta_1^{i_1}\ldots\theta_N^{i_N}, \quad f_{i_1,\ldots,i_N} \in \mathbb{K}.$$

On appelle **dérivée partielle première** de la série formelle f par rapport à la $k^{ième}$ composante, $k \in [1..N]$, la série formelle de $\mathbb{K}[\![\Theta]\!]$:

$$D_kf = \frac{\partial f}{\partial\theta_k} = \sum_{\substack{i_1,\ldots,i_{k-1},\\ i_{k+1},\ldots,i_N \geq 0}} \sum_{i_k\geq 1} i_kf_{i_1,\ldots,i_N}\theta_1^{i_1}\ldots\theta_k^{i_k-1}\ldots\theta_N^{i_N}.$$

De la même manière, pour tout entier positif j_k, on appelle **j_k-dérivée partielle** de la série formelle f par rapport à la $k^{ième}$ composante, $k \in [1..N]$, la série formelle de $I\!K[\![\Theta]\!]$:

$$D_k^{j_k} f = j_k! \sum_{\substack{i_1,\dots,i_{k-1},\\ i_{k+1},\dots,i_N \geq 0}} \sum_{i_k \geq j_k} \binom{i_k}{i_k - j_k} f_{i_1,\dots,i_N} \theta_1^{i_1} \dots \theta_k^{i_k - j_k} \dots \theta_N^{i_N}.$$

Plus généralement, pour tous entiers positifs $j_1, \dots, j_N$, nous avons :

$$D_1^{j_1} \dots D_N^{j_N} f = j_1! \dots j_N! \sum_{i_1 \geq j_1, \dots, i_N \geq j_N} \binom{i_1}{i_1 - j_1} \dots \binom{i_N}{i_N - j_N} f_{i_1,\dots,i_N} \theta_1^{i_1 - j_1} \dots \theta_N^{i_N - j_N}.$$

Nous notons $\mathcal{D}$ l'ensemble des symboles de **dérivation partielle** $\{D_k\}_{1 \leq k \leq N}$. Le monoïde commutatif libre engendré par l'alphabet fini $\mathcal{D}$ est $\mathcal{D}^{\&} = \bigcup_{n \geq 0} \mathcal{D}^n$, où $\mathcal{D}^n$ est l'ensemble de tous les mots ordonnés de longueur exactement égale à n. Pour tout mot ordonné $\alpha \in \mathcal{D}^{\&}$, il existe N entiers positifs $j_1, \dots, j_N$ tels que $\alpha = D_1^{j_1} \dots D_N^{j_N}$. Dans ce cas, le degré de la lettre D_k dans le mot ordonné α est j_k.

Définition 1.7.1. : *Soit f une série formelle de $I\!K[\![\Theta]\!]$. Nous posons :*

$$E(f) = \{\rho \in I\!R_+^N : \exists C_f \in I\!R_+ \text{ tel que pour tous } i_1, \dots, i_N \geq 0, \; |f_{i_1,\dots,i_N}| \rho_1^{i_1} \dots \rho_N^{i_N} \leq C_f\}$$

$$\mathring{E}(f): \text{ l'intérieur de } E(f) \text{ dans } I\!R^N.$$

$$Conv(f) = \text{domaine de convergence de } f = \{q \in I\!K^N : (|q_1|, \dots, |q_N|) \in \mathring{E}(f)\}.$$

On dit que f est **convergente** *si et seulement si $Conv(f) \neq \emptyset$. Soit $\mathcal{U}$ un ouvert de $I\!K^N$ et soit q un point de $I\!K^N$. On dit que f est* **convergente en** *q (resp.* **sur** *$\mathcal{U}$) si et seulement si $q \in Conv(f)$ (resp. $\mathcal{U} \subset Conv(f)$). On pose :*

$$I\!K^{conv}[\![\Theta]\!] = \{f \in I\!K[\![\Theta]\!] : \; Conv(f) \neq \emptyset\}.$$

Soit q un point de $Conv(f)$. On peut alors trouver deux vecteurs $\rho, \mathring{\rho}$ de $I\!R_+^N$ et une constante C_f tels que l'on ait $|q_1| < \mathring{\rho}_1 < \rho_1, \dots, |q_N| < \mathring{\rho}_N < \rho_N$ et pour tous entiers positifs $i_1 \dots, i_N$, $|f_{i_1,\dots,i_N}| \rho_1^{i_1} \dots \rho_N^{i_N} \leq C_f$.

Définition 1.7.2. : *Le triplet* $(\rho, \overset{\circ}{\rho}, C_f)$ *est appelé* **module de convergence pour f en q.**

Si $f_1, \ldots, f_k$ sont des séries de $I\!K^{conv}[\![\Theta]\!]$, et si q est un point de $\bigcap_{l=1}^{k} Conv(f_l)$, alors on vérifie qu'il existe deux vecteurs $\rho, \overset{\circ}{\rho}$ de $I\!R_+^N$ tels que l'on ait $|q_1| < \overset{\circ}{\rho}_1 < \rho_1, \ldots, |q_N| < \overset{\circ}{\rho}_N < \rho_N$, et pour chaque f_l, une constante particulière C_{f_l} telle que le triplet $(\rho, \overset{\circ}{\rho}, C_{f_l})$ soit un module de convergence pour f_l en q.

Supposons que $Conv(f)$ est non vide et soit $(\rho, \overset{\circ}{\rho}, C_f)$ un module de convergence pour f en $q \in Conv(f)$. Nous avons la majoration suivante :

$$|f_{i_1,\ldots,i_N} q_1^{i_1} \ldots q_N^{i_N}| \leq \frac{C_f}{\rho_1^{i_1} \ldots \rho_N^{i_N}} \overset{\circ}{\rho}_1^{i_1} \ldots \overset{\circ}{\rho}_N^{i_N}.$$

Par conséquent, pour tout point q de $Conv(f)$, la série $f_{|q}$ est majorée terme à terme par la "série majorante"

$$C_f \prod_{k=1}^{N} \left(1 - \frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1}.$$

D'où $f_{|q}$ est uniformément absolument convergente dans le polycylindre :

$$\{q \in I\!K^N : \ |q_1| < \overset{\circ}{\rho}_1, \ldots, |q_N| < \overset{\circ}{\rho}_N\},$$

qui est un ouvert de $I\!K^N$, par conséquent $Conv(f)$ est un ouvert de $I\!K^N$.

Soient r et τ deux réels définis comme suit :

$$\tau = \min_{1 \leq k \leq N} \rho_k \quad \text{et} \quad r = \max_{1 \leq k \leq N} \frac{\overset{\circ}{\rho}_k}{\rho_k}.$$

Soient $G(r)$ et $F(r)$ les séries :

$$G(r) = (1-r)^{-1} \quad \text{et} \quad F(r) = (1-r)^{-N} = G^N(r).$$

Notons aussi Δ l'opérateur différentiel suivant :

$$\Delta = F(r)\frac{\mathrm{d}}{\mathrm{d}r}.$$

Nous avons les résultats suivants :

Proposition 1.7.1. :

$$
\begin{array}{lll}
(a) & \forall j \geq 0, & \dfrac{\mathrm{d}^j}{\mathrm{d}r^j} G(r) = j! G^{j+1}(r). \\
(b) & \forall j \geq 0, & \dfrac{\mathrm{d}^j}{\mathrm{d}r^j} F(r) = \dfrac{(N+j-1)!}{(N-1)!} F(r) G^j(r). \\
(c) & \forall p, q \geq 0, & \dfrac{\mathrm{d}}{\mathrm{d}r} F^p(r) G^q(r) = (pN+q) F^p(r) G^{q+1}(r). \\
(d) & \forall p \geq 0, & \Delta^p F(r) = N(2N+1)\dots(pN+(p-1)) F^{p+1}(r) G^p(r). \\
(e) & \forall p \geq 0, & \Delta^p F(r) \leq \dfrac{(N+1)^p}{(1-r)^N} (1-r)^{-p(N+1)} p!.
\end{array}
$$

<u>*Preuve*</u> :

(a) La preuve est immédiate.

(b) Le résultat est immédiat pour $j = 0$. Supposons que le résultat soit vrai pour tout $0 \leq i \leq j$. Pour $i = j+1$, nous avons :

$$
\begin{aligned}
\frac{\mathrm{d}^{j+1}}{\mathrm{d}r^{j+1}} F(r) &= \frac{\mathrm{d}}{\mathrm{d}r}\left[\frac{(N+j-1)!}{(N-1)!} F(r) G^j(r)\right] \\
&\qquad \text{(d'après l'hypothèse de récurrence)} \\
&= \frac{(N+j-1)!}{(N-1)!}[N F(r) G(r)^{j+1} + j F(r) G(r)^{j+1}] \\
&\qquad \text{(d'après l'hypothèse de récurrence et d'après (a))} \\
&= \frac{(N+j)!}{(N-1)!} F(r) G(r)^{j+1}.
\end{aligned}
$$

(c) Dérivons le produit $F^p(r)G^q(r)$:

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}r}\Big[F^p(r)G^q(r)\Big] &= pF^{p-1}(r)\left[\frac{\mathrm{d}}{\mathrm{d}r}F(r)\right]G^q(r) + qF^p(r)G^{q-1}(r)\left[\frac{\mathrm{d}}{\mathrm{d}r}G(r)\right] \\
&= pNF^p(r)G^{q+1}(r) + qF^p(r)G^{q+1}(r) \quad \text{(d'après (a) et (b))} \\
&= (pN+q)F^p(r)G^{q+1}(r).
\end{aligned}
$$

(d) Le résultat est immédiat pour $p = 0$. Supposons que le résultat soit vrai jusqu'au rang $p-1$. Au rang p, puisque $\Delta^p F(r) = F(r)\frac{\mathrm{d}}{\mathrm{d}r}[\Delta^{p-1}F(r)]$, d'après l'hypothèse de récurrence, nous avons :

$$
\begin{aligned}
\Delta^p F(r) &= F(r)\frac{\mathrm{d}}{\mathrm{d}r}[N(2N+1)\dots((p-1)N+(p-2))F^p(r)G^{p-1}(r)] \\
&= N(2N+1)\dots((p-1)N+(p-2))F(r)\frac{\mathrm{d}}{\mathrm{d}r}[F^p(r)G^{p-1}(r)] \\
&= N(2N+1)\dots((p-1)N+(p-2))(pN+(p-1))F^{p+1}(r)G^p(r).
\end{aligned}
$$

La dernière égalité s'obtient en appliquant (c).

(e) D'après (d), nous pouvons écrire :

$$\begin{aligned}\Delta^p F(r) &= ((N+1)-1)(2(N+1)-1)\dots(p(N+1)-1))F^{p+1}(r)G^p(r)\\ &= \prod_{k=1}^{p}(k(N+1)-1)F^{p+1}(r)G^p(r).\end{aligned}$$

Pour tout entier $k > 1$, on peut majorer $k(N+1)-1$ par $k(N+1)$. Il s'en déduit immédiatement :

$$\Delta^p F(r) \leq (N+1)^p p! F^{p+1}(r)G^p(r).$$

Puique $G(r) = (1-r)^{-1}$ et $F(r) = (1-r)^{-N}$, nous avons alors :

$$\Delta^p F(r) \leq \frac{(N+1)^p}{(1-r)^N}(1-r)^{-p(N+1)}p! \quad \bullet$$

Remarque 1.7.1. : *Soient j un entier positif et $l \in [1..N]$, nous avons :*

$$\frac{\partial^j}{\partial \overset{\circ}{\rho}{}_l^j}\prod_{k=1}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1} = \frac{j!}{\rho_l^j}\left(1-\frac{\overset{\circ}{\rho}_l}{\rho_l}\right)^{-j-1}\prod_{k=1,k\neq l}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1}.$$

Plus généralement nous avons :

$$\frac{\partial^{j_1}\dots\partial^{j_N}}{\partial \overset{\circ}{\rho}{}_l^{j_1}\dots\partial \overset{\circ}{\rho}{}_N^{j_N}}\prod_{k=1}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1} = \prod_{k=1}^{N}\frac{j_k!}{\rho_k^{j_k}}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-j_k-1}.$$

Proposition 1.7.2. : *Soit f une série de $\mathbb{K}^{conv}[\![\Theta]\!]$. Soit q un point de $Conv(f)$ et soit $(\rho, \overset{\circ}{\rho}, C_f)$ est un module de convergence pour f en q. Pour tous entiers positifs $j_1, \dots, j_N$, nous avons :*

$$\left|(D_1^{j_1}\dots D_N^{j_N}f)_{|q}\right| \leq C_f \frac{\partial^{j_1}\dots\partial^{j_N}}{\partial \overset{\circ}{\rho}{}_1^{j_1}\dots\partial \overset{\circ}{\rho}{}_N^{j_N}}\prod_{k=1}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1}.$$

<u>*Preuve*</u> : En effet, soit q un point de $Conv(f)$ et soit $(\rho, \overset{\circ}{\rho}, C_f)$ est un module de convergence pour f en q. Dans ce cas, pour tous entiers positifs

$j_1,\dots,j_N$, en tout point q de $Conv(f)$, la série $(D_1^{j_1}\dots D_N^{j_N}f)_{|q}$ est majorée en valeur absolue terme à terme par l'expression suivante :

$$\begin{aligned}
&C_f\, j_1!\dots j_N! \sum_{i_1\geq j_1}\dots\sum_{i_N\geq j_N} \binom{i_1}{i_1-j_1}\dots\binom{i_N}{i_N-j_N}\frac{\overset{\circ}{\rho}_1^{\,i_1-j_1}\dots\overset{\circ}{\rho}_N^{\,i_N-j_N}}{\rho_1^{i_1}\dots\rho_N^{i_N}}\\
&= C_f\frac{j_1!\dots j_N!}{\rho_1^{j_1}\dots\rho_N^{j_N}}\sum_{l_1\geq 0,\dots,l_N\geq 0}\binom{l_1+j_1}{l_1}\dots\binom{l_N+j_N}{l_N}\left(\frac{\overset{\circ}{\rho}_1}{\rho_1}\right)^{l_1}\dots\left(\frac{\overset{\circ}{\rho}_N}{\rho_N}\right)^{l_N}\\
&= C_f\prod_{k=1}^{N}\frac{j_k!}{\rho_k^{j_k}}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-j_k-1}\\
&= C_f\frac{\partial^{j_1}\dots\partial^{j_N}}{\partial\overset{\circ}{\rho}_N^{\,j_1}\dots\partial\overset{\circ}{\rho}_l^{\,j_N}}\prod_{k=1}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1} \qquad \text{(d'après la remarque 1.7.1.)} \quad \bullet
\end{aligned}$$

Proposition 1.7.3. : *Pour tous entiers positifs $j_1,\dots,j_N$, nous avons :*

$$\frac{\partial^{j_1}\dots\partial^{j_N}}{\partial\overset{\circ}{\rho}_1^{\,j_1}\dots\partial\overset{\circ}{\rho}_N^{\,j_N}}\prod_{k=1}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1} \leq \frac{1}{\binom{N+p-1}{p}\tau^p}\frac{\mathrm{d}^p}{\mathrm{d}r^p}F(r) \text{ avec } p=j_1+\dots+j_N.$$

<u>*Preuve*</u> : En effet, d'après la remarque 1.7.1., nous avons :

$$\frac{\partial^{j_1}\dots\partial^{j_N}}{\partial\overset{\circ}{\rho}_l^{\,j_1}\dots\partial\overset{\circ}{\rho}_N^{\,j_N}}\prod_{k=1}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1} = \frac{j_1!\dots j_N!}{\rho_1^{j_1}\dots\rho_N^{j_N}}\prod_{k=1}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-j_k-1}.$$

Nous déduisons alors :

$$\begin{aligned}
\frac{\partial^{j_1}\dots\partial^{j_N}}{\partial\overset{\circ}{\rho}_l^{\,j_1}\dots\partial\overset{\circ}{\rho}_l^{\,j_N}}\prod_{k=1}^{N}\left(1-\frac{\overset{\circ}{\rho}_k}{\rho_k}\right)^{-1} &\leq \frac{p!}{\tau^p}F(r)G^p(r)\\
&= \frac{1}{\binom{N+p-1}{p}\tau^p}\frac{\mathrm{d}^p}{\mathrm{d}r^p}F(r)\\
&\qquad \text{(d'après la proposition 1.7.1.(b))} \quad \bullet
\end{aligned}$$

Les propositions 1.7.2. et 1.7.3. expriment qu'en tout point q de $Conv(f)$, les dérivées successives de la série f évaluées en q sont majorées en valeur absolue terme à terme par les dérivées successives (de même ordre) de la "série majorante". Ces dernières dérivées successives sont aussi majorées (à coefficient près) par les dérivées successives (de même ordre) de la série $F(r)$. Nous pouvons ainsi conclure que si la série f évaluée en q est convergente alors ses dérivées successives évaluées en q sont aussi convergentes :

Théorème 1.7.1. : *Soit f une série de $\mathbb{K}^{conv}[\![\Theta]\!]$. Pour tous entiers positifs $j_1, \ldots, j_N$, nous avons :*

$$(a) \quad \left|(D_1^{j_1} \ldots D_N^{j_N} f)_{|q}\right| \leq \frac{C_f}{\binom{N+p-1}{p}\tau^p} \frac{\mathrm{d}^p}{\mathrm{d}r^p} F(r) \quad \text{avec} \quad p = j_1 + \ldots + j_N.$$

$$(b) \quad Conv(f) \subset Conv(D_1^{j_1} \ldots D_N^{j_N} f).$$

Soient f_0, f_1, f_2 trois séries de $\mathbb{K}^{conv}[\![\Theta]\!]$, soit q est un point de $\bigcap_{l=0}^{2} Conv(f_l)$, et soient $(\rho, \mathring{\rho}, C_{f_l})$, $(l = 0, 1, 2)$, les modules de convergence pour f_l en q. Soient D_{i_1} et D_{i_2} deux opérateurs différentiels linéaires de $\mathcal{D}$. Nous avons :

$$f_2 D_{i_2} f_1 D_{i_1} f_0 = f_2[(D_{i_2} f_1)(D_{i_1} f_0) + f_1 D_{i_2} D_{i_1} f_0].$$

D'après le théorème 1.7.1., nous avons :

$$\left|(f_2 D_{i_2} f_1 D_{i_1} f_0)_{|q}\right| \leq C_{f_2} C_{f_1} C_{f_0} F(r) \left[\left(\frac{1}{N\tau} \frac{\mathrm{d}}{\mathrm{d}r} F(r)\right)^2 + F(r) \frac{1}{\binom{N+1}{2}\tau^2} \frac{\mathrm{d}^2}{\mathrm{d}r^2} F(r)\right].$$

Puisque pour tous entiers $n \geq 1$ on a :

$$\binom{n+1}{2} \leq n^2,$$

alors :

$$\begin{aligned}
\left|(f_2 D_{i_2} f_1 D_{i_1} f_0)_{|q}\right| &\leq \frac{C_{f_2} C_{f_1} C_{f_0}}{\binom{N+1}{2}\tau^2} F(r) \left[\left(\frac{\mathrm{d}}{\mathrm{d}r} F(r)\right)^2 + F(r) \frac{\mathrm{d}^2}{\mathrm{d}r^2} F(r)\right] \\
&= \frac{C_{f_2} C_{f_1} C_{f_0}}{\binom{N+1}{2}\tau^2} F(r) \frac{\mathrm{d}}{\mathrm{d}r} \left[F(r) \frac{\mathrm{d}}{\mathrm{d}r} F(r)\right] \\
&= \frac{C_{f_2} C_{f_1} C_{f_0}}{\binom{N+1}{2}\tau^2} \Delta^2 F(r).
\end{aligned}$$

Plus généralement, nous avons le théorème suivant :

Théorème 1.7.2. : *Soient $f_0, f_1, \ldots, f_p$ les séries de $\mathbb{K}^{conv}[\![\Theta]\!]$, soit q un point de $\bigcap_{l=0}^{p} Conv(f_l)$, et soient $(\rho, \mathring{\rho}, C_{f_l})$, $(l = 0, 1, \ldots, p)$, des modules de convergence pour f_l en q. Soient $D_{i_1}, \ldots, D_{i_p}$ des opérateurs différentiels de $\mathcal{D}$. Nous avons :*

$$\left|(f_p D_{i_p} \ldots f_1 D_{i_1} f_0)_{|q}\right| \leq \frac{C_{f_p} \ldots C_{f_1} C_{f_0}}{\binom{N+p-1}{p}\tau^p} \Delta^p F(r).$$

1.8. Polysystème

Définition 1.8.1. : *Soit* $Z = \{z_0, z_1, \ldots, z_m\}$ *un alphabet fini. On appelle* **polysystème** *([L4]) sur* Θ *relatif à l'alphabet fini* Z*, la donnée d'un opérateur différentiel linéaire associé (ou encore une* **dérivation de Lie***) à la lettre* $z \in Z$ *:*

$$A_z = \sum_{D \in \mathcal{D}} A_z^D D,$$

avec les $A_z^D, D \in \mathcal{D}, z \in Z$*, sont des séries de* $\mathbb{K}^{conv}[\![\Theta]\!]$ *telles que pour tout point* $q \in \bigcap_{D \in \mathcal{D}} Conv(A_z^D)$*, pour tout module de convergence* $(\rho, \mathring{\rho}, C_z)$ *en* q *de* A_z^D*,* $(D \in \mathcal{D},\ z \in Z)$*, on ait :*

$$\forall z \in Z, \quad \forall D \in \mathcal{D}, \quad |(A_z^D)_{|q}| \le C_z F(r),$$

où $r = \max_{1 \le k \le N} \frac{\mathring{\rho}_k}{\rho_k}$ *et* $F(r) = (1 - r)^{-N}$.

1.8.1. Action à gauche de $A < Z >$ sur $\mathbb{K}^{conv}[\![\Theta]\!]$ ([HJ5])

Soit "*" **l'action à gauche de** Z **sur** $\mathbb{K}^{conv}[\![\Theta]\!]$, qui à chaque lettre z de Z, et à toute série formelle f de $\mathbb{K}^{conv}[\![\Theta]\!]$, fait correspondre la dérivée de Lie de f par rapport à A_z :

$$z * f = A_z \circ f.$$

On peut étendre l'action "*" à Z^* comme suit :

$$\forall w \in Z^*, \quad \forall f \in \mathbb{K}^{conv}[\![\Theta]\!], \quad w * f = A_w \circ f.$$

Lemme 1.8.1.1. : *Soient* f *et* g *deux séries de* $\mathbb{K}^{conv}[\![\Theta]\!]$*. Nous avons pour tout mot* w *de* Z^* *:*

$$w * (fg) = \sum_{u,v \in Z^*} < u ⧢ v | w > (u * f)(v * g).$$

<u>*Preuve*</u> :

. Le résultat est immédiat pour $w = \epsilon$.

. Supposons que le résultat soit vrai pour tout mot w, $0 \le |w| \le n$.

. Pour w, $|w| = n + 1$, w peut s'écrire $w = w_1 z$, $w_1 \in Z^*$. Alors nous avons :

$$w * (fg) = w_1 * [z * (fg)] = w_1 * [(z * f)g + f(z * g)].$$

D'après l'hypothèse de récurrence, nous avons :

$$\begin{aligned}
w*(fg) &= \sum_{u_1,v\in Z^*} < u_1 ш v|w_1 > (u_1 z * f)(v*g) \\
&+ \sum_{u,v_1\in Z^*} < u ш v_1|w_1 > (u*f)(v_1 z * g) \\
&= \sum_{u,v\in Z^*} < (z \triangleleft u) ш v|w_1 > (u*f)(v*g) \\
&+ \sum_{u,v\in Z^*} < u ш (z \triangleleft v)|w_1 > (u*f)(v*g) \\
&= \sum_{u,v\in Z^*} [< (z \triangleleft u) ш v|w_1 > + < u ш (z \triangleleft v)|w_1 >](u*f)(v*g),
\end{aligned}$$

puisque le résiduel à gauche est une dérivation pour le produit de mélange (lemme 1.4.3.3.) alors nous avons :

$$\begin{aligned}
w*(fg) &= \sum_{u,v\in Z^*} < z \triangleleft (u ш v)|w_1 > (u*f)(v*g) \\
&= \sum_{u,v\in Z^*} < u ш v|w_1 z > (u*f)(v*g) \\
&= \sum_{u,v\in Z^*} < u ш v|w > (u*f)(v*g) \quad \bullet
\end{aligned}$$

Le lemme 1.8.1.1. peut être interprété en termes de dérivation de Lie comme suit :

$$\forall f,\ g \in \mathbb{K}^{conv}[\![\Theta]\!],\ \forall w \in Z^*,\ A_w \circ (fg) = \sum_{u,v\in Z^*} < u ш v|w > (A_u \circ f)(A_v \circ g).$$

On étend l'action "*" à $A < Z >$ comme suit :

$$\forall P \in A < Z >, \quad P * f = \sum_{w \in \mathrm{supp}(P)} < P|w > w * f,$$

cette somme étant finie puisque le support de P est fini.

Une implantation de l'action à gauche de $A < Z >$ sur $\mathbb{K}^{conv}[\![\Theta]\!]$ en MACSYMA est donnée au chapitre VI.

Soit S une série formelle de $A \ll Z \gg$. Soit f une série formelle de $\mathbb{K}^{conv}[\![\Theta]\!]$. Soit q un point de $Conv(f)$. On considère la somme infinie suivante :

$$\sum_{w \in Z^*} < S|w > (w * f)_{|q}.$$

Si cette somme converge normalement, alors on définit l'action de la série S sur f par :

$$(S * f)_{|q} = \sum_{w \in Z^*} < S|w > (w * f)_{|q},$$

et on note $A^c \ll Z \gg$ l'ensemble des séries formelles S de $A \ll Z \gg$ telles que pour toute série f de $\mathbb{K}^{conv}[\![\Theta]\!]$, la somme précédente soit normalement convergente.

1.8.2. Annulateurs ([HJ5])

Soit f une série de $\mathbb{K}^{conv}[\![\Theta]\!]$. On appelle **annulateur** de f, l'ensemble des polynômes de $A < Z >$ dont l'action sur f est nulle :

$$Ann_Z f = \{P \in A < Z > \mid \quad P * f \equiv 0\},$$

On définit aussi :

$$Ann_Z^L f = Ann_Z f \bigcap Lie < Z > .$$

Lorsqu'il n'y a pas d'ambiguïté sur l'alphabet Z, on note $Ann f$ (resp. $Ann^L f$) au lieu de $Ann_Z f$ (resp. $Ann_Z^L f$).

Soit F une partie de $\mathbb{K}^{conv}[\![\Theta]\!]$. On note :

$$AnnF = \bigcap_{f \in F} Ann f,$$

$$Ann^L F = \bigcap_{f \in F} Ann^L f,$$

on a alors :

$$Ann^L F = AnnF \bigcap Lie < Z > .$$

Soit Y un sous alphabet de Z, et soit F une partie de $\mathbb{K}^{conv}[\![\Theta]\!]$, nous avons les résultats généraux suivants :

Lemme 1.8.2.1. :

$$[Y, AnnF] \subset AnnF \iff AnnF \subset Ann(Y * F)$$

<u>*Preuve*</u> : Car :

$$[Y, AnnF] * F = AnnF * (Y * F) \quad \bullet$$

Corollaire 1.8.2.1. : *Pour tout entier positif* n, *les propriétés suivantes sont équivalentes :*

$$(a)\ [Y, Ann(Y^n * F)] \subset Ann(Y^n * F).$$
$$(b)\ Ann(Y^n * F) \subset Ann(Y^{n+1} * F).$$

Lemme 1.8.2.2. : *Si :*

$$AnnF \subset Ann(Y * F)$$

alors pour tout entier positif n, *on a :*

$$Ann(Y^n * F) \subset Ann(Y^{n+1} * F).$$

Preuve : Montrons par récurrence sur n :
. Le résultat est immédiat pour $n = 0$.
. Supposons que $\forall \nu \in [0, n],\ Ann(Y^\nu * F) \subset Ann(Y^{\nu+1} * F)$.
. Pour $\nu = n + 1$, soit :

$$P \in Ann(Y^{n+1} * F),$$
$$\iff PY \in Ann(Y^n * F),$$

d'après l'hypothèse de récurrence on déduit que :

$$PY \in Ann(Y^{n+1} * F),$$
$$\iff P \in Ann(Y^{n+2} * F),$$

par conséquent $Ann(Y^{n+1} * F) \subset Ann(Y^{n+2} * F)$, on a donc le résultat •

Corollaire 1.8.2.2. : *Si*

$$[Y, AnnF] \subset AnnF,$$

on a alors :

$$\bigcap_{n \geq 0} Ann(Y^n * F) = Ann(Y^* * F) = AnnF.$$

Preuve : Si $AnnF$ est stable par crochet de Lie avec Y alors d'après le lemme 1.8.2.1., on a :

$$AnnF \subset Ann(Y * F),$$

et d'après le lemme 1.8.2.2. on a la suite emboîtée :

$$AnnF \subset Ann(Y * F) \subset \ldots \subset Ann(Y^n * F) \subset \ldots$$

d'où le résultat •

Lemme 1.8.2.3. : *Soit la suite emboîtée :*

$$AnnF \subset Ann(Y * F) \subset \ldots \subset Ann(Y^n * F) \subset \ldots$$

S'il existe un entier positif l_0, tel que :

$$Ann(Y^{l_0} * F) = Ann(Y^{l_0+1} * F)$$

alors la suite $(Ann(Y^n * F))_{n\geq 0}$ est stationnaire à partir du rang l_0.

<u>*Preuve*</u> : Il suffit de montrer, pour un entier positif l_0 l'implication :

$$Ann(Y^{l_0} * F) = Ann(Y^{l_0+1} * F) \quad \Rightarrow \quad Ann(Y^{l_0+1} * F) = Ann(Y^{l_0+2} * F).$$

En effet, on a :

$$\begin{aligned} & P \ \in Ann(Y^{l_0+2} * F), \\ \Longleftrightarrow\ & PY \in Ann(Y^{l_0+1} * F), \\ \Longleftrightarrow\ & PY \in Ann(Y^{l_0} * F), \\ \Longleftrightarrow\ & P \ \in Ann(Y^{l_0+1} * F) \quad \bullet \end{aligned}$$

Lemme 1.8.2.4. : *Soit J une partie de Lie $< Z >$. Si $J \subset AnnF$ et J est stable par crochet de Lie avec Y alors :*

$$JY^* \subset AnnF.$$

<u>*Preuve*</u> : Il suffit de montrer (par récurrence) que :

$$\forall n \geq 0, \quad JY^n \subset AnnF.$$

. Le résultat est immédiat pour $n = 0$.
. Supposons que pour tout entier ν, $\nu \in [0..n[$, $JY^\nu \subset AnnF$.
. Pour $\nu = n$, on a :

$$JY^n = JYY^{n-1} = YJY^{n-1} - [Y, J]Y^{n-1}.$$

On a d'après l'hypothèse :

$$[Y, J] \subset J,$$

d'où d'après l'hypothèse de récurrence :

$$YJY^{n-1} \subset AnnF$$

et :

$$[Y, J]Y^{n-1} \subset AnnF.$$

Par conséquent :

$$JY^* \subset AnnF \quad \bullet$$

1.9. Rappels sur les variétés analytiques

1.9.1. Variétés analytiques ([G1], [I])

Une *variété topologique* de dimension N est un espace topologique séparé dont chaque point admet un voisinage ouvert homéomorphe à un ouvert de $\mathbb{R}^N$. On appelle *carte locale* sur une variété Q de dimension N la donnée (U, ϕ) d'un ouvert U de Q et d'un homéomorphisme ϕ de U dans $\mathbb{R}^N$. Un *atlas analytique* sur Q (variété de dimension N) est la donnée d'une famille de cartes locales sur Q, $(U_i, \phi_i)_{i\in I}$, vérifiant :

(a) Q est la réunion des ouverts U_i.

(b) si deux cartes locales (U, ϕ) et (V, ψ) vérifient $W = U \cap V \neq \emptyset$ alors l'application de $\phi(W)$ dans $\psi(W)$ induite par l'homéomorphisme composé $\psi \circ \phi^{-1}$ analytique sur W, ainsi que son inverse.

Enfin, on appelle *variété analytique* (abstraite) de dimension N la donnée d'une variété topologique Q définie par une famille de cartes locales qui est un atlas analytique.

Soit p un point d'une variété analytique Q de dimension N. On note $C^\omega(p)$ l'ensemble des fonctions f définies sur l'ouvert U d'une carte locale (U, ϕ) en p qui sont analytiques, c'est à dire telles que l'application composée $f \circ \phi^{-1} : \phi(U) \longrightarrow \mathbb{R}$ soit une application analytique (en effet, $\phi(U) \subset \mathbb{R}^N$). On montre alors que pour toute fonction analytique $f \in C^\omega(p)$, pour tout point q de U, il existe une série formelle s de $\mathbb{K}^{conv}[\![\Theta]\!]$, telle que $f(q) = s_{|\phi(q)-\phi(p)}$. On vérifie aisément que $C^\omega(p)$ est un espace vectoriel.

Soit Q une variété analytique de dimension N. On appelle *vecteur tangent* à Q au point $p \in Q$ toute application linéaire Y de $C^\omega(p)$ vérifiant la "règle de Leibnitz" :

$$Y(fg) = f(p)Yg + g(p)Yf.$$

L'ensemble des vecteurs tangents en p à Q est *l'espace tangent à la variété Q au point* p. Il est clair que c'est un espace vectoriel. On le note T_pQ. L'espace vectoriel T_pQ est de dimension N et admet une "base naturelle" formée des N vecteurs $\left\{\left(\frac{\partial}{\partial\phi_k}\right)_p\right\}_{1\le k\le N}$ définis comme suit :

$$\forall f \in C^\omega(p), \quad \left(\frac{\partial}{\partial\phi_k}\right)_p(f) = \left(\frac{\partial(f\circ\phi)^{-1}}{\partial q_k}\right)_{q=\phi(p)}.$$

Un *champ de vecteur Y sur une variété Q* de dimension N est une application Y qui associe à tout point p de Q un vecteur tangent $Y(p)$ à Q au point p. Le champ de vecteur Y est dit analytique si tout point q appartenant à la carte locale (U, ϕ) en p, il existe N fonctions $Y^1, \dots, Y^N$ analytiques telles que la restriction de Y à U soit donnée par :

$$\forall q \in U, \quad Y(q) = \sum_{k=1}^{N} Y^k(q)\left(\frac{\partial}{\partial\phi_k}\right)_p.$$

Remarque : *Si la variété analytique Q de dimension N est donnée comme une sous-variété de $\mathbb{R}^n$, avec $N < n$, on dit alors que Q est plongée dans $\mathbb{R}^n$. En tout point de $\mathbb{R}^n$, l'espace tangent à $\mathbb{R}^n$ en $p \in Q$ est $\mathbb{R}^n$ tout entier, et l'on pourra donc aussi exprimer les vecteurs tangents à Q dans la base $\left\{\left(\frac{\partial}{\partial q_k}\right)\right\}_{1\leq k\leq n}$ de $T_p\mathbb{R}^n$. Mais alors $T_pQ \subset T_p\mathbb{R}^n$ (les dimensions respectives étant d'une part N, dimension de Q, et d'autre part n, dimension de $\mathbb{R}^n$).*

1.9.2. Etude dans $Lie < Z >$ des champs de vecteurs ([HJ5])

L'étude des champs de vecteurs sur Q peut être faite dans l'algèbre de Lie libre engendrée par Z. En effet, puisque le crochet de Lie de deux champs de vecteurs est encore un champ de vecteurs ([G1], [I]), alors toute application :

$$\begin{aligned} R : Z &\longrightarrow T_qQ \\ z &\longmapsto A_z, \end{aligned}$$

peut être prolongée naturellement en un homomorphisme d'algèbre de Lie :

$$\begin{aligned} R : Lie < Z > &\longrightarrow T_qQ \\ P &\longmapsto \sum_{w\in Z^*} < P|w > A_w. \end{aligned}$$

En effet, si P est un polynôme de Lie, l'opérateur différentiel $\sum_{w\in Z^*} < P|w > \ A_w$ est encore un champ de vecteurs.

Soit P un polynôme de $Lie < Z >$. On a :

$$\begin{aligned} R(P) \equiv 0 &\iff \forall f \in C^\omega(Q), \quad \sum_{w\in Z^*} < P|w > A_w \circ f \equiv 0 \\ &\iff \forall f \in C^\omega(Q), \quad \sum_{w\in Z^*} < P|w > w * f \equiv 0 \\ &\iff \forall f \in C^\omega(Q), \quad P * f \equiv 0 \\ &\iff P \in Ann^L C^\omega(Q), \end{aligned}$$

soit encore $KerR = Ann^L C^\omega(Q)$. On a alors la suite exacte :

$$0 \longrightarrow Ann^L C^\omega(Q) \longrightarrow Lie < Z > \xrightarrow{R} Im(R) \subset T_qQ,$$

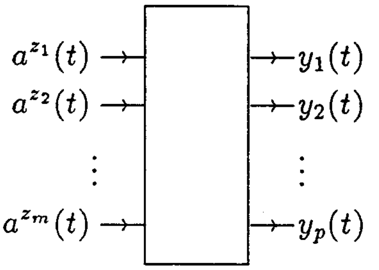
d'où $Lie < Z > /Ann^L C^\omega(Q)$ est isomorphe à $Im(R)$ qui est un sous-espace vectoriel de T_qQ, c'est donc un espace vectoriel de dimension au plus égale à la dimension de la variété Q, c'est-à-dire N.

CHAPITRE 2

TRANSFORMATION D'EVALUATION

2.0. Introduction

Tout système dynamique peut être considéré comme une “boîte noire” :



Nous choisissons de représenter chaque sortie $y_s(t)$, $s \in [1..p]$, d'un tel système par un signal paramétré, dépendant du temps t et des primitives des entrées a^z, $z \in Z = \{z_0, z_1, \ldots, z_m\}$:

$$y_s(t) = g_s(t, \int_0^t a^{z_1}(\tau)d\tau, \ldots, \int_0^t a^{z_m}(\tau)d\tau).$$

Si on note a^{z_0} l'application constante, et égale à 1, les signaux $\{g_s\}_{s\in[1..p]}$ sont alors les fonctionnelles qui dépendent les fonctions $\xi_z(t) = \displaystyle\int_0^t a^z(\tau)d\tau$, $z \in Z$.

M. Fliess, en commençant l'introduction de son article [F5], a fait la remarque suivante :

Les fonctionnelles, c'est-à-dire les fonctions de fonctions, ont, dès leur origine, été considérées comme des fonctions d'une infinité de “variables ordinaires”. Ce point de vue, certes indispensable, ne facilite pas toujours des choses : comment calculer avec une infinité de variables ?

Fortement influencé par les techniques de M.P. Schützenberger qui consistent à coder un grand nombre d'informations par un petit nombre de symboles (ou lettres) d'un alphabet fini, il introduit $m+1$ indéterminées non commutatives $\{z_0, z_1, \ldots, z_m\}$ pour coder des fonctionnelles causales analytiques (fonctions de $m+1$ fonctions dépendant du temps) par leurs séries génératrices (ou encore séries de Fliess). Ce sont des séries formelles en les indéterminées non commutatives $z \in Z$. De la même manière, les signaux $\{g_s\}_{s\in[1..p]}$ peuvent être symboliquement décrits par leurs séries génératrices

$$S_s = \sum_{w\in Z^*} < S_s|w > w.$$

Maintenant, étant donné une série formelle S en indéterminées non commutatives, à quelle condition et comment peut-on calculer (par ordinateur de surcroît) entièrement son comportement temporel ? Lorsque la série S est finie (c'est-à-dire que S est un polynôme), il suffit de remplaçer chaque mot w dans S par son intégrale itérée $\int_0^t \delta_a w$. Si la série S est infinie, on peut toujours la tronquer et se ramener aussi au cas polynômial. Cependant pourquoi tronquer la série $\left(\sum_{z\in Z} \alpha_z z\right)^*$ (les α_z sont des nombres complexes), par exemple, sachant que le signal associé est la fonction $\exp\left(\sum_{z\in Z} \alpha\xi_z(t)\right)$?

D'autre part, il est montré par M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue ([FLL], [L1], [L2]) que la série génératrice de nombreux systèmes dynamiques (provenant de la modélisation des circuits électroniques non linéaires par exemple) peut être approximée par une combinaison linéaire de fractions rationnelles non commutatives de la forme :

$$(c_0 z_0)^{*p_0} z_{i_1} (c_1 z_0)^{*p_1} z_{i_2} (c_2 z_0)^{*p_2} \dots z_{i_{k-1}} (c_{k-1} z_0)^{*p_{k-1}} z_{i_k} (c_k z_0)^{*p_k},$$

où les $p_0, \dots, p_k$ sont des entiers positifs , les $c_0, \dots, c_k$ sont des nombres complexes, les $z_{i_0}, \dots, z_{i_k}$ sont des lettres de Z (c'est une approximation du type Volterra). Ces approximations peuvent être obtenues par les méthodes proposées dans ([FLL], [L1], [L2]) pour calculer itérativement une solution approchée des équations différentielles non linéaires en régime forcé du type :

$$y^{(n)}(t) + \alpha_1 y^{(n-1)}(t) + \dots + \alpha_n y(t) + \sum_{i=2}^{d} p_i y^i(t) = u(t).$$

On peut aussi les obtenir par les méthodes proposées par G. Viennot et P. Leroux en utilisant des "arbres croissants", des "chemins de Motzkin" et des "histoires" ([VL]), et par C. Hespel et G. Jacob en utilisant les automates structurels ([J2], [HJ1], [HJ2], [HJ3]).

Le calcul de la sortie peut être mené en restant dans le cadre des transformées de Laplace-Borel : on code chaque entrée par sa transformée de Laplace-Borel, et on calcule la transformée de Laplace-Borel de la sortie. Comme l'ont montré M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue ([FLL], [L1], [L2]) la sortie de tels systèmes peut être calculée par le procédé suivant : calculer la transformée de Laplace-Borel g_{a^z} de l'entrée a^z, puis, en "lisant" la série de gauche à droite, remplacer chaque lettre $z \neq z_0$ par l'opérateur $[.\sqcup\!\sqcup\, g_{a^z}]z_0$. Ainsi, ils obtiennent la transformée de Laplace-Borel du signal de sortie qui est une série sur une seule lettre z_0. En utilisant la transformation de Laplace-Borel inverse ils trouvent le comportement temporel associé à cette série. Ce procédé a été effectivement implanté par ses auteurs en LISP dans ([FLL], [L1], [L2]). Cette technique appelle les remarques suivantes : elle est coûteuse en temps machine puisqu'elle utilise le produit de mélange (ignoré en général par les machines symboliques actuellement sur le marché), en outre, elle fait perdre tout le caractère du paramétrage par les entrées, que les auteurs supposent développables en séries entières de la forme $\sum_{n\geq 0} c_n \frac{t^n}{n!}$ (dont la transformée de Laplace-Borel sera une série en z_0 : $\sum_{n\geq 0} c_n z_0^n$).

Nous proposons dans ce chapitre, un procédé plus systématique pour calculer entièrement le comportement temporel associé à une série génératrice en évitant le calcul terme à terme quand cela est possible. Notre résultat le plus général est le calcul de l'Evaluation des séries de la forme :

$$G_0 z_{i_1} G_1 z_{i_2} G_2 \ldots z_{i_{k-1}} G_{k-1} z_{i_k} G_k,$$

où les séries $G_0, \ldots, G_k$ sont échangeables et $z_{i_0}, \ldots, z_{i_k}$ sont des lettres de Z. En particulier, pour calculer le comportement temporel par programmes écrits en MACSYMA (chapitre VI), nous considérons la classe des séries qui sont des combinaisons linéaires de fractions rationnelles non commutatives de la forme

$$(c_0 z_{j_0})^{*p_0} z_{i_1} (c_1 z_{j_1})^{*p_1} z_{i_2} (c_2 z_{j_2})^{*p_2} \ldots z_{i_{k-1}} (c_{k-1} z_{j_{k-1}})^{*p_{k-1}} z_{i_k} (c_k z_{j_k})^{*p_k},$$

où les $p_0, \ldots, p_k$ sont des entiers positifs, les $z_{j_0}, \ldots, z_{j_k}$ et $z_{i_0}, \ldots, z_{i_k}$ sont des lettres de Z. Dans le cas où les lettres $z_{j_0}, \ldots, z_{j_k}$ sont égales à z_0, nous retrouvons des approximants du type Volterra traités par la technique de M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue dans ([FLL], [L1], [L2]).

Nous développons, dans le premier temps, les bases pour notre "transformation d'Evaluation" avec noyau. Ce noyau peut être considéré comme la "mémoire temporelle" aussi bien au sens de Volterra qu'au sens de la programmation (chapitre VI). La transformation d'Evaluation est alors une fonctionnelle dépendant du noyau et des entrées. Nous démontrons une correspondance entre certains produits de convolution de signaux et des produits de Cauchy des séries de Fliess. Nous donnons ensuite des applications de ce théorème.

Ainsi, si les séries génératrices de M. Fliess sont considérées comme un codage symbolique du comportement entrée-sortie des systèmes dynamiques non linéaires, et si la transformation de Laplace-Borel inverse permet de retrouver le comportement temporel à partir des fonctions de transfert (séries formelles sur une seule lettre) dans le cadre des systèmes linéaires, de la même manière, la transformation d'Evaluation que nous allons définir permet, en retour, de déduire très simplement le comportement temporel de ces systèmes à partir de cette description symbolique. En outre, elle se traduit d'une manière naturelle et concise en écriture de programmation (voir chapitre VI).

2.1. Entrées relatives à un alphabet

Soit $Z = \{z_0, z_1, \ldots, z_m\}$ un alphabet fini.

Soit $[\alpha, \beta]$ un intervalle fini. Nous considérons les fonctions d'entrée $\{a^z\}_{z\in Z}$ qui sont dans $L^\infty(\alpha,\beta)$. Alors, pour toute fonction f dans $L^1(\alpha,\beta)$, nous avons l'inégalité de Hölder suivante :

$$\|fa^z\|_{L^1(\alpha,\beta)} \leq \|f\|_{L^1(\alpha,\beta)} \|a^z\|_{L^\infty(\alpha,\beta)}.$$

Définition 2.1.1. : *Nous appelerons* **entrée relative à l'alphabet fini** *Z la donnée du vecteur $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ de fonctions de $L^\infty(0,t)$ avec a^{z_0} désignant l'application constante, et égale à 1.*

Suivant K.T. Chen ([C1]), nous appelons **chemin associé à l'entrée** *$a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, le vecteur $\xi = (\, \xi_{z_0} \quad \xi_{z_1} \quad \ldots \quad \xi_{z_m}\,)$ défini comme suit :*

$$\forall z \in Z, \quad \xi_z(\tau) = \int_0^\tau d\xi_z(\rho) = \int_0^\tau a^z(\rho)d\rho.$$

Ainsi nous avons $\xi_{z_0}(\tau) = \displaystyle\int_0^\tau d\rho = \tau$, et pour toute lettre $z \in Z$, $\xi_z(0) = 0$.

Définition 2.1.2. : *Nous appelons* **morphisme majorant** *de l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$* **au temps** *$t$ tout morphisme $M_t : Z^* \longrightarrow \mathbb{R}_+$ vérifiant :*

$$M_t(\epsilon) = 1 \quad et \quad \|a^z\|_{L^\infty(0,t)} \leq M_t(z).$$

Dans la suite, nous supposons que f **est une fonction de** $L^1(0,t)$ et que f **s'annule en zéro.** La fonction unité (s'annulant en zéro) est notée par "*un*" :

$$\forall \tau \in \,]0,t], \quad un(\tau) = 1, \quad un(0) = 0.$$

Plus précisément, la fonction d'Evaluation est définie comme suit :

2.2. Evaluation des mots

Définition 2.2.1. : *L'***Evaluation d'un mot** *w* **de** *Z^** **par rapport au noyau** *f, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, est définie par récurrence sur la longueur de w comme suit :*

$$\mathcal{E}_a(f;w)(t) = \begin{cases} f(t) & si \quad w = \epsilon, \\ \displaystyle\int_0^t \mathcal{E}_a(f;v)(\tau)\, d\xi_z(\tau) & si \quad w = vz. \end{cases}$$

En particulier, lorsque f est la fonction unité un, nous retrouvons l'intégrale itérée $\int_0^t \delta_a w$ associée au mot w que nous notons aussi $\mathcal{E}_a(w)(t)$.

Lemme 2.2.1. : *Soit* $a = (\,a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ *une entrée et* M_t *un morphisme majorant de cette entrée (définition 2.1.2.). Nous avons la majoration suivante :*

$$\|\mathcal{E}_a(f;w)\|_{L^1(0,t)} \leq \|f\|_{L^1(0,t)}\, M_t(w)\frac{t^{|w|}}{|w|!}.$$

<u>*Preuve*</u> :

. Le résultat est évident pour $w = \epsilon$ ($|w| = 0$).

. Supposons que le résultat soit vrai pour tout mot w de longueur inférieure ou égale à n.

. Soit w un mot de Z^{n+1}, $w = w_1 z$. Nous avons :

$$\begin{aligned}
\|\mathcal{E}_a(f;w)\|_{L^1(0,t)} &= \int_0^t |\mathcal{E}_a(f;w)(\tau)|\, d\tau \\
&= \int_0^t \left| \int_0^\tau \mathcal{E}_a(f;w_1)(\rho) a^z(\rho) d\rho \right| d\tau \\
&\leq \int_0^t \int_0^\tau |\mathcal{E}_a(f;w_1)(\rho) a^z(\rho) d\rho|\, d\tau \\
&= \int_0^t \|\mathcal{E}_a(f;w_1) a^z\|_{L^1(0,\tau)}\, d\tau \\
&\leq \int_0^t \|\mathcal{E}_a(f;w_1)\|_{L^1(0,\tau)}\, \|a^z\|_{L^\infty(0,\tau)}\, d\tau \\
&\quad \text{(inégalité de Hölder)} \\
&\leq \int_0^t \|f\|_{L^1(0,\tau)}\, M_\tau(w_1)\frac{\tau^{|w_1|}}{|w_1|!} M_\tau(z) d\tau \\
&\quad \text{(d'après l'hypothèse de récurrence)} \\
&\leq \|f\|_{L^1(0,t)}\, M_t(w_1 z) \int_0^t \frac{\tau^{|w_1|}}{|w_1|!} d\tau \\
&= \|f\|_{L^1(0,t)}\, M_t(w)\frac{t^{|w|}}{|w|!}.
\end{aligned}$$

Ainsi, étant donnée une entrée $a = (\,a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ et étant donné un noyau f de $L^1(0,t)$, l'Evaluation du mot w est aussi un élément de $L^1(0,t)$. Par conséquent, on peut utiliser $\mathcal{E}_a(f;w)$ de nouveau comme un noyau comme l'exprime le lemme suivant :

Lemme 2.2.2. : *Soient u, v deux mots de Z^*. Alors :*

$$\mathcal{E}_a(f;uv) = \mathcal{E}_a(\mathcal{E}_a(f;u);v).$$

<u>*Preuve*</u> : Montrons ce lemme par récurrence sur la longueur du mot v :

. Le résultat est évident pour $|v| = 0$.

. Supposons que le résultat soit vrai pour tout mot v de longueur inférieure ou égale à n.

. Pour $v \in Z^{n+1}Z^*$, $v = wz$, nous avons :

$$\begin{aligned}
\mathcal{E}_a(\mathcal{E}_a(f;u);v)(t) &= \int_0^t \mathcal{E}_a(\mathcal{E}_a(f;u);w)(\tau)\ d\xi_z(\tau) \\
&= \int_0^t \mathcal{E}_a(f;uw)(\tau)\ d\xi_z(\tau) \\
&\qquad \text{(d'après l'hypothèse de récurrence)} \\
&= \mathcal{E}_a(f;uv)(t) \quad \bullet
\end{aligned}$$

Nous étendons par linéarité, la définition 2.2.1. à $I\!\!K < Z >$ de la manière suivante :

2.3. Evaluation des polynômes

Dans la suite de ce chapitre $I\!\!K$ désigne $I\!\!R$ ou $\mathbb{C}$.

Soit a une entrée relative à l'alphabet fini $Z = \{z_0, z_1, \ldots, z_m\}$.

Définition 2.3.1. : *Nous appelerons* **Evaluation du polynôme P de $I\!\!K < Z >$ par rapport au noyau f**, *pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, la fonction :*

$$\mathcal{E}_a(f;P) = \sum_{w \in \mathrm{supp}(P)} < P|w > \mathcal{E}_a(f;w).$$

*En particulier, pour $f = un$, l'***Evaluation du polynôme P de $I\!\!K < Z >$**, *pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, est la fonction ($[H2], [HJ3], [HJ4]$) :*

$$\mathcal{E}_a(P) = \mathcal{E}_a(un;P).$$

Théorème 2.3.1. : *Soient P et Q deux polynômes de $I\!K < Z >$. Alors :*

$$\mathcal{E}_a(f;P.Q) = \mathcal{E}_a(\mathcal{E}_a(f;P);Q).$$

Preuve : En effet, d'après la définition 2.3.1., nous avons :

$$\begin{aligned}
\mathcal{E}_a(\mathcal{E}_a(f;P);Q) &= \sum_{u\in\mathrm{supp}(P)}\ \sum_{v\in\mathrm{supp}(Q)} < P|u >< Q|v > \mathcal{E}_a(\mathcal{E}_a(f;u);v)\\
&= \sum_{u\in\mathrm{supp}(P)}\ \sum_{v\in\mathrm{supp}(Q)} < P|u >< Q|v > \mathcal{E}_a(f;uv) \quad \text{(lemme 2.2.2.)}\\
&= \mathcal{E}_a\Big(f;\sum_{u\in\mathrm{supp}(P)}\ \sum_{v\in\mathrm{supp}(Q)} < P|u >< Q|v > uv\Big)\\
&= \mathcal{E}_a(f;P.Q) \quad \bullet
\end{aligned}$$

Théorème 2.3.2. : *Soient P et Q deux polynômes de $I\!K < Z >$. Soit r un scalaire. Alors :*

$$\mathcal{E}_a(f;P+rQ) = \mathcal{E}_a(f;P) + r\mathcal{E}_a(f;Q).$$

Preuve : La preuve est immédiate d'après la linéarité des intégrales ordinaires $\bullet$

Lemme 2.3.1. : *Soient u et v deux mots de Z^*. Alors l'Evaluation du polynôme $u \sqcup\!\sqcup v$, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, est donnée par :*

$$\mathcal{E}_a(u \sqcup\!\sqcup v) = \mathcal{E}_a(u)\mathcal{E}_a(v).$$

Preuve :

. Le résultat est immédiat pour $|u| = 0$ ou $|v| = 0$ car $u \sqcup\!\sqcup \epsilon = u$, $\epsilon \sqcup\!\sqcup v = v$ et $\mathcal{E}_a(\epsilon) = 1$.

. Supposons que le résultat soit vrai pour tous mots u, v tels que la longueur de $|uv|$ soit inférieure ou égale à n.

. Si $|uv| = n+1$, alors nous pouvons écrire $u = u_1x$ et $v = v_1y$ avec $x,\ y \in Z$, $u_1,\ v_1 \in Z^*$ et $|u_1v| = |uv_1| = n$. En utilisant le fait que

$u ш v = (u ш v_1)y + (u_1 ш v)x$, nous avons :

$$\begin{aligned}
\mathcal{E}_a(u ш v)(t) &= \mathcal{E}_a((u ш v_1)y)(t) + \mathcal{E}_a((u_1 ш v)x)(t) \\
&= \int_0^t \mathcal{E}_a(u ш v_1)(\tau)\ d\xi_y(\tau) + \int_0^t \mathcal{E}_a(u_1 ш v)(\tau)\ d\xi_x(\tau) \\
&= \int_0^t \Big[\mathcal{E}_a(u)(\tau)\mathcal{E}_a(v_1)(\tau)\Big] d\xi_y(\tau) + \int_0^t \Big[\mathcal{E}_a(u_1)(\tau)\mathcal{E}_a(v)(\tau)\Big] d\xi_x(\tau) \\
&\quad \text{(d'après l'hypothèse de récurrence)} \\
&= \int_0^t \mathcal{E}_a(u)(\tau) d[\mathcal{E}_a(v_1 y)(\tau)] + \int_0^t \mathcal{E}_a(v)(\tau) d[\mathcal{E}_a(u_1 x)(\tau)] \\
&= \Big[\mathcal{E}_a(u_1 x)(\tau)\mathcal{E}_a(v_1 y)(\tau)\Big]_0^t \quad \text{(intégration par parties)} \\
&= \mathcal{E}_a(u)(t)\mathcal{E}_a(v)(t) \quad \bullet
\end{aligned}$$

Nous verrons dans le corollaire suivant, que cette transformation d'Evaluation est aussi une "extension en variables non commutatives" de la transformation exponentielle de la "série génératrice ordinaire" $S = \sum_{n \geq 0} c_n z^n$ en la "série génératrice exponentielle" associée $\mathcal{E}_a(S) = \sum_{n \geq 0} c_n \frac{\xi_z^n}{n!}$ bien connue en "combinatoire" ([F2], [F8]).

Corollaire 2.3.1. : *Soit z une lettre de Z. Alors pour tout entier positif n, nous avons :*

$$\mathcal{E}_a(z^n)(t) = \frac{\xi_z^n(t)}{n!}.$$

En particulier, si $z = z_0$ alors pour tout entier positif n, nous avons :

$$\mathcal{E}_a(z_0^n)(t) = \frac{t^n}{n!}.$$

Preuve :

. Le résultat est évident pour $n = 0$.

. Supposons que le résultat soit vrai pour tout ν, $0 \leq \nu \leq n$.

. Pour $\nu = n + 1$, puisque $z^n ш z = (n+1)z^{n+1}$ (lemme 1.4.3.1.), et d'après le lemme 2.3.1., nous avons :

$$\mathcal{E}_a(z^{n+1}) = \frac{1}{n+1}\mathcal{E}_a(z^n ш z) = \frac{1}{n+1}\mathcal{E}_a(z^n)\mathcal{E}_a(z) = \frac{1}{n+1}\frac{\xi_z^n}{n!}\xi_z = \frac{\xi_z^{n+1}}{(n+1)!}.$$

En particulier, pour $z = z_0$, puisque $\xi_{z_0}(t) = t$, alors nous avons le résultat annoncé $\bullet$

Remarque : *Si $f \neq un$ alors $\mathcal{E}_a(f; u \shuffle v) \neq \mathcal{E}_a(f;u)\mathcal{E}_a(f;v)$ en général car il suffit de remarquer que $w(u \shuffle v) \neq (wu) \shuffle (wv)$ en général.*

Lemme 2.3.2. : *Soient x et y deux lettres de Z. Soient i et j deux entiers positifs. L'Evaluation du polynôme $x^i \shuffle y^j$ par rapport au noyau f est*

$$\mathcal{E}_a(f; x^i \shuffle y^j) = \int_0^t \frac{(\xi_x(t) - \xi_x(\tau))^i}{i!} \frac{(\xi_y(t) - \xi_y(\tau))^j}{j!} df\ (\tau).$$

<u>*Preuve*</u> :

. Le résultat est immédiat pour $i = 0$ ou $j = 0$.

. Supposons que le résultat soit vrai pour tous entiers positifs i_1 et j_1 tel que $i_1 + j_1 < i + j$.

. Si $i_1 = i$ et $j_1 = j$, alors nous pouvons utiliser le fait que

$$x^i \shuffle y^j = x(x^{i-1} \shuffle y^j) + y(x^i \shuffle y^{i-1}),$$

dans ce cas, nous avons :

$$\begin{aligned}
\mathcal{E}_a(f; x^i \shuffle y^j) &= \mathcal{E}_a(f; x(x^{i-1} \shuffle y^j)) + \mathcal{E}_a(f; y(x^i \shuffle y^{i-1})) \\
&\quad \text{(d'après le théorème 2.3.2.)} \\
&= \mathcal{E}_a(\mathcal{E}_a(f;x); x^{i-1} \shuffle y^j) + \mathcal{E}_a(\mathcal{E}_a(f;y); x^i \shuffle y^{i-1}) \\
&\quad \text{(d'après le théorème 2.3.1.)} \\
&= \int_0^t \frac{(\xi_x(t) - \xi_x(\tau))^{i-1}}{(i-1)!} \frac{(\xi_y(t) - \xi_y(\tau))^j}{j!} d\mathcal{E}_a(f;x)(\tau) \\
&\quad + \int_0^t \frac{(\xi_x(t) - \xi_x(\tau))^i}{i!} \frac{(\xi_y(t) - \xi_y(\tau))^{j-1}}{(j-1)!} d\mathcal{E}_a(f;y)(\tau) \\
&\quad \text{(d'après l'hypothèse de récurrence)} \\
&= \int_0^t \frac{(\xi_x(t) - \xi_x(\tau))^{i-1}}{(i-1)!} \frac{(\xi_y(t) - \xi_y(\tau))^j}{j!} f(\tau) d\xi_x(\tau) \\
&\quad + \int_0^t \frac{(\xi_x(t) - \xi_x(\tau))^i}{i!} \frac{(\xi_y(t) - \xi_y(\tau))^{j-1}}{(j-1)!} f(\tau) d\xi_y(\tau) \\
&= \int_0^t f(\tau)\ d\left[\frac{(\xi_x(t) - \xi_x(\tau))^i}{i!} \frac{(\xi_y(t) - \xi_y(\tau))^j}{j!}\right] \\
&\quad \text{(dérivée d'un produit)} \\
&= \left[f(\tau) \frac{(\xi_x(t) - \xi_x(\tau))^i}{i!} \frac{(\xi_y(t) - \xi_y(\tau))^j}{j!}\right]_0^t \\
&\quad + \int_0^t \frac{(\xi_x(t) - \xi_x(\tau))^i}{i!} \frac{(\xi_y(t) - \xi_y(\tau))^j}{j!} df(\tau) \\
&\quad \text{(intégration par parties)} \\
&= \int_0^t \frac{(\xi_x(t) - \xi_x(\tau))^i}{i!} \frac{(\xi_y(t) - \xi_y(\tau))^j}{j!} df(\tau) \qquad (f(0) = 0) \quad \bullet
\end{aligned}$$

Plus généralement, nous avons la proposition suivante :

Proposition 2.3.1. : (Evaluation des polynômes échangeables) *Pour toutes lettres* $x_1, \ldots, x_k$ *de* Z *et pour tous entiers* $\alpha_1, \ldots, \alpha_k$, *l'Evaluation du polynôme* $x_1^{\alpha_1} ш \ldots ш x_k^{\alpha_k}$ *est :*

$$\mathcal{E}_a(f; x_1^{\alpha_1} ш \ldots ш x_k^{\alpha_k}) = \int_0^t \frac{(\xi_{x_1}(t) - \xi_{x_1}(\tau))^{\alpha_1}}{\alpha_1!} \ldots \frac{(\xi_{x_k}(t) - \xi_{x_k}(\tau))^{\alpha_k}}{\alpha_k!} df \quad (\tau).$$

<u>*Preuve*</u> : D'après 1.4.3., nous avons :

$$x_1^{\alpha_1} ш \ldots ш x_k^{\alpha_k} = \sum_{i=1}^{k} x_i (x_1^{\alpha_1} ш \ldots ш x_{i-1}^{\alpha_{i-1}} ш x_i^{\alpha_i - 1} ш x_{i+1}^{\alpha_{i+1}} ш \ldots ш x_k^{\alpha_k}),$$

d'après le théorème 2.3.2. et avec la même démonstration que précédemment, nous avons le résultat anoncé •

Nous étendons la définition 2.3.1. à $\mathbb{K} \ll Z \gg$ (sous certaines conditions de convergence) de la manière suivante :

2.4. Evaluation des séries formelles

Dans la suite de ce chapitre $\mathbb{K}$ désigne $\mathbb{R}$ ou $\mathbb{C}$.
Soit a une entrée relative à l'alphabet fini $Z = \{z_0, z_1, \ldots, z_m\}$.

Définition 2.4.1. : *Nous appellerons* **Evaluation de la série formelle** S **de** $\mathbb{K} \ll Z \gg$ **par rapport au noyau** f, *pour l'entrée* $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, *quand elle est définie, la fonction*

$$\mathcal{E}_a(f; S) = \sum_{w \in Z^*} < S|w > \mathcal{E}_a(f; w).$$

En particulier, pour $f = un$, *l'***Evaluation de la série formelle** S **de** $\mathbb{K} \ll Z \gg$, *pour l'entrée* $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, *est la fonction* $([H2], [HJ3], [HJ4])$ *:*

$$\mathcal{E}_a(S) = \mathcal{E}_a(un; S).$$

Théorème 2.4.1. : *Soit* $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ *une entrée et* M_t *un morphisme majorant de cette entrée (définition 2.1.2.). Soit* χ *un morphisme de monoïde, de* Z^* *dans* $I\!R_+$, *vérifiant le condition suivante :*

$$\sum_{z \in Z} \chi(z) M_t(z) < \frac{1}{t}$$

Dans ces conditions, pour toute série formelle S *de* $I\!K^{\chi-cc} \ll Z \gg$ *(définition 1.6.1.2.), la série* $\sum_{w \in Z^*} < S|w > \mathcal{E}_a(f;w)(t)$ *est normalement convergente et en plus, nous avons :*

$$\mathcal{E}_a(f;S) \in L^1(0,t).$$

<u>*Preuve*</u> : Cela est immédiat d'après le lemme 2.2.1., nous prendrons $K_1 = \|f\|_{L^1(0,t)}$ et ϕ_t le morphisme de monoïde, de Z^* dans $I\!R_+$ défini comme suit :

$$\forall z \in Z, \quad \phi_t(z) = t M_t(z).$$

Dans ces conditions, nous avons :

$$\forall w \in Z^*, \quad \|\mathcal{E}_a(f;w)\|_{L^1(0,t)} \leq K_1 \phi_t(w).$$

Soit χ le morphisme de monoïde, de Z^* dans $I\!R_+$ vérifiant :

$$\sum_{z \in Z} \chi(z) M_t(z) < \frac{1}{t}.$$

En d'autres termes, nous avons :

$$\sum_{z \in Z} \chi(z) \phi_t(z) < 1.$$

D'après le lemme 1.6.1.3., la série $\sum_{w \in Z^*} < S|w > \mathcal{E}_a(f;w)(t)$ est donc normalement convergente •

D'après la formule fondamentale de M. Fliess, nous constatons que l'Evaluation d'une série formelle S peut être considérée comme une transformation qui associe à S (sous condition de convergence du théorème 2.4.1.) un signal dépendant des primitives $\{\xi_z\}_{z \in Z}$ des entrées $\{a^z\}_{z \in Z}$, et la transformation d'Evaluation n'est rien d'autre qu'une généralisation adéquate de l'inverse des transformations de Laplace et de Fourier ([H2], [HJ3], [HJ4]). Dans le cadre du calcul symbolique pour les systèmes linéaire, l'opérateur d'intégration est noté par "$\frac{1}{p}$". Ici, il coïncide avec la lettre z_0. Et chaque lettre z de l'alphabet de codage Z de M. Fliess joue un rôle analogue : il code "l'opérateur d'intégration de Stieltjes" par rapport à ξ_z.

Théorème 2.4.2. : *Soient S et T deux séries formelles de $\mathbb{K} \ll Z \gg$ vérifiant les hypothèses du théorème 2.4.1. Alors :*

$$\mathcal{E}_a(f; S.T) = \mathcal{E}_a(\mathcal{E}_a(f; S); T).$$

<u>*Preuve*</u> : En effet, d'après la définition 2.4.1., nous avons :

$$\begin{aligned}
\mathcal{E}_a(\mathcal{E}_a(f;S);T) &= \sum_{u \in Z^*} \sum_{v \in Z^*} < S|u >< T|v > \mathcal{E}_a(\mathcal{E}_a(f;u);v) \\
&= \sum_{u \in Z^*} \sum_{v \in Z^*} < S|u >< T|v > \mathcal{E}_a(f;uv) \quad \text{(lemme 2.2.2.)} \\
&= \mathcal{E}_a\Big(f; \sum_{u \in Z^*} \sum_{v \in Z^*} < S|u >< T|v > uv\Big) \\
&= \mathcal{E}_a(f; S.T) \quad \bullet
\end{aligned}$$

L'introduction du noyau f pour la fonction d'Evaluation $\mathcal{E}_a$ ([H2], [HJ3], [HJ4]) permet de donner une notion de mémoire pour les systèmes. Ce noyau peut être considéré comme la mémoire temporelle des systèmes au sens de Volterra aussi bien qu'au sens de la programmation (voir chapitre VI) justifiant ainsi notre approche. La transformation d'Evaluation devient alors une fonctionnelle dépendant du noyau f et des entrées $\{a^z\}_{z \in Z}$. Pour implanter ces fonctionnelles nous utilisons (chapitre VI) la "λ_notation" de MACSYMA ([M3]).

Théorème 2.4.3. : *Soient S et T deux séries formelles de $\mathbb{K} \ll Z \gg$ vérifiant les hypothèses du théorème 2.4.1. Soit r un scalaire. Alors :*

$$\mathcal{E}_a(f; S + rT) = \mathcal{E}_a(f; S) + r\mathcal{E}_a(f; T).$$

<u>*Preuve*</u> : La preuve est immédiate d'après la linéarité des intégrales ordinaires $\bullet$

Théorème 2.4.4. : *Soient S et T deux séries formelles vérifiant les conditions du théorème 2.4.1. Alors l'Evaluation du produit de mélange $S ш T$ est le produit des Evaluations de S et T :*

$$\mathcal{E}_a(S ш T) = \mathcal{E}_a(S)\mathcal{E}_a(T).$$

<u>*Preuve*</u> : D'après la définition de $S ш T$ et le lemme 2.3.1., nous avons :

$$\begin{aligned}
\mathcal{E}_a(S ш T) &= \sum_{u,v \in Z^*} < S|u >< T|v > \mathcal{E}_a(u ш v) \\
&= \sum_{u \in Z^*} \sum_{v \in Z^*} < S|u >< T|v > \mathcal{E}_a(u)\mathcal{E}_a(v) \\
&= \Big(\sum_{u \in Z^*} < S|u > \mathcal{E}_a(u)\Big)\Big(\sum_{v \in Z^*} < T|v > \mathcal{E}_a(v)\Big) \\
&= \mathcal{E}_a(S)\mathcal{E}_a(T) \quad \bullet
\end{aligned}$$

Théorème 2.4.5. : (théorème de Convolution) *Soit H une série échangeable :*

$$H = \sum_{\alpha_0,\ldots,\alpha_m \geq 0} h_{\alpha_0,\alpha_1,\ldots,\alpha_m} z_0^{\alpha_0} ⧢ \ldots ⧢ z_m^{\alpha_m},$$

vérifiant les hypothèses du théorème 2.4.1. L'Evaluation de la série H par rapport au noyau f, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, est :

$$\mathcal{E}_a(f;H)(t) = \int_0^t h(\xi(t) - \xi(\tau))df(\tau),$$

où $h(\xi(t))$ est l'Evaluation de la série H, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \ldots \quad a^{z_m}\,)$. $h(\xi(t))$ une fonction analytique des $\{\xi_z\}_{z\in Z}$:

$$h(\xi(t)) = h(t, \xi_{z_1}(t), \ldots, \xi_{z_m}(t)) = \sum_{\alpha_0,\ldots,\alpha_m \geq 0} h_{\alpha_0,\alpha_1,\ldots,\alpha_m} \frac{t^{\alpha_0}\xi_{z_1}^{\alpha_1}(t)\ldots\xi_{z_m}^{\alpha_m}(t)}{\alpha_0!\alpha_1!\ldots\alpha_m!}.$$

<u>*Preuve*</u> : D'après la proposition 2.3.1., on a :

$$\begin{aligned}
\mathcal{E}_a(f;H)(t) &= \sum_{\alpha_0,\ldots,\alpha_m \geq 0} h_{\alpha_0,\alpha_1,\ldots,\alpha_m} \mathcal{E}_a(f; z_0^{\alpha_0} ⧢ \ldots ⧢ z_m^{\alpha_m}) \\
&= \sum_{\alpha_0,\ldots,\alpha_m \geq 0} h_{\alpha_0,\alpha_1,\ldots,\alpha_m} \\
&\quad \int_0^t \frac{(t-\tau)^{\alpha_0}(\xi_{z_1}(t) - \xi_{z_1}(\tau))^{\alpha_1}\ldots(\xi_{z_m}(t) - \xi_{z_m}(\tau))^{\alpha_m}}{\alpha_0!\alpha_1!\ldots\alpha_m!} df(\tau) \\
&= \int_0^t \sum_{\alpha_0,\ldots,\alpha_m \geq 0} h_{\alpha_0,\alpha_1,\ldots,\alpha_m} \\
&\quad \frac{(t-\tau)^{\alpha_0}(\xi_{z_1}(t) - \xi_{z_1}(\tau))^{\alpha_1}\ldots(\xi_{z_m}(t) - \xi_{z_m}(\tau))^{\alpha_m}}{\alpha_0!\alpha_1!\ldots\alpha_m!} df(\tau) \\
&= \int_0^t h(\xi(t) - \xi(\tau))df(\tau) \quad \bullet
\end{aligned}$$

Les opérateurs d'intégration de Stieltjes coïncident donc avec ceux de la transformation de Laplace multidimensionnelle lorsqu'ils commutent entre eux, c'est-à-dire lorsque le noyau f est la fonction unité et la série formelle S est échangeable (théorème 2.4.5.).

Nous en déduisons une correspondance entre une certaine convolution de signaux et un produit de Cauchy de séries génératrices :

Corollaire 2.4.1. : *Soit G une série formelle de $\mathbb{K} \ll Z \gg$ et soit H une série échangeable vérifiant les hypothèses du théorème 2.4.1. L'Evaluation de la série formelle GH, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, est :*

$$\mathcal{E}_a(GH)(t) = < G|\epsilon > \ h(\xi(t)) + \int_0^t h(\xi(t) - \xi(\tau))d\mathcal{E}_a(G)(\tau),$$

où $h(\xi(t))$ est l'Evaluation de la série H, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \ldots \quad a^{z_m}\,)$.

<u>*Preuve*</u> : En effet, puisque $GH = < G|\epsilon > H + G_1 H$, où G_1 est une série formelle propre de $\mathbb{K} \ll Z \gg$, et d'après le théorème 2.4.3., l'Evaluation de la série formelle GH, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$, est :

$$\mathcal{E}_a(GH)(t) = < G|\epsilon > \mathcal{E}_a(H)(t) + \mathcal{E}_a(G_1 H)(t).$$

Comme H est une série échangeable, alors en utilisant le théorème de Convolution dans le cas particulier où le noyau f est l'Evaluation de la série formelle propre G_1, nous obtenons le résultat annoncé •

On peut observer qu'il existe une certaine dissymétrie dans cette convolution, cela est une conséquence immédiate de la non commutativité du produit de Cauchy des séries formelles G et H. Cette correspondance semble évidente mais elle est très pratique pour effectuer les calculs (voir 2.5.). En outre, elle nous conduit à une programmation très naturelle en MACSYMA (voir chapitre VI) pour l'Evaluation des combinaisons linéaires des fractions rationnelles non commutatives décrites par

$$(c_0 z_{j_0})^{*p_0} z_{i_1} (c_1 z_{j_1})^{*p_1} z_{i_2} (c_2 z_{j_2})^{*p_2} \ldots z_{i_{k-1}} (c_{k-1} z_{j_{k-1}})^{*p_{k-1}} z_{i_k} (c_k z_{j_k})^{*p_k},$$

où les $p_0, \ldots, p_k$ sont des entiers positifs , les $z_{j_0}, \ldots, z_{j_k}$ et $z_{i_0}, \ldots, z_{i_k}$ sont des lettres de Z. Dans le cas où les lettres $z_{j_0}, \ldots, z_{j_k}$ sont égales à z_0, nous retrouvons des approximants du type Volterra traités par la technique de M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue dans ([FLL], [L1], [L2]).

2.5. Exemple de calcul

Lemme 2.5.1. : *Pour tout entier $n \geq 1$, nous avons :*

$$\mathcal{E}_a(z^{*n})(t) = \exp(\xi_z(t)) g_n(\xi_z(t)),$$

où les g_n sont des polynômes en l'indéterminée $\xi_z(t)$, et vérifient l'équation récurrente suivante :

$$g_n(\xi_z(t)) = \begin{cases} 1 & si \quad n = 1 \\ g_{n-1}(\xi_z(t)) + \int_0^t g_{n-1}(\xi_z(t) - \xi_z(\tau))d\xi_z(\tau) & si \quad n > 1. \end{cases}$$

<u>*Preuve*</u> :

. Puisque $\mathcal{E}_a(z^*)(t) = \exp(\xi_z(t))$, on peut écrire $g_1(\xi_z(t)) = 1$.

. Supposons que le résultat soit vrai pour tout ν, $0 \leq \nu \leq n-1$.

. Pour $\nu = n$, nous avons $z^{*n} = z^* z^{*n-1}$. D'après l'hypothèse de récurrence et d'après le corollaire 2.4.1., nous obtenons le résultat annoncé :

$$\begin{aligned}
\mathcal{E}_a(z^{*n})(t) &= \exp(\xi_z(t))g_{n-1}(\xi_z(t)) \\
&\quad + \int_0^t \exp(\xi_z(t) - \xi_z(\tau))g_{n-1}(\xi_z(t) - \xi_z(\tau))d\exp(\xi_z(\tau)) \\
&= \exp(\xi_z(t))\left[g_{n-1}(\xi_z(t)) + \int_0^t g_{n-1}(\xi_z(t) - \xi_z(\tau))d\xi_z(\tau)\right] \quad \bullet
\end{aligned}$$

Lemme 2.5.2. : *La famille* $g_n(\xi_z(t)) = \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{\xi_z^j(t)}{j!}$, *pour* $n \geq 1$, *est l'unique solution de l'équation récurrente :*

$$g_n(\xi_z(t)) = \begin{cases} 1 & si \quad n = 1 \\ g_{n-1}(\xi_z(t)) + \int_0^t g_{n-1}(\xi_z(t) - \xi_z(\tau))d\xi_z(\tau) & si \quad n > 1. \end{cases}$$

<u>*Preuve*</u> : Etant donné $G_1 = 1 \in I\!K \ll Z \gg$, nous avons clairement :

$$g_1(\xi_z(t)) = \mathcal{E}_a(G_1)(t) = 1.$$

Supposons que pour tout entier $n \geq 1$, nous ayons :

$$g_n(\xi_z(t)) = \mathcal{E}_a(G_n)(t),$$

avec $G_n \in I\!K \ll Z \gg$. Alors nous avons pour tout entier $n \geq 1$ (voir le corollaire 2.4.1.) :

$$\mathcal{E}_a(G_n) = \mathcal{E}_a(G_{n-1}) + \mathcal{E}_a(zG_{n-1}).$$

Cette équation reste vraie si nous avons (théorème 2.4.3.) :

$$G_n = (1+z)G_{n-1}.$$

En d'autres termes :

$$G_n = (1+z)^{n-1} = \sum_{j=0}^{n-1} \binom{n-1}{j} z^j.$$

Puisque $\mathcal{E}_a(z^j)(t) = \frac{\xi_z^j(t)}{j!}$, $j \geq 0$ (corollaire 2.3.1.), alors nous avons pour tout entier $n \geq 1$:

$$g_n(\xi_z(t)) = \mathcal{E}_a(G_n)(t) = \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{\xi_z^j(t)}{j!} \quad \bullet$$

D'après le lemme 2.5.1. et le lemme 2.5.2., nous avons la proposition suivante :

Propositon 2.5.1. : *Pour tout entier positif n, nous avons :*

$$\mathcal{E}_a(z^{*n})(t) = \begin{cases} 1 & si \quad n = 0 \\ \exp(\xi_z(t)) \displaystyle\sum_{j=0}^{n-1} \binom{n-1}{j} \frac{\xi_z^j(t)}{j!} & si \quad n > 0. \end{cases}$$

En particulier, pour $z = z_0$, nous avons :

$$\mathcal{E}_a(z_0^{*n})(t) = \begin{cases} 1 & si \quad n = 0 \\ \exp(t) \displaystyle\sum_{j=0}^{n-1} \binom{n-1}{j} \frac{t^j}{j!} & si \quad n > 0. \end{cases}$$

De la même manière, nous avons la proposition suivante :

Proposition 2.5.2. : *Pour tout entier positif n, pour tout nombre complexe α, nous avons :*

$$\mathcal{E}_a[(\alpha z)^{*n}](t) = \begin{cases} 1 & si \quad n = 0 \\ \exp[\alpha\xi_z(t)] \displaystyle\sum_{j=0}^{n-1} \binom{n-1}{j} \frac{[\alpha\xi_z(t)]^j}{j!} & si \quad n > 0. \end{cases}$$

En particulier, pour $z = z_0$, nous avons :

$$\mathcal{E}_a[(\alpha z_0)^{*n}](t) = \begin{cases} 1 & si \quad n = 0 \\ \exp(\alpha t) \displaystyle\sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\alpha t)^j}{j!} & si \quad n > 0. \end{cases}$$

D'après le théorème de convolution et la proposition 2.5.2., nous en déduisons le théorème suivant :

Théorème 2.5.1. : *Pour tout entier positif n, pour tout nombre complexe α, nous avons :*

$$\mathcal{E}_a[f;(\alpha z)^{*n}](t) = \begin{cases} f(t) & si \quad n = 0 \\ \displaystyle\sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp[\alpha(\xi_z(t) - \xi_z(\tau))] \\ \qquad \dfrac{[\alpha(\xi_z(t) - \xi_z(\tau))]^j}{j!} df\ (\tau) & si \quad n > 0. \end{cases}$$

En particulier, pour $z = z_0$, nous avons :

$$\mathcal{E}_a[f;(\alpha z_0)^{*n}](t) = \begin{cases} f(t) \text{ si } n = 0 \\ \sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp[\alpha(t-\tau)] \frac{[\alpha(t-\tau)]^j}{j!} df(\tau) \text{ si } n > 0. \end{cases}$$

Théorème 2.5.2. : *Pour tout entier positif k, supposons que G_k est une série formelle échangeable et nous notons $g_k(\xi(t))$ son Evaluation :*

$$\mathcal{E}_a(G_k)(t) = g_k(\xi(t)) = g_k(t, \xi_1(t), \ldots, \xi_m(t)).$$

Pour tout entier positif k, nous posons :

$$S_k = G_0 z_{i_1} G_1 \ldots z_{i_k} G_k.$$

où z_{i_1}, $z_{i_2}, \ldots,$ z_{i_k} sont des lettres de Z. Avec ces notations, nous avons, pour tout entier $k > 0$:

$$\mathcal{E}_a(S_k)(t) = \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} g_0(\xi(\tau_1)) g_1(\xi(\tau_2) - \xi(\tau_1)) \ldots g_k(\xi(t) - \xi(\tau_k))$$
$$d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k).$$

<u>*Preuve*</u> :

. Pour $k = 1$, d'après le lemme 2.2.1., nous avons :

$$\mathcal{E}_a(G_0 z_{k_1})(t) = \int_0^t g_0(\xi(t)) a^{z_{k_1}}(\tau) d\tau,$$

et d'après le corollaire 2.4.1., nous avons :

$$\mathcal{E}_a(G_0 z_{k_1} G_1)(t) = \int_0^t g_0(\xi(\tau)) a^{z_{k_1}}(\tau) g_1(\xi(t) - \xi(\tau)) d\tau$$
$$= \int_0^t g_0(\xi(\tau)) g_1(\xi(t) - \xi(\tau)) d\xi_{z_{k_1}}(\tau).$$

. Le résultat est supposé vrai pour tout ν, $1 \leq \nu \leq k-1$:

$$\mathcal{E}_a(S_\nu)(t) = \int_0^t \int_0^{\tau_\nu} \ldots \int_0^{\tau_2} g_0(\xi(\tau_1)) g_1(\xi(\tau_2) - \xi(\tau_1)) \ldots$$
$$g_\nu(\xi_{z_{j_\nu}}(t) - \xi_{z_{j_\nu}}(\tau_\nu)) d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_\nu}}(\tau_\nu).$$

. Pour $\nu = k$, d'après le lemme 2.2.1., nous avons :

$$\mathcal{E}_a(S_{k-1} z_{i_k})(t) = \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} g_0(\xi(\tau_1)) g_1(\xi(\tau_2) - \xi(\tau_1)) \ldots$$
$$g_{k-1}(\xi_{z_{j_{k-1}}}(\tau_k) - \xi_{z_{j_{k-1}}}(\tau_{k-1})) d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k).$$

Finalement, d'après le corollaire 2.4.1., nous avons :

$$\mathcal{E}_a(S_k)(t) = \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} g_0(\xi(\tau_1)) g_1(\xi(\tau_2) - \xi(\tau_1)) \ldots g_k(\xi(t) - \xi(\tau_k))$$
$$d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k) \quad \bullet$$

Nous en déduisons le résultat suivant :

Corollaire 2.5.1. : *Soient $k+1$ entiers positifs p_0, p_1, ..., p_k. Soient z_{i_1}, $z_{i_2}\ldots$, z_{i_k}, les lettres de $Z_0 = Z \setminus \{z_0\}$. Alors l'Evaluation du mot $z_0^{p_0} z_{i_1} z_0^{p_1} \ldots z_{i_k} z_0^{p_k}$ est (voir le corollaire 2.2.1.) :*

$$\int_0^t \int_0^{\tau_k} \int_0^{\tau_{k-1}} \cdots \int_0^{\tau_2} \frac{\tau_1^{p_0}(\tau_2-\tau_1)^{p_1}\ldots(t-\tau_k)^{p_k}}{p_0!p_1!\ldots p_k!} d\xi_{z_{i_1}}(\tau_1)\ldots d\xi_{z_{i_k}}(\tau_k).$$

Ce résultat est utilisé dans [HJ4] pour donner le développement de Taylor des noyaux de Volterra. Il exprime que pour tout mot $z_0^{\alpha_0} z_{i_1} z_0^{\alpha_1} \ldots z_{i_k} z_0^{\alpha_k}$, nous pouvons associer une intégrale itérée

$$\int_0^t \int_0^{\tau_k} \cdots \int_0^{\tau_2} \frac{\tau_1^{\alpha_0}(\tau_2-\tau_1)^{\alpha_1}\ldots(t-\tau_k)^{\alpha_k}}{\alpha_0!\alpha_1!\ldots\alpha_k!} d\xi_{z_{i_1}}(\tau_1)\ldots d\xi_{z_{i_k}}(\tau_k)$$

dont on remarquera la diminution du nombre d'intégrations à effectuer. Cela généralise la correspondance classique entre le mot zz_0^α et l'intégrale $\int_0^t a^z(\theta)\frac{(t-\theta)^\alpha}{\alpha!}d\theta$. Cette dernière intégrale est la $\alpha^{i\grave{e}me}$ primitive de a^z, qui s'annule en 0 aussi bien que ses premières dérivées ([F5]).

D'après le théorème 2.5.1. et le théorème 2.5.2., nous en déduisons le résultat suivant :

Corollaire 2.5.2. : *Soit c_0, c_1, ..., c_k $k+1$ nombres complexes. Soient $k+1$ entiers positifs p_0, p_1, ..., p_k. Soient z_{i_1}, $z_{i_2}\ldots$, z_{i_k}, les lettres de $Z_0 = Z\setminus\{z_0\}$. Alors l'Evaluation de la fraction rationnelle non commutative $(c_0z_0)^{*p_0} z_{i_1}(c_1z_0)^{*p_1}\ldots z_{i_k}(c_kz_0)^{*p_k}$ est :*

$$\int_0^t \int_0^{\tau_k} \int_0^{\tau_{k-1}} \cdots \int_0^{\tau_2} f_0(\tau_1)f_1(\tau_2-\tau_1)\ldots f_k(t-\tau_k)d\xi_{z_{i_1}}(\tau_1)\ldots d\xi_{z_{i_k}}(\tau_k),$$

avec *(voir la proposition 2.5.2.) :*

$$\forall n \in [0..k], \quad f_n(t) = \begin{cases} 1 & si \quad p_n = 0 \\ \exp(c_nt)\displaystyle\sum_{j=0}^{p_n-1}\binom{p_n-1}{j}\frac{(c_nt)^j}{j!} & si \quad p_n > 0. \end{cases}$$

Ce résultat est aussi présenté dans ([FLL], [L1], [L2]) pour calculer itérativement les noyaux de Volterra de la solution des équations différentielles en régime forcé. Dans le chapitre VI, nous présenterons une implantation concise et efficace en MACSYMA du calcul de l'Evaluation de ces séries S_k.

2.6. Evaluation de quelque séries formelles usuelles

Soit z une lettre de Z. Nous avons les Evaluations de quelque séries formelles usuelles suivantes :

S	$\mathcal{E}_a(f;S)$
ϵ	$f(t)$
z^n	$\int_0^t \frac{(\xi_z(t)-\xi_z(\tau))^n}{n!} df(\tau)$
$z^{*n}, n \geq 1$	$\sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp(\xi_z(t)-\xi_z(\tau)) \frac{(\xi_z(t)-\xi_z(\tau))^j}{j!} df(\tau)$
$\sum_{n\geq 0} c_n z^n$	$\sum_{n\geq 0} c_n \int_0^t \frac{(\xi_z(t)-\xi_z(\tau))^n}{n!} df(\tau)$
alors, en particulier	
$z^* = \sum_{n\geq 0} z^n$	$\int_0^t \exp(\xi_z(t)-\xi_z(\tau)) df(\tau)$
$\sum_{n\geq 0} n z^n$	$\int_0^t (\xi_z(t)-\xi_z(\tau)) \exp(\xi_z(t)-\xi_z(\tau)) df(\tau)$
$\sum_{n\geq 0} (-1)^n z^{2n}$	$\int_0^t \cos(\xi_z(t)-\xi_z(\tau)) df(\tau)$
$\sum_{n\geq 0} (-1)^n z^{2n+1}$	$\int_0^t \sin(\xi_z(t)-\xi_z(\tau)) df(\tau)$

En particulier, pour $f = un$, nous avons les Evaluations suivantes ([H2], [HJ3], [HJ4]) :

S	$\mathcal{E}_a(S)$
ϵ	1
z^n	$\frac{\xi_z^n(t)}{n!}$
$z^{*n}, n \geq 1$	$\exp(\xi_z(t)) \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\xi_z(t))^j}{j!}$
$\sum_{n\geq 0} c_n z^n$	$\sum_{n\geq 0} c_n \frac{\xi_z^n(t)}{n!}$
alors, en particulier	
$z^* = \sum_{n\geq 0} z^n$	$\exp(\xi_z(t))$
$\sum_{n\geq 0} n z^n$	$\xi_z(t) \exp(\xi_z(t))$
$\sum_{n\geq 0} (-1)^n z^{2n}$	$\cos(\xi_z(t))$
$\sum_{n\geq 0} (-1)^n z^{2n+1}$	$\sin(\xi_z(t))$

Evidemment, pour $z = z_0$, nous avons :

S	$\mathcal{E}_a(f;S)$
ϵ	$f(t)$
z_0^n	$\int_0^t \frac{(t-\tau)^n}{n!} df(\tau)$
$z_0^{*n}, n \geq 1$	$\sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp(t-\tau)\frac{(t-\tau)^j}{j!} df(\tau)$
$\sum_{n\geq 0} c_n z_0^n$	$\sum_{n\geq 0} c_n \int_0^t \frac{(t-\tau)^n}{n!} df(\tau)$
alors, en particulier	
$z_0^* = \sum_{n\geq 0} z_0^n$	$\int_0^t \exp(t-\tau) df(\tau)$
$\sum_{n\geq 0} n z_0^n$	$\int_0^t (t-\tau)\exp(t-\tau) df(\tau)$
$\sum_{n\geq 0} (-1)^n z_0^{2n}$	$\int_0^t \cos(t-\tau) df(\tau)$
$\sum_{n\geq 0} (-1)^n z_0^{2n+1}$	$\int_0^t \sin(t-\tau) df(\tau)$

Et finalement, nous obtenons la transformation Laplace-Borel inverse :

S	$\mathcal{E}_a(S)$
ϵ	1
z_0^n	$\frac{t^n}{n!}$
$z_0^{*n}, n \geq 1$	$\exp(t) \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(t)^j}{j!}$
$\sum_{n\geq 0} c_n z_0^n$	$\sum_{n\geq 0} c_n \frac{t^n}{n!}$
alors, en particulier	
$z_0^* = \sum_{n\geq 0} z_0^n$	$\exp(t)$
$\sum_{n\geq 0} n z_0^n$	$t\exp(t)$
$\sum_{n\geq 0} (-1)^n z_0^{2n}$	$\cos(t)$
$\sum_{n\geq 0} (-1)^n z_0^{2n+1}$	$\sin(t)$

CHAPITRE 3

SERIES DE CHEN, SERIES DE FLIESS, SERIES DE VOLTERRA

3.0. Introduction

Les séries de Chen ont été introduites par M. Fliess et H.J. Sussmann en théorie de la commande pour coder la contribution des entrées des systèmes dynamiques non linéaires. Elles sont des outils adéquats pour décrire les "entrées généralisées" ([F3]). Elles interviennent aussi dans la démonstration par H.J. Sussmann de la conjecture de H. Hermes sur la commandabilité locale ([S4]). Elles sont, à homomorphisme près, les "opérateurs de transport" qu'étudient simultanément T. Huillet, A. Monin et G. Salut ([HMS1], [HMS2], [HMS3]) et H.J. Sussmann ([S6]) pour donner une factorisation en produit infini des sorties des systèmes dynamiques non linéaires en utilisant la base de Hall de l'algèbre de Lie des champs de vecteurs. En fait, on peut obtenir un tel produit infini en utilisant n'importe quelle base de l'algèbre de Lie libre engendrée par l'alphabet de codage de M. Fliess, mais la base de Lyndon est probablement mieux adaptée à un calcul par la tranformation d'Evaluation (voir l'annexe E).

Dans ce chapitre, nous redémontrerons un résultat concernant la concaténation des séries de Chen en utilisant un lemme de translation temporelle. Nous donnerons une graduation des séries de Chen basée sur l'égalité suivante entre expressions rationnelles :

$$(z_0 + z)^* = z_0^*(zz_0^*)^*.$$

Ensuite nous étudierons le lien entre les séries de Chen et les séries de Fliess par intermédiaire de la dualité suivante :

. si ϕ et χ sont les morphismes de monoïde, de Z^* dans $\mathbb{R}_+$ vérifiant la condition $\sum_{z \in Z} \chi(z)\phi(z) < 1$,

. si la série de Chen C est ϕ−exponentiellement majorée (voir définition 1.6.1.1.),

. si la série génératrice $\sigma_q f$, associée à une observation f relative à un polysystème, vérifie la condition de χ−croissance (voir définition 1.6.1.2.),

. alors la série $\sum_{z \in Z^*} < C|w >< \sigma_q f|w >$ est normalement convergente.

Nous obtenons ainsi une condition suffisante pour que la série de $\sigma_q f$ soit "Evaluable" au sens du chapitre II. Et dans ce cas, l'Evaluation de la série de $\sigma_q f$ peut s'interpréter comme l'action de la série C sur l'observation f (voir 1.8.1.). Avec les mêmes conditions, nous déduisons simplement les développements de Taylor des noyaux de Volterra.

Avant de commencer l'étude du lien entre les séries de Chen, les séries de Fliess et les séries de Volterra, nous rappelons très brièvement les propriétés des séries de Volterra et les techniques qui s'y rattachent (transformation de Laplace multidimensionnelle et technique d'association de variables) :

Pour généraliser le développement de Taylor ordinaire, V. Volterra a proposé ([V2]), pour une fonctionnelle de $a : [0,1] \to I\!\!K$, le développement suivant :

$$k_0 + \int_0^1 k_1(\tau_1)a(\tau_1)d\tau_1 + \int_0^1 \int_0^1 k_2(\tau_2,\tau_1)a(\tau_2)a(\tau_1)d\tau_2 d\tau_1 + \ldots,$$

où $k_0 \in I\!\!K, k_1 : [0,1] \to I\!\!K, k_2 : [0,1]^2 \to I\!\!K, \ldots$ sont des noyaux, qui sont choisis, ainsi que a, de manière à assurer l'existence des intégrales et la convergence de la formule précédente. Les bornes d'intégration de cette formule étant fixes, il n'y a pas d'évolution temporelle. Elles ont été modifiées par J.F. Barrett ([B2]) et M. Schetzen ([S1]) pour devenir dynamiques et on obtient ainsi "les séries de Volterra" :

$$h_0(t) + \int_0^t h_1(t,\tau_1)a(\tau_1)d\tau_1 + \int_0^t \int_0^t h_2(t,\tau_2,\tau_1)a(\tau_2)a(\tau_1)d\tau_2 d\tau_1 + \ldots$$

dont les noyaux sont les fonctions $h_0(t), h_1(t,\tau_1), h_2(t,\tau_2,\tau_1), \ldots$ Ces noyaux sont dits triangulaires, si lorsqu'il existe un entier positif i tel que $\tau_i \geq t$, alors les noyaux $\{h_n\}_{n \geq i}$ sont nuls. Dans ce cas la série de Volterra s'écrit encore sous la forme dite "triangulaire" :

$$h_0(t) + \int_0^t h_1(t,\tau_1)a(\tau_1)d\tau_1 + \int_0^t \int_0^{\tau_1} h_2(t,\tau_2,\tau_1)a(\tau_2)a(\tau_1)d\tau_2 d\tau_1 + \ldots$$

Sous ces formes dynamiques les séries de Volterra sont introduites en théorie de la commande par C. Lesiak et A.J. Krener ([LK]) pour approximer les sorties des systèmes dynamiques non linéaires à un nombre finis de noyaux non nuls. La structure géométrique de ces noyaux est étudiée par R.W. Brockett ([B6]) et par E.G. Gilbert ([G4]). Ces études montrent que l'approximation par un nombre fini de noyaux non nuls revient à approximer un système analytique par un système bilinéaire (ou régulier) dont la série génératrice est rationnelle. Les propriétés des noyaux de Volterra des systèmes bilinéaires ont été étudiées par C. Bruni, G. Di Pillo et G. Koch ([BDK]). Ces propriétés sont aussi étudiées par W.J. Rugh en utilisant les tranformations de Laplace multidimensionnelles ([R5]) :

Etant donné une entrée a, la réponse y du système est exprimée par la somme infinie suivante :

$$y(t) = y_0(t) + y_1(t) + \ldots + y_n(t) + \ldots$$

où pour tout entier positif n :

$$y_n(t) = \int_0^t \int_0^t \cdots \int_0^t h_n(t, \tau_1, \ldots, \tau_n) a(\tau_n) \ldots a(\tau_1) d\tau_n \ldots d\tau_1$$

avec les noyaux $\{h_n\}_{n \geq i}$ triangulaires (système causal). Ainsi pour t fixé, on peut écrire :

$$y_n(t) = \int_0^\infty \int_0^\infty \cdots \int_0^\infty h_n(t, \tau_1, \ldots, \tau_n) a(\tau_n) \ldots a(\tau_1) d\tau_n \ldots d\tau_1.$$

Le système est dit stationnaire s'il existe, pour tout entier positif n, une fonction g_n telle que $g_n(t - \tau_1, \ldots, t - \tau_n) = h_n(t, \tau_1, \ldots, \tau_n)$.

Pour mieux exposer, nous nous plaçons dans le cadre des systèmes stationnaires :

$$y(t) = \sum_{n \geq 0} y_n(t) = \sum_{n \geq 0} \int_0^\infty \cdots \int_0^\infty g_n(t - \tau_1, \ldots, t - \tau_n) a(\tau_n) \ldots a(\tau_1) d\tau_n \ldots d\tau_1.$$

Une méthode pour calculer cette réponse y du système consiste à évaluer dans le domaine temporel les intégrales y_n. L'intérêt d'une telle méthode est relativement limité car elle peut introduire pour des temps "grands" une erreur importante qui s'ajoute à l'erreur de troncature de la série. Une autre technique consiste à introduire un calcul symbolique généralisant le calcul de Heaviside et effectuer le calcul dans le cadre des transformées de Laplace multidimensionnelles. En effet, en introduisant n variables $t_1, \ldots, t_n$ et une nouvelle $\tilde{y}_n$ fonction tels que :

$$\tilde{y}_n(t_1, \ldots, t_n) = \int_0^\infty \cdots \int_0^\infty g_n(t_1 - \tau_1, \ldots, t_n - \tau_n) a(\tau_n) \ldots a(\tau_1) d\tau_n \ldots d\tau_1,$$

et $\quad y_n(t) = \tilde{y}_n(t_1, \ldots, t_n)_{|t_1 = \ldots = t_n = t}$,

on obtient d'après le théorème de convolution (voir [R5]) :

$$\tilde{Y}_n(s_1, \ldots, s_n) = G_n(s_1, s_2, \ldots, s_n) A(s_1) \ldots A(s_n),$$

où $\tilde{Y}_n(s_1, \ldots, s_n), G_n(s_1, s_2, \ldots, s_n)$ et $A(s)$ sont des transformées de Laplace de $\tilde{y}_n(t_1, \ldots, t_n), g_n(t_1 - \tau_1, \ldots, t_n - \tau_n)$ et $a(t)$ respectivement :

$$\tilde{Y}_n(s_1, \ldots, s_n) = \int_0^\infty \int_0^\infty \cdots \int_0^\infty y_n(t_1, \ldots, t_n) e^{-(s_1 t_1 + \ldots + s_n t_n)} dt_1 \ldots dt_n,$$

$$G_n(s_1, s_2, \ldots, s_n) = \int_0^\infty \int_0^\infty \cdots \int_0^\infty g_n(t_1, \ldots, t_n) e^{-(s_1 t_1 + \ldots + s_n t_n)} dt_1 \ldots dt_n,$$

$$A(s) = \int_0^\infty a(t) e^{-st} dt.$$

On peut ainsi déterminer la fonction $\tilde{y}_n$ par la transformation de Laplace multidimensionnelle inverse :

$$\tilde{y}_n(t_1,\ldots,t_n) = \mathcal{L}^{-1}\tilde{Y}_n(s_1,\ldots,s_n) = \left(\frac{1}{2i\pi}\right)^n \int_{\sigma-i\infty}^{\sigma+i\infty}\cdots\int_{\sigma-i\infty}^{\sigma+i\infty} \tilde{Y}_n(s_1,\ldots,s_n)e^{s_1t_1+\ldots+s_nt_n}ds_1\ldots ds_n,$$

et on obtient alors y_n (rappelons que $y_n(t) = \tilde{y}_n(t_1,\ldots,t_n)_{|t_1=\ldots=t_n=t}$) :

$$y_n(t) = \left(\frac{1}{2i\pi}\right)^n \int_{\sigma-i\infty}^{\sigma+i\infty}\cdots\int_{\sigma-i\infty}^{\sigma+i\infty} \tilde{Y}_n(s_1,\ldots,s_n)e^{(s_1+\ldots+s_n)t}ds_1\ldots ds_n.$$

En posant $s = s_1 + \ldots + s_n$, on peut écrire :

$$y_n(t) = \frac{1}{2i\pi}\int_{\sigma-i\infty}^{\sigma+i\infty}\left[\left(\frac{1}{2i\pi}\right)^{n-1}\int_{\sigma-i\infty}^{\sigma+i\infty}\cdots\int_{\sigma-i\infty}^{\sigma+i\infty} \tilde{Y}_n(s_1,\ldots,s_{n-1},s-s_1-\ldots-s_{n-1})ds_1\ldots ds_{n-1}\right]e^{st}ds.$$

Or si $Y_n(s)$ désigne la transformée de Laplace monodimensionnelle de y_n :

$$y_n(t) = \frac{1}{2i\pi}\int_{\sigma-i\infty}^{\sigma+i\infty} Y_n(s)e^{st}ds$$

alors on en déduit que :

$$\begin{aligned} Y_n(s) &= \left(\frac{1}{2i\pi}\right)^{n-1}\int_{\sigma-i\infty}^{\sigma+i\infty}\cdots\int_{\sigma-i\infty}^{\sigma+i\infty} \tilde{Y}_n(s_1,\ldots,s_{n-1},s-s_1-\ldots-s_{n-1})ds_1\ldots ds_{n-1} \\ &= Ass_n[\tilde{Y}_n(s_1,\ldots,s_n)](s). \end{aligned}$$

Finalement la réponse y du système est donnée par de simples transformées de Laplace monodimensionnelle inverses comme dans le cas linéaire :

$$y(t) = \frac{1}{2i\pi}\sum_{n\geq 0}\int_{\sigma-i\infty}^{\sigma+i\infty} Ass_n[\tilde{Y}_n(s_1,\ldots,s_n)](s)e^{st}ds.$$

Une telle technique est dûe à D. George ([G3]), elle est appelée technique d'association de variables, dont les principales propriétés sont ([R5]) :

(P1) *Si* $F(s_1,\ldots,s_n) = H(s_1,\ldots,s_k)G(s_{k+1},\ldots,s_n)$ *alors :*

$$Ass_n[F(s_1,\ldots,s_n)](s) = \frac{1}{2i\pi}\int_{\sigma-i\infty}^{\sigma+i\infty} Ass_k[H(s_1,\ldots,s_k)](s-u)\, Ass_{n-k}[G(s_{k+1},\ldots,s_n)](u)du.$$

(P2) *Si* $F(s_1,\ldots,s_n) = H(s_1+\ldots+s_n)G(s_1,\ldots,s_n)$ *alors :*

$$Ass_n[F(s_1,\ldots,s_n)](s) = Ass_n[H(s_1+\ldots+s_n)](s)Ass_n[G(s_1,\ldots,s_n)](s).$$

La technique d'association de variables semble puissante mais elle pose de redoutables problèmes d'algorithmique quand à l'implantation effective. C'est une raison pour laquelle cette technique est très peu utilisée.

3.1. Séries de Chen ([C1], [C2], [R1])

3.1.1. Définitions et exemple fondamental

Définition 3.1.1.1. : ([F5], [S4]) *On appelle* **série de Chen de l'entrée** $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ *la série définie somme suit :*

$$\mathcal{C}_a = \sum_{w \in Z^*} \mathcal{E}_a(w)(t) w.$$

Théorème 3.1.1.1. : *Toute série de Chen $\mathcal{C}_a$ de l'entrée a est ϕ_t–exponentiellement majorée (définition 1.6.1.1.), où ϕ_t le morphisme de monoïde, de Z^* dans $\mathbb{R}_+$, défini par :*

$$\forall z \in Z, \quad \phi_t(z) = M_t(z)t,$$

et M_t est un morphisme majorant de l'entrée a au temps t (définition 2.1.2.). Par conséquent, nous avons :

$$\mathcal{C}_a \in \mathbb{R}^{\phi_t - em} \ll Z \gg .$$

<u>*Preuve*</u> : C'est une conséquence directe du lemme 2.2.1. •

Définition 3.1.1.2. : *Nous notons par "$\mathcal{CHEN}^{\phi_t}$" l'ensemble de tous les séries de Chen des entrées relatives à l'alphabet fini Z.*

Les propriétés de l'intégration par parties permettent de montrer (lemme 2.3.1.) que toute série de Chen $\mathcal{C}_a$ vérifie le "critère de Friedrich" :

$$\forall u,\ v \in Z^*, \quad < \mathcal{C}_a | u \sqcup\!\sqcup v > = < \mathcal{C}_a | u >< \mathcal{C}_a | v > .$$

Par conséquent ([R1]), $\mathcal{C}_a$ est l'exponentielle d'une série de Lie ([JO1]) :

$$\mathcal{C}_a = \exp\Big(\sum_{k \geq 1} L_k(\mathcal{C}_a) \Big),$$

où $L_k(\mathcal{C}_a)$, $k \geq 1$, est un polynôme de Lie homogène degré k. Ce résultat est une généralisation de la formule de Baker-Campbell-Hausdorff. On peut aussi obtenir $\mathcal{C}_a$ comme produit infini d'exponentielles d'éléments d'une base de Hall de l'algèbre de Lie libre ([HMS1], [HMS2], [HMS3], [S6]). En fait, on peut obtenir un tel produit infini en utilisant n'importe quelle base de l'algèbre de Lie libre engendrée par l'alphabet de codage de M. Fliess, mais la base de Lyndon est probablement mieux adaptée à un calcul par la tranformation d'Evaluation (voir l'annexe E).

Exemple 3.1.1.1.

La série de Chen de l'entrée **constante** c est :

$$\mathcal{C}_c = \exp\left(t \sum_{z \in Z} c^z z\right).$$

En particulier, si $t = 0$ alors la série de Chen de l'entrée o définie en zéro est :

$$\mathcal{C}_o = \epsilon.$$

La série de Chen de l'entrée $e_{z_0} = (1 \quad 0 \quad \ldots \quad 0)$ définie sur $[0, t^2]$ est :

$$\mathcal{C}^t_{e_{z_0}} = \exp(t^2 z_0).$$

Pour toute lettre $z \in Z_0$, soit e_z l'entrée définie par :

$$\forall \tau \in [0, t^2], \quad e_z(\theta) = \begin{pmatrix} 1 & 0 & \ldots & \frac{1}{t} & \ldots & 0 \end{pmatrix}.$$

La série de Chen de l'entrée e_z est :

$$\mathcal{C}^t_{e_z} = \exp(t^2 z_0 + tz).$$

On peut vérifier aisément (par d'éveloppement de Taylor) que :

$$\lim_{t \to 0} \frac{1}{t^2} [\mathcal{C}^t_{e_{z_0}} - 1] = z_0,$$

et pour toute lettre $z \in Z_0$, on a :

$$\lim_{t \to 0} \frac{1}{t} [\mathcal{C}^t_{e_z} - \mathcal{C}^t_{e_{z_0}}] = z.$$

3.1.2. Concaténation des séries de Chen

Définition 3.1.2.1. : ([F5], [S4]) *Nous appelons* **concaténation de deux entrées** $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ *et* $b = (b^{z_0} \quad b^{z_1} \quad \ldots \quad b^{z_m})$ *définies sur* $[0, t_a]$ *et* $[0, t_b]$, *la nouvelle entrée* $a\sharp b$ *définie comme suit :*

$$a\sharp b(t) = a(t) \quad \text{pour} \quad 0 \le t < t_a$$
$$a\sharp b(t_a + t) = b(t) \quad \text{pour} \quad 0 \le t < t_b.$$

Proposition 3.1.2.1. : ([C2], [S4]) *Avec les notations précédentes, la série de Chen de l'entrée $a\sharp b$ est égale produit de Cauchy de la série de Chen $\mathcal{C}_a$ de l'entrée a avec la série de Chen $\mathcal{C}_b$ de l'entrée b :*

$$\mathcal{C}_{a\sharp b} = \mathcal{C}_a \mathcal{C}_b.$$

<u>*Exemple*</u> : La série de Chen de l'entrée constante a (resp. b) définie sur $[0, t_a]$ (resp. $[0, t_b]$) est :

$$\mathcal{C}_a = \exp\Big(t_a \sum_{z\in Z} a^z z\Big) \quad \Big(\text{resp. } \mathcal{C}_b = \exp\Big(t_b \sum_{z\in Z} b^z z\Big)\Big),$$

et la série de Chen de l'entrée $a\sharp b$ définie sur $[0, t_a + t_b]$ est :

$$\mathcal{C}_{a\sharp b} = \exp\Big(t_a \sum_{z\in Z} a^z z\Big) \exp\Big(t_b \sum_{z\in Z} b^z z\Big).$$

La preuve de la proposition 3.1.2.1. donnée par H.J. Sussman dans [S4] utilise une équation différentielle structurelle vérifiée par $\mathcal{C}_{a\sharp b}$. On peut aussi donner une preuve élémentaire directement comme suit :

Définition 3.1.2.2. : ([HJ4], [HJ5]) *Soit θ un réel fixé. Soit e une fonction de $L^\infty(\theta, \theta+t)$, $t \geq 0$. Nous définissons l'***intégrale itérée θ–translatée** *par :*

$$\int_\theta^{\theta+t} \delta_e w = \begin{cases} 1 & si \quad w = \epsilon, \\ \displaystyle\int_0^t \Big(\int_\theta^{\theta+\tau} \delta_e v\Big) d\xi_z(\theta+\tau) & si \quad w = vz. \end{cases}$$

Remarque :

(i) L'intégrale itérée $\displaystyle\int_0^t \delta_e w$ *n'est pas une intégrale ordinaire. En particulier, l'additivité est en défaut. En effet on a :*

$$\int_0^{\theta+t} \delta_e w \neq \int_0^\theta \delta_e w + \int_\theta^{\theta+t} \delta_e w.$$

(Déjà pour $w = \epsilon$, cela entraînerait 1=1+1!)

(ii) Les intégrales itérées $\displaystyle\int_{t_a}^{t_a+t_b} \delta_{a\sharp b} v$ *et* $\displaystyle\int_0^{t_b} \delta_b v = \mathcal{E}_b(v)(t_b)$ *sont égales, ayant la même définition récurrente ($a\sharp b$ étant la concaténée des entrées $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ et $b = (\, b^{z_0} \quad b^{z_1} \quad \ldots \quad b^{z_m}\,)$ définies sur $[0, t_a]$ et $[0, t_b]$).*

La proposition 3.1.2.1. équivaut alors au lemme suivant :

Lemme 3.1.2.1. : (lemme de translation temporelle, [HJ4], [HJ5]) *Soit θ un réel fixé. Soit e une fonction de $L^\infty(0, \theta + t)$, $t \geq 0$. Nous avons l'égalité suivante :*

$$\mathcal{E}_e(w)(\theta + t) = \sum_{\substack{u,v \in Z^* \\ uv = w}} \mathcal{E}_e(u)(\theta) \int_\theta^{\theta+t} \delta_e v = \sum_{\substack{u,v \in Z^* \\ uv = w}} \int_0^\theta \delta_e u \int_\theta^{\theta+t} \delta_e v.$$

<u>*Preuve*</u> :

. Puisque $\displaystyle\int_0^{\theta+t} \delta_e \epsilon = 1$, alors le résultat est immédiat pour $w = \epsilon$.

. Supposons que le résultat soit satisfait pour tout mot w, $0 \leq |w| \leq n$.

. Si $|w| = n + 1$ alors w peut s'écrire en $w = w_1 z$, $w_1 \in Z^*$, $z \in Z$. Par conséquent, nous avons :

$$\begin{aligned}
\mathcal{E}_e(w)(\theta + t) &= \int_0^{\theta+t} \mathcal{E}_e(w_1)(\rho)\ d\xi_z(\rho) \\
&= \int_0^\theta \mathcal{E}_e(w_1)(\rho) d\xi_z(\rho) + \int_\theta^{\theta+t} \mathcal{E}_e(w_1)(\rho)\ d\xi_z(\rho) \\
&\qquad \text{(additivité des intégrales ordinaires)} \\
&= \mathcal{E}_e(w_1 z)(\theta) + \int_0^t \mathcal{E}_e(w_1)(\theta + \tau)\ d\xi_z(\theta + \tau) \\
&\qquad \text{(nous avons posé } \rho = \theta + \tau\text{)} \\
&= \mathcal{E}_e(w_1 z)(\theta) + \int_0^t \sum_{\substack{u_1,v_1 \in Z^* \\ u_1 v_1 = w_1}} \mathcal{E}_e(u_1)(\theta) \int_\theta^{\theta+\tau} \delta_e v_1\ d\xi_z(\theta + \tau) \\
&\qquad \text{(d'après l'hypothèse de récurrence)} \\
&= \mathcal{E}_e(w_1 z)(\theta).1 + \sum_{\substack{u_1,v_1 \in Z^* \\ u_1 v_1 = w_1}} \mathcal{E}_\epsilon(u_1)(\theta) \int_0^t \int_\theta^{\theta+\tau} \delta_e v_1\ d\xi_z(\theta + \tau) \\
&= \mathcal{E}_e(w_1 z)(\theta) \int_\theta^{\theta+t} \delta_e \epsilon + \sum_{\substack{u_1,v_1 \in Z^* \\ u_1 v_1 = w_1}} \mathcal{E}_e(u_1)(\theta) \int_\theta^{\theta+t} \delta_e(v_1 z) \\
&\qquad \text{(d'après la définition 3.1.2.1.)} \\
&= \mathcal{E}_e(w_1 z)(\theta) \int_\theta^{\theta+t} \delta_e \epsilon + \sum_{\substack{u,v \in Z^*, v \neq \epsilon \\ uv = w_1 z}} \mathcal{E}_e(u)(\theta) \int_\theta^{\theta+t} \delta_e v \\
&\qquad \text{(distributivité des intégrales ordinaires)} \\
&= \sum_{\substack{u,v \in Z^* \\ uv = w}} \mathcal{E}_e(u)(\theta) \int_\theta^{\theta+t} \delta_e v \quad \bullet
\end{aligned}$$

Preuve de la proposition 3.1.2.1. : D'après la définition de $a\sharp b$ et le lemme de translation temporelle, nous avons :

$$\mathcal{C}_{a\sharp b} = \sum_{w\in Z^*} \mathcal{E}_{a\sharp b}(w)(t_a+t_b)w = \sum_{w\in Z^*}\left[\sum_{\substack{u,v\in Z^*\\ uv=w}} \mathcal{E}_{a\sharp b}(u)(t_a)\int_{t_a}^{t_a+t_b}\delta_{a\sharp b}v\right]w.$$

D'après la remarque (ii), on a :

$$\begin{aligned}\mathcal{C}_{a\sharp b} &= \sum_{w\in Z^*}\sum_{\substack{u,v\in Z^*\\ uv=w}} \mathcal{E}_a(u)(t_a)\mathcal{E}_b(v)(t_b)uv\\ &= \left[\sum_{u\in Z^*}\mathcal{E}_a(u)(t_a)u\right]\left[\sum_{v\in Z^*}\mathcal{E}_b(v)(t_b)v\right] = \mathcal{C}_a\mathcal{C}_b \quad \bullet\end{aligned}$$

3.1.3. Indiscernabilité

Définition 3.1.3.1. : ([HJ4], [HJ5]) *Soit S une série formelle continue sur une classe Cl. Alors S est dite* **indiscernable sur une classe** Cl *si :*

$$\forall C \in Cl, \quad < S\|C > = 0.$$

Lemme 3.1.3.1. : ([HJ4], [HJ5]) *Si une série formelle S continue sur $\mathcal{CHEN}^{\phi_t}$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$ alors pour toute lettre z de Z, la série $z \triangleleft S$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$*

Preuve :

. Si $z \neq z_0$ alors :

$$\begin{aligned}< z\triangleleft S\|\mathcal{C}_a > &= < S\|\mathcal{C}_a \lim_{t\to 0}\frac{1}{t}[\mathcal{C}^t_{e_z} - \mathcal{C}^t_{e_{z_0}}] > && \text{(exemple 3.1.1.1.)}\\ &= \lim_{t\to 0}\frac{1}{t} < S\|\mathcal{C}_a[\mathcal{C}^t_{e_z} - \mathcal{C}^t_{e_{z_0}}] > && \text{(convergence uniforme)}\\ &= \lim_{t\to 0}\frac{1}{t} < S\|\mathcal{C}_{a\sharp e_z} - \mathcal{C}_{a\sharp e_{z_0}} > && \text{(proposition 3.1.2.1.)}\\ &= \lim_{t\to 0}\frac{1}{t}[< S\|\mathcal{C}_{a\sharp e_z} > - < S\|\mathcal{C}_{a\sharp e_{z_0}} >]\\ &= 0 \quad (S \text{ est indiscernable sur } \mathcal{CHEN}^{\phi_t}).\end{aligned}$$

. Si $z = z_0$ alors :

$$\begin{aligned}< z_0\triangleleft S\|\mathcal{C}_a > &= < S\|\mathcal{C}_a \lim_{t\to 0}\frac{1}{t^2}[\mathcal{C}^t_{e_{z_0}} - 1] > && \text{(exemple 3.1.1.1.)}\\ &= \lim_{t\to 0}\frac{1}{t^2} < S\|\mathcal{C}_a[\mathcal{C}^t_{e_{z_0}} - 1] > && \text{(convergence uniforme)}\\ &= \lim_{t\to 0}\frac{1}{t^2} < S\|\mathcal{C}_{a\sharp e_{z_0}} - \mathcal{C}_a > && \text{(proposition 3.1.2.1.)}\\ &= \lim_{t\to 0}\frac{1}{t^2}[< S\|\mathcal{C}_{a\sharp e_{z_0}} > - < S\|\mathcal{C}_a >]\\ &= 0 \quad (S \text{ est indiscernable sur } \mathcal{CHEN}^{\phi_t}) \quad \bullet\end{aligned}$$

Corollaire 3.1.3.1. : ([HJ4], [HJ5]) *Si une série formelle S continue sur $\mathcal{CHEN}^{\phi_t}$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$ alors pour tout mot w de Z^*, $w \triangleleft S$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$.*

Preuve :

. Puisque S est indiscernable sur $\mathcal{CHEN}^{\phi_t}$, le résultat est immédiat pour $w = \epsilon$.

. Supposons que le résultat soit vrai pour tout mot w, $0 \leq |w| \leq n$.

. Pour w, $|w| = n+1$, on peut écrire $w = zw_1$. On a :

$$w \triangleleft S = z \triangleleft (w_1 \triangleleft S) \quad \text{(voir 1.3.)}.$$

D'après l'hypothèse de récurrence, $w_1 \triangleleft S$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$. Par conséquent, $w \triangleleft S = z \triangleleft (w_1 \triangleleft S)$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$ (lemme 3.1.3.1.) •

Théorème 3.1.3.1. : ([HJ4], [HJ5]) *Une série formelle S continue sur $\mathcal{CHEN}^{\phi_t}$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$ si et seulement si $S \equiv 0$.*

Preuve : Il est immédiat que si $S \equiv 0$ alors S est indiscernable sur $\mathcal{CHEN}^{\phi_t}$. Si S est indiscernable sur $\mathcal{CHEN}^{\phi_t}$ alors d'après le corollaire 4.1.3.1., $w \triangleleft S$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$ pour tout mot w de Z^*, nous avons :

$$< S|w > = < w \triangleleft S|\epsilon > = < w \triangleleft S||\mathcal{C}_o > = 0 \quad \text{(voir l'exemple 3.1.1.1.)}.$$

Par conséquent :

$$S \equiv 0 \quad \bullet$$

3.1.4. Une graduation des séries de Chen

Soit a une entrée relative à l'alphabet fini $Z = \{z_0, z_1, \ldots, z_m\}$.

D'après 1.4.1., la série de Chen $\mathcal{C}_a$ de entrée a définie sur $[0,t]$ peut être s'écrire en :

$$\mathcal{C}_a = \sum_{k \geq 0} V_k(\mathcal{C}_a) = \sum_{k \geq 0} \sum_{w \in (z_0^* Z_0)^k z_0^*} \mathcal{E}_a(w)(t) w.$$

Proposition 3.1.4.1. : ([HJ4], [HJ5]) *Soit a une entrée relative à l'alphabet fini $Z = \{z_0, z_1, \ldots, z_m\}$. Si $\mathcal{C}_a$ est la série de Chen de cette entrée, alors pour tout entier positif k, la série $V_k(\mathcal{C}_a)$ peut s'exprimer par :*

$$\sum_{z_{i_1} \ldots z_{i_k} \in Z_0^k} \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} e^{\tau_1 z_0} z_{i_1} e^{(\tau_2 - \tau_1) z_0} \ldots z_{i_k} e^{(t - \tau_k) z_0} \, d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k).$$

<u>*Preuve*</u> : Pour tout entier positif k, chaque mot $w \in (z_0^* Z_0)^k z_0^*$ peut s'exprimer en $w = z_0^{\alpha_0} z_{i_1} z_0^{\alpha_1} \ldots z_{i_k} z_0^{\alpha_k}$, où $\alpha_0, \ldots, \alpha_k$ sont des entiers positifs, et $z_{i_1}, \ldots, z_{i_k}$ sont des lettres de Z_0. D'après le corollaire 2.5.1., nous avons :

$$\mathcal{E}_a(w)(t) = \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} \frac{\tau_1^{\alpha_0}(\tau_2 - \tau_1)^{\alpha_1} \ldots (t - \tau_k)^{\alpha_k}}{\alpha_0! \ldots \alpha_k!} d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k).$$

Nous pouvons alors déduire que pour tout entier positif k :

$$\begin{aligned}
V_k(\mathcal{C}_a) &= \sum_{z_{i_1} \ldots z_{i_k} \in Z_0^k} \sum_{\alpha_0, \ldots, \alpha_k \geq 0} \mathcal{E}_a(z_0^{\alpha_0} z_{i_1} z_0^{\alpha_1} \ldots z_{i_k} z_0^{\alpha_k})(t)\; z_0^{\alpha_0} z_{i_1} z_0^{\alpha_1} \ldots z_{i_k} z_0^{\alpha_k} \\
&= \sum_{z_{i_1} \ldots z_{i_k} \in Z_0^k} \sum_{\alpha_0, \ldots, \alpha_k \geq 0} \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} \frac{\tau_1^{\alpha_0}(\tau_2 - \tau_1)^{\alpha_1} \ldots (t - \tau_k)^{\alpha_k}}{\alpha_0! \alpha_1! \ldots ! \alpha_k!} \\
&\qquad d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k)\; z_0^{\alpha_0} z_{i_1} z_0^{\alpha_1} \ldots z_{i_k} z_0^{\alpha_k} \\
&= \sum_{z_{i_1} \ldots z_{i_k} \in Z_0^k} \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} \sum_{\alpha_0, \ldots, \alpha_k \geq 0} \\
&\qquad \frac{(\tau_1 z_0)^{\alpha_0} z_{i_1} [(\tau_2 - \tau_1) z_0]^{\alpha_1} \ldots z_{i_k} [(t - \tau_k) z_0]^{\alpha_k}}{\alpha_0! \alpha_1! \ldots ! \alpha_k!} d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k) \\
&= \sum_{z_{i_1} \ldots z_{i_k} \in Z_0^k} \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} \\
&\qquad e^{\tau_1 z_0} z_{i_1} e^{(\tau_2 - \tau_1) z_0} \ldots z_{i_k} e^{(t - \tau_k) z_0} d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k) \quad \bullet
\end{aligned}$$

Puisque $\mathcal{C}_a = \sum_{k \geq 0} V_k(\mathcal{C}_a)$, nous avons le théorème de graduation des séries de Chen suivant :

Théorème 3.1.4.1. : ([HJ4], [HJ5]) *Soit a une entrée relative à l'alphabet fini $Z = \{z_0, z_1, \ldots, z_m\}$. La série de Chen $\mathcal{C}_a$ est décrite par :*

$$\begin{aligned}
\mathcal{C}_a &= \sum_{k \geq 0} \sum_{z_{i_1} \ldots z_{i_k} \in Z_0^k} \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} e^{\tau_1 z_0} z_{i_1} e^{(\tau_2 - \tau_1) z_0} \ldots z_{i_k} e^{(t - \tau_k) z_0} \\
&\qquad d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k) \\
&= \sum_{k \geq 0} \sum_{z_{i_1} \ldots z_{i_k} \in Z_0^k} \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} (e^{ad_{\tau_1 z_0}} z_{i_1}) \ldots (e^{ad_{\tau_k z_0}} z_{i_k}) e^{t z_0} \\
&\qquad d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k).
\end{aligned}$$

La seconde égalité est obtenue en utilisant la formule suivante ([B5]) :

$$e^{\alpha}\beta e^{-\alpha} = \sum_{\nu\geq 0} \frac{1}{\nu!} ad_{\alpha}^{\nu}\beta = e^{ad_{\alpha}}\beta,$$

où $ad_{\alpha}^{\nu}\beta$, $\nu \geq 1$, est le crochet de Lie ν–itéré $[\alpha, [\ldots, [\alpha, \beta]\ldots]]$ et $ad_{\alpha}^{0}\beta = \beta$.

Exemple : Si l'alphabet fini Z est réduit à $\{z_0\}$, alors :

$$\mathcal{C}_a = V_0(\mathcal{C}_a) = \sum_{n\geq 0} \mathcal{E}_a(z_0^n)(t) z_0^n = \sum_{n\geq 0} \frac{(tz_0)^n}{n!} = e^{tz_0}.$$

3.2. Séries de Fliess

Soit $\{A_z\}_{z\in Z}$ un polysystème sur $\Theta = \{\theta_1, \ldots, \theta_N\}$ relatif à l'alphabet fini $Z = \{z_0, z_1, \ldots, z_m\}$ (voir 1.8.). Rappelons que pour toute lettre $z \in Z$, la dérivation de Lie A_z est de la forme :

$$A_z = \sum_{D\in\mathcal{D}} A_z^D D,$$

avec les $A_z^D, D \in \mathcal{D}, z \in Z$, sont des séries de $I\!K^{conv}[\![\Theta]\!]$ telles que pour tout point $q \in \bigcap_{D\in\mathcal{D}} Conv(A_z^D)$, pour tout module de convergence $(\rho, \mathring{\rho}, C_z)$ en q de A_z^D, $(D \in \mathcal{D},\ z \in Z)$, on ait :

$$\forall z \in Z, \quad \forall D \in \mathcal{D}, \quad \left|(A_z^D)_{|q}\right| \leq C_z F(r),$$

où $r = \max_{1\leq k\leq N} \frac{\mathring{\rho}_k}{\rho_k}$ et $F(r) = (1-r)^{-N}$.

Définition 3.2.1. : *Soit f une série formelle de $I\!K^{conv}[\![\Theta]\!]$. On appelle* **série génératrice** *associée à la série f relative au polysystème $\{A_z\}_{z\in Z}$, la série formelle $\sigma_q f$ à coefficients dans $I\!K^{conv}[\![\Theta]\!]$ définie comme suit :*

$$\sigma f = \sum_{w\in Z^*} (w * f)\, w \in I\!K^{conv}[\![\Theta]\!] \ll Z \gg .$$

On appelle **série génératrice en un état** *$q \in Conv(f)$ associée à la série f relative au polysystème $\{A_z\}_{z\in Z}$, la série formelle $\sigma_q f$ à coefficients dans $I\!K$ définie comme suit :*

$$\sigma_q f = \sum_{w\in Z^*} (w * f)_{|q}\, w \in I\!K \ll Z \gg .$$

On sait (c'est la formule fondamentale de M. Fliess [F5]) que la série $\sigma_q f$ caractérise complètement le comportement local (au voisinage de q) du polysystème $\{A_z\}_{z\in Z}$, pour l'observation $f \in \mathbb{K}^{conv}[\![\Theta]\!]$, et l'état initial q. Nous verrons que deux séries formelles S, T définissent la même sortie $y(t)$ si et seulement si $S = T$ (théorème 3.2.5.).

Notation : *Pour tout mot $w \in Z^*$, on note :*

$$C_w = \begin{cases} 1 & si \quad w = \epsilon, \\ C_z C_{w_1} & si \quad w = zw_1. \end{cases}$$

Proposition 3.2.1. : *Supposons que f et A_z^D, $(D \in \mathcal{D},\ z \in Z)$ soient des séries de $\mathbb{K}^{conv}[\![\Theta]\!]$. Soit $q \in Conv(f) \bigcap_{D\in\mathcal{D}, z\in Z} Conv(A_z^D)$, soit $(\rho, \mathring{\rho}, C_f)$ un module de convergence en q de f, et soient $(\rho, \mathring{\rho}, C_z)$, $(z \in Z)$, des modules de convergence en q de A_z^D, $(D \in \mathcal{D},\ z \in Z)$. En posant $\tau = \min_{1\le k\le N} \rho_k$ et $r = \max_{1\le k\le N} \frac{\mathring{\rho}_k}{\rho_k}$, nous avons :*

$$\forall w \in Z^*, \quad \left|(w * f)_{|q}\right| \le \frac{C_f}{(1-r)^N} \frac{1}{\binom{N+|w|-1}{|w|}} \left[\frac{N(N+1)}{\tau(1-r)^{(N+1)}}\right]^{|w|} C_w\, |w|!.$$

<u>*Preuve*</u> : Soit $w = z_{i_p} \dots z_{i_1}$ un mot de Z^*. Nous avons :

$$w * f = \sum_{j_p=1}^{N} A_{z_{i_p}}^{D_{j_p}} D_{j_p} \dots \sum_{j_1=1}^{N} A_{z_{i_1}}^{D_{j_1}} D_{j_1} f = \sum_{j_p=1}^{N} \dots \sum_{j_1=1}^{N} A_{z_{i_p}}^{D_{j_p}} D_{j_p} \dots A_{z_{i_1}}^{D_{j_1}} D_{j_1} f.$$

D'après le théorème 1.7.2., nous avons :

$$\left|(A_{z_{i_p}}^{D_{j_p}} D_{j_p} \dots A_{z_{i_1}}^{D_{j_1}} D_{j_1} f)_{|q}\right| \le \frac{C_w C_f}{\binom{N+p-1}{p}\tau^p} \Delta^p F(r).$$

Par conséquent :

$$\left|(w * f)_{|q}\right| \le \frac{C_w C_f N^p}{\binom{N+p-1}{p}\tau^p} \Delta^p F(r).$$

Et finalement, d'après la proposition 1.7.1.(e), nous avons :

$$\begin{aligned} \left|(w * f)_{|q}\right| &\le \frac{C_w C_f N^p}{\binom{N+p-1}{p}\tau^p} \frac{(N+1)^p}{(1-r)^N} (1-r)^{-p(N+1)} p! \\ &= \frac{C_f}{(1-r)^N} \frac{1}{\binom{N+|w|-1}{|w|}} \left[\frac{N(N+1)}{\tau(1-r)^{(N+1)}}\right]^{|w|} C_w\, |w|! \quad \bullet \end{aligned}$$

Théorème 3.2.1. : ([F3]) *Supposons que* f *et* A_z^D, $(D \in \mathcal{D},\ z \in Z)$ *soient des séries de* $\mathbb{K}^{conv}[\![\Theta]\!]$. *Soit* $q \in Conv(f) \bigcap_{D \in \mathcal{D}, z \in Z} Conv(A_z^D)$, *soit* $(\rho, \overset{\circ}{\rho}, C_f)$ *un module de convergence en* q *de* f, *et soient* $(\rho, \overset{\circ}{\rho}, C_z)$, $(z \in Z)$, *des modules de convergence en* q *de* A_z^D, $(D \in \mathcal{D},\ z \in Z)$.

La série génératrice en q *associée à la série* f *en* q *relative au polysystème* $\{A_z\}_{z \in Z}$ *vérifie la condition de* χ*-croissance (définition 1.6.1.2.), avec :*

$$K = \frac{C_f}{(1-r)^N},$$

et χ *est le morphisme défini par :*

$$\forall z \in Z, \quad \chi(z) = \frac{N(N+1)C_z}{\tau(1-r)^{(N+1)}},$$

où $\tau = \min_{1 \le k \le N} \rho_k$ *et* $r = \max_{1 \le k \le N} \frac{\overset{\circ}{\rho}_k}{\rho_k}$.

Preuve : En effet, puisque $\binom{N+p-1}{p} \geq 1$, alors d'après la proposition 3.2.1., on a le résultat •

Théorème 3.2.2. : *Supposons que* f *et* A_z^D, $(D \in \mathcal{D},\ z \in Z)$ *soient des séries de* $\mathbb{K}^{conv}[\![\Theta]\!]$. *Soit* $q \in Conv(f) \bigcap_{D \in \mathcal{D}, z \in Z} Conv(A_z^D)$, *soit* $(\rho, \overset{\circ}{\rho}, C_f)$ *un module de convergence en* q *de* f, *et soient* $(\rho, \overset{\circ}{\rho}, C_z)$, $(z \in Z)$, *des modules de convergence en* q *de* A_z^D, $(D \in \mathcal{D},\ z \in Z)$. *Soit* σf *la série génératrice associée à la série* f *relative au polysystème* $\{A_z\}_{z \in Z}$.

Soit $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ *une entrée relative à l'alphabet fini* Z *et soit* M_t *un morphisme majorant de* a *au temps* t *(définition 2.1.2.). Soit* $\mathcal{C}_a$ *la série de Chen de l'entrée* a.

On pose $\tau = \min_{1 \le k \le N} \rho_k$ *et* $r = \max_{1 \le k \le N} \frac{\overset{\circ}{\rho}_k}{\rho_k}$. *Si :*

$$\sum_{z \in Z} C_z M_t(z) < \frac{\tau(1-r)^{(N+1)}}{N(N+1)t}$$

alors la série $\sum_{w \in Z^*} < \sigma_q f | w >< \mathcal{C}_a | w >$ *est normalement convergente.*

Preuve : C'est une conséquence immédiate du théorème 3.2.1., du théorème 3.1.1.1. et du théorème 2.4.1. •

Par conséquent, l'Evaluation de la série génératrice associée à la série f en q peut s'exprimer par l'action de la série de Chen de l'entrée a sur f :

$$\mathcal{E}_a(\sigma_q f) = \sum_{w \in Z^*} \mathcal{E}_a(w)(w * g)_{|q} = \left[\left(\sum_{w \in Z^*} \mathcal{E}_a(w) w \right) * f \right]_{|q},$$

en d'autres termes :

Théorème 3.2.3. : ([HJ4], [HJ5]) *Sous les hypothèses du théorème 3.2.2., nous avons :*

$$\mathcal{E}_a(\sigma_q f)(t) = < \sigma_q f \| \mathcal{C}_a > = (\mathcal{C}_a * f)_{|q},$$

où $\mathcal{C}_a$ est la série de Chen de l'entrée a définie sur $[0,t]$.

Corollaire 3.2.1. : ([HJ4], [HJ5]) *Soient f et g deux séries formelles de $\mathbb{K}^{conv}[\![\Theta]\!]$ vérifiant les hypothèses du théorème 3.2.2. et $\{A_z\}_{z\in Z}$ un polysystème. Nous avons l'égalité suivante :*

$$[\mathcal{C}_a * (fg)]_{|q} = (\mathcal{C}_a * f)_{|q}(\mathcal{C}_a * g)_{|q},$$

où $\mathcal{C}_a$ est la série de Chen de l'entrée a définie sur $[0,t]$.

<u>*Preuve*</u> : D'après le lemme 1.9.1., nous avons :

$$\begin{aligned}
[\mathcal{C}_a * (fg)]_{|q} &= \sum_{w\in Z^*} \mathcal{E}_a(w)[w * (fg)]_{|q} \\
&= \sum_{w\in Z^*} \mathcal{E}_a(w) \sum_{u,v\in Z^*} < u ⧢ v | w > (u * f)_{|q}(v * g)_{|q} \\
&= \sum_{u,v\in Z^*} \left[\sum_{w\in Z^*} < u ⧢ v | w > \mathcal{E}_a(w)\right](u * f)_{|q}(v * g)_{|q} \\
&= \sum_{u\in Z^*} \sum_{v\in Z^*} \mathcal{E}_a(u ⧢ v)(u * f)_{|q}(v * g)_{|q} \\
&= \sum_{u\in Z^*} \sum_{v\in Z^*} \mathcal{E}_a(u)\mathcal{E}_a(v)(u * f)_{|q}(v * g)_{|q} \quad \text{(lemme 1.8.1.1.)} \\
&= \left[\sum_{u\in Z^*} \mathcal{E}_a(u)(u * f)_{|q}\right]\left[\sum_{v\in Z^*} \mathcal{E}_a(v)(v * g)_{|q}\right] \\
&= (\mathcal{C}_a * f)_{|q}(\mathcal{C}_a * g)_{|q} \quad \bullet
\end{aligned}$$

Théorème 3.2.4. : ([HJ4], [HJ5]) *Soit f (resp. g) une série formelle de $\mathbb{K}^{conv}[\![\Theta]\!]$ vérifiant les hypothèses du théorème 3.2.2. La série génératrice associée à la série f (resp. g) relative au polysystème $\{A_z\}_{z\in Z}$ est $\sigma_q f$ (resp. $\sigma_q g$). Nous avons les égalités suivantes :*

$$\mathcal{E}_a(\sigma_q(fg)) = \quad \mathcal{E}_a(\sigma_q f)\mathcal{E}_a(\sigma_q g) \quad = \mathcal{E}_a((\sigma_q f) ⧢ (\sigma_q g)),$$

ou équivalemment :

$$< \sigma_q(fg) \| \mathcal{C}_a > = < \sigma_q f \| \mathcal{C}_a >< \sigma_q g \| \mathcal{C}_a > = < (\sigma_q f) ⧢ (\sigma_q g) \| \mathcal{C}_a >,$$

où $\mathcal{C}_a$ est la série de Chen de l'entrée a définie sur $[0,t]$.

Preuve :

$$\begin{aligned}
\mathcal{E}_a(\sigma_q(fg)) &= [\mathcal{C}_a * (fg)]_{|q} && \text{(théorème 3.2.3.)}\\
&= (\mathcal{C}_a * f)_{|q}(\mathcal{C}_a * g)_{|q} && \text{(corollaire 3.2.1.)}\\
&= \mathcal{E}_a(\sigma_q f)\mathcal{E}_a(\sigma_q g) && \text{(théorème 3.2.3.)}\\
&= \mathcal{E}_a((\sigma_q f) \sqcup\!\sqcup (\sigma_q g)) && \text{(théorème 2.4.4.).}
\end{aligned}$$

Et d'après le théorème 3.2.3., nous avons :

$$< \sigma_q(fg) \| \mathcal{C}_a > = < \sigma_q f \| \mathcal{C}_a >< \sigma_q g \| \mathcal{C}_a > = < (\sigma_q f) \sqcup\!\sqcup (\sigma_q g) \| \mathcal{C}_a > \quad \bullet$$

Définition 3.2.2. : ([HJ4], [HJ5]) *Soient S, T deux séries formelles de $\mathbb{K} \ll Z \gg$ continues sur $\mathcal{CHEN}^{\phi_t}$ (définition 1.6.1.3.). S et T sont* **indiscernables sur** $\mathcal{CHEN}^{\phi_t}$ *si et seulement si :*

$$\forall \mathcal{C}_a \in \mathcal{CHEN}^{\phi_t}, \quad < S \| \mathcal{C}_a > = < T \| \mathcal{C}_a >,$$

ou si équivalemment :

$$\textit{pour toute entrée a relative à } Z, \quad \mathcal{E}_a(S) = \mathcal{E}_a(T).$$

Théorème 3.2.5. : (théorème de l'unicité des Evaluations, [F3]) *Etant données deux séries formelles S, T de $\mathbb{K} \ll Z \gg$ continues sur $\mathcal{CHEN}^{\phi_t}$, S et T sont indiscernables sur $\mathcal{CHEN}^{\phi_t}$ si et seulement si $S = T$.*

Preuve : Si pour toute série $\mathcal{C}_a$ de $\mathcal{CHEN}^{\phi_t}$, $< S \| \mathcal{C}_a > = < T \| \mathcal{C}_a >$, alors nous avons $< S - T \| \mathcal{C}_a > = 0$. Par conséquent, $S - T$ est indiscernable sur $\mathcal{CHEN}^{\phi_t}$ (définition 3.1.3.1.). D'après le théorème 3.1.3.1., nous avons la conclusion $\bullet$

Théorème 3.2.6. : ([F3], [F6], [R3]) *L'application σ est un morphisme de l'algèbre des séries formelles en variables commutatives $\theta_1, \ldots, \theta_N$, $\mathbb{K}^{conv}[\![\Theta]\!]$, dans l'algèbre des séries formelles en variables non commutatives $z_0, z_1, \ldots, z_m$ munie le produit de mélange, $\mathbb{K}^{conv}[\![\Theta]\!] \ll Z \gg$.*

Preuve : On montre aisément que :

$$\forall f,\ g \in \mathbb{K}^{conv}[\![\Theta]\!], \quad \forall \lambda,\ \mu \in \mathbb{R}, \quad \sigma(\lambda f + \mu g) = \lambda \sigma f + \mu \sigma g.$$

D'après le théorème 3.2.4. et le théorème de l'unicité des Evaluations, nous avons :

$$\forall f,\ g \in \mathbb{K}^{conv}[\![\Theta]\!], \quad \sigma(fg) = (\sigma f) \sqcup\!\sqcup (\sigma g) \quad \bullet$$

3.3. Séries de Volterra ([B6], [LK])

Soit a une entrée relative à l'alphabet fini $Z = \{z_0, z_1, \ldots, z_m\}$.

Soit f une série formelle de $\mathbb{K}^{r\,conv}[\![\Theta]\!]$.

La série génératrice associée à la série f relative au polysystème $\{A_z\}_{z\in Z}$ est notée par σf.

Sous les hypothèses du théorème 3.2.2., d'après le théorème 3.1.4.1., l'Evaluation de $\sigma_q f$ pour l'entrée a peut s'exprimer par :

$$\begin{aligned}
\mathcal{E}_a(\sigma_q f)(t) &= \sum_{k\geq 0} \sum_{z_{i_1}\ldots z_{i_k}\in Z_0^k} \int_0^t \int_0^{\tau_k} \cdots \int_0^{\tau_2} \\
&\qquad [e^{\tau_1 z_0} z_{i_1} e^{(\tau_2-\tau_1)z_0} \ldots z_{i_k} e^{(t-\tau_k)z_0} * f]_{|q}\, d\xi_{z_{i_1}}(\tau_1)\ldots d\xi_{z_{i_k}}(\tau_k) \\
&= \sum_{k\geq 0} \sum_{u=z_{i_1}\ldots z_{i_k}\in Z_0^k} \int_0^t \int_0^{\tau_k} \cdots \int_0^{\tau_2} \\
&\qquad \kappa_k^u(t,\tau_1,\ldots,\tau_k)_{|q}\, d\xi_{z_{i_1}}(\tau_1)\ldots d\xi_{z_{i_k}}(\tau_k),
\end{aligned}$$

qu'on l'appelle **la série Volterra associée à** f ([B6], [LK]), avec les **noyaux de Volterra triangulaires** κ_k^u, pour $k \geq 0$ et $u \in Z_0^k$, définis comme suit :

$$\begin{aligned}
&\forall k \geq 0,\ \forall u = z_{i_1}\ldots z_{i_k} \in Z_0^k,\ 0 \leq \ldots \leq \tau_k \leq t, \\
\kappa_k^u(t,\tau_1,\ldots,\tau_k)_{|q} &= [e^{\tau_1 z_0} z_{i_1} e^{(\tau_2-\tau_1)z_0} \ldots z_{i_k} e^{(t-\tau_k)z_0} * f]_{|q}, \\
&= [e^{\tau_1 z_0} z_{i_1} e^{-\tau_1 z_0} e^{\tau_2 z_0} \ldots z_{i_k} e^{-\tau_k z_0} e^{t z_0} * f]_{|q}, \\
&= [(e^{ad_{\tau_1 z_0}} z_{i_1}) \ldots (e^{ad_{\tau_k z_0}} z_{i_k}) e^{t z_0} * f]_{|q}.
\end{aligned}$$

Ainsi en termes de dérivations de Lie, on retrouve les notations de M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue ([FLL]) :

$$\begin{aligned}
&\forall k \geq 0,\ \forall u = z_{i_1}\ldots z_{i_k} \in Z_0^k,\ 0 \leq \ldots \leq \tau_k \leq t, \\
\kappa_k^u(t,\tau_1,\ldots,\tau_k)_{|q} &= [e^{\tau_1 A_{z_0}} A_{z_{i_1}} e^{-\tau_1 A_{z_0}} e^{\tau_2 A_{z_0}} \ldots A_{z_{i_k}} e^{-\tau_k A_{z_0}} e^{t A_{z_0}} \circ f]_{|q} \\
&= [(e^{ad_{\tau_1 A_{z_0}}} A_{z_{i_1}}) \ldots (e^{ad_{\tau_k A_{z_0}}} A_{z_{i_k}}) e^{t A_{z_0}} \circ f]_{|q}.
\end{aligned}$$

Remarque : *Dans la seconde écriture les noyaux de Volterra κ_k^u, pour $k \geq 0$ et $u \in Z_0^k$, ne dépendent plus que des $\tau_1,\ \tau_2, \ldots,\ \tau_k$ et de t, et non plus de $\tau_2 - \tau_1, \ldots,\ t - \tau_k$ comme dans la première écriture. Cela a permis à F. Lamnabhi-Lagarrique d'obtenir facilement les expressions hamitoniennes des noyaux de Volterra dans l'étude de la commande optimale singulière ([L1]).*

Exemple : *Dans cet exemple, nous calculons l'approximation de Volterra de la sortie associée à la série formelle* $(z_0 + z)^*$.

D'après [SS], nous avons l'égalité entre suivante :

$$(z_0 + z)^* = z_0^*(zz_0^*)^*,$$

ou équivalemment :

$$(z_0 + z)^* = \sum_{k=0}^{K} z_0^*(zz_0^*)^k + \sum_{k \geq K+1} z_0^*(zz_0^*)^k.$$

Par conséquent, l'approximation de Volterra à l'ordre K *de la sortie associée à* $(z_0+z)^*$ *est l'Evaluation des* $K+1$ *premiers termes de l'expression précédente :*

$$\mathcal{E}_a((z_0 + z)^*) = \sum_{k=0}^{K} \mathcal{E}_a(z_0^*(zz_0^*)^k) + \ldots$$

Nous notons y_K, $(K \geq 0)$, *ces approximants :*

$$y_K = \sum_{k=0}^{K} \mathcal{E}_a(z_0^*(zz_0^*)^k).$$

D'une part, d'après le corollaire 2.5.2., nous avons :

$$\begin{aligned} y_K(t) &= \mathrm{e}^t \sum_{k=0}^{K} \int_0^t \int_0^{\tau_k} \int_0^{\tau_{k-1}} \cdots \int_0^{\tau_2} d\xi_z(\tau_1) \ldots d\xi_z(\tau_k) \\ &= \mathrm{e}^t \sum_{k=0}^{K} \mathcal{E}_a(z^k)(t) \\ &= \mathrm{e}^t \sum_{k=0}^{K} \frac{\xi_z^k(t)}{k!}, \end{aligned}$$

et les noyaux de Volterra valent tous e^t.

D'autre part, sachant que la série formelle $(z_0 + z)^*$ *est échangeable et* $(z_0 + z)^* = z_0^* \sqcup\!\sqcup\, z^*$, *alors d'après le théorème 2.4.4., nous avons :*

$$\mathcal{E}_a((z_0 + z)^*) = \mathrm{e}^t \mathrm{e}^{\xi_z(t)}.$$

Par conséquent, l'approximation de Volterra de la sortie associée à la série formelle $(z_0 + z)^*$ *revient donc à approximer la sortie* $y(t) = \mathrm{e}^t \mathrm{e}^{\xi_z(t)}$ *par les polynômes exponentiels* $\mathrm{e}^t \sum_{k=0}^{K} \frac{\xi_z^k(t)}{k!}$, $(K \geq 0)$. *Ce qui revient à approximer la série formelle* $(z_0 + z)^*$ *par les séries formelles* $z_0^* \sqcup\!\sqcup \sum_{k=0}^{K} z^k$, $(K \geq 0)$.

CHAPITRE 4

TRANSFORMATION D'EVALUATION ET EQUATIONS DIFFERENTIELLES NON LINEAIRES EN REGIME FORCE

4.0. Introduction

Certains systèmes dynamiques non linéaires sont décrits par les équations différentielles non linéaires en régime forcé de la forme :

$$y^{(n)}(t) + \alpha_1 y^{(n-1)}(t) + \ldots + \alpha_n y(t) + \sum_{i=2}^{d} \beta_i y^i(t) = u(t),$$

où u est l'entrée, y est la sortie, les $\alpha_1, \ldots, \alpha_n$ et les $\beta_2, \ldots, \beta_d$ sont des constantes et les conditions initiales sont supposées nulles pour simplifier l'écriture :

$$y^{(n)}(0) = y^{(n-1)}(0) = \ldots = y(0) = 0.$$

Les méthodes proposées pour calculer et analyser les signaux produits par les systèmes dynamiques non linéaires de ce type sont souvent basées sur les développements de Volterra ([BR1], [CN1], [CN2], [G2]). Les algorithmes correspondant à ces méthodes sont, en général, lourds et coûteux en temps de calcul, ils nécessitent des outils sophistiqués comme les fonctions transfert multidimensionnelles et l'association des variables.

Depuis leur introduction en théorie de la commande par M. Fliess, les séries formelles en variables non commutatives apparaissent comme un outil fécond pour résoudre des problèmes relatifs à des développements fonctionnels de la sortie des systèmes dynamiques non linéaires. Cela donne une justification théorique pour les méthodes heuristiques en Physique en utilisant le calcul de Heaviside dans le cadre non linéaire. Cette approche donne aussi une nouvelle méthode pour calculer itérativement une solution approchée de ces équations différentielles. La méthode proposée par M. Fliess ([F5]) consiste en une transformation ces équations différentielles en des équations intégrales de la forme :

$$y(t) + \alpha_1 \int_0^t y(\tau)d\tau + \ldots + \alpha_n \int_0^t \int_0^{\tau_1} \ldots \int_0^{\tau_n} y(\tau_n)d\tau_{n-1} \ldots d\tau_1 =$$

$$\int_0^t \int_0^{\tau_1} \ldots \int_0^{\tau_n} u(\tau_n)d\tau_{n-1} \ldots d\tau_1 - \sum_{i=2}^{d} \beta_i \int_0^t \int_0^{\tau_1} \ldots \int_0^{\tau_n} y^i(\tau_n)d\tau_{n-1} \ldots d\tau_1.$$

Ces équations intégrales sont codées par l'équation suivante (aux notations près) :

$$S + \alpha_1 S z_0 + \ldots + \alpha_n S z_0 = \left[G - \sum_{i=2}^{d} \beta_i S^{\sqcup\!\sqcup i} \right] z_0^n$$

$$\Longleftrightarrow \quad S = \left[G - \sum_{i=2}^{d} \beta_i S^{\sqcup\!\sqcup i} \right] z_0^n (1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1}.$$

Notons que si $\beta_2 = \ldots = \beta_d = 0$ on retrouve, à un changement variable près $(z_0 = \frac{1}{p} = s = \int_0^t . \, dt)$, la fonction transfert classique $(1 + \alpha_1 s + \ldots + \alpha_n s)^{-1}$.

Pour résoudre ces équations, M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue ont proposé le schéma suivant ([FLL]) :

$$\begin{cases} T_1 = G z_0^n (1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1} \\ T_k = \left[- \sum_{i=2}^{d} \beta_i \sum_{\substack{k_1, \ldots, k_i \\ k_1 + \ldots + k_i = k}} T_{k_1} \sqcup\!\sqcup \ldots \sqcup\!\sqcup T_{k_i} \right] z_0^n (1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1}, \end{cases}$$

et la série approximante à l'ordre k de l'équation précédente est la somme

$$T_1 + \ldots + T_k.$$

Exemple 4.0. : *Soit S une série formelle vérifiant l'équation polynômiale suivante :*

$$S = z_1 + (S \sqcup\!\sqcup S) z_0.$$

Alors le schéma précédent donne :

$$\begin{aligned} T_1 &= z_1, \\ T_2 = (T_1 \sqcup\!\sqcup T_1) z_0 &= 2 z_1^2 z_0, \\ T_3 = 2(T_1 \sqcup\!\sqcup T_2) z_0 &= 12 z_1^3 z_0^2 + 4 z_1^2 z_0 z_1 z_0, \\ T_4 = (2(T_1 \sqcup\!\sqcup T_3) + T_2 \sqcup\!\sqcup T_2) z_0 &= 24 (z_1^2 z_0)^2 z_0 + 144 z_1^4 z_0^3 + 24 z_1^3 z_0^2 z_1 z_0 \\ &\quad + 24 z_1^3 z_0 z_1 z_0^2 + 48 z_1^2 (z_1 z_0)^2 z_0 + 8 z_1^2 (z_0 z_1)^2 z_0, \\ &\vdots \end{aligned}$$

Par conséquent, les approximants correspondants sont :

$$\begin{aligned} F_1 = T_1 &= z_1, \\ F_2 = T_2 + T_1 &= 2 z_1^2 z_0 + z_1, \\ F_3 = T_3 + T_2 + T_1 &= 12 z_1^3 z_0^2 + 4 z_1^2 z_0 z_1 z_0 + 2 z_1^2 z_0 + z_1, \\ F_4 = T_4 + T_3 + T_2 + T_1 &= 24 (z_1^2 z_0)^2 z_0 + 144 z_1^4 z_0^3 + 24 z_1^3 z_0^2 z_1 z_0 \\ &\quad + 24 z_1^3 z_0 z_1 z_0^2 + 48 z_1^2 (z_1 z_0)^2 z_0 + 8 z_1^2 (z_0 z_1)^2 z_0 \\ &\quad + 12 z_1^3 z_0^2 + 4 z_1^2 z_0 z_1 z_0 + 2 z_1^2 z_0 + z_1, \\ &\vdots \end{aligned}$$

Depuis 1980, M. Lamnabhi ([L2]) et F. Lamnabhi-Lagarrigue ([L1]) ont développé et approfondi cette méthode. Ils l'ont appliquée à l'analyse du comportement d'entrée/sortie des circuits électroniques non linéaires. Ces circuits peuvent comporter, en plus des éléments linéaires usuels, d'autres non linéaires tels que des résistances, des capacités, des inductances ou encore des sources de tension liées. Ils ont montré sur les exemples pratiques comment peut s'effectuer systématiquement l'analyse transitoire ou permanente lorsque les entrées typiques (harmoniques, Dirac, échelons,...) sont introduites ou encore comment déduire des propriétés statistiques en présence d'une entrée aléatoire. Dans le cas de l'analyse transitoire, la méthode proposée est une alternative aux techniques d'association de variables de D. George ([G2]). Ces études sont la base du projet FANEC - Functional Analysis of Nonlinear Electronic Circuits.

Il est à noter que le schéma itératif proposé par M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue ([FLL]), nécessite, dans le cas d'une implantation pour un traitement automatique (voir projet FANEC), de garder en mémoire les résultats intermédiaires ($\{T_k\}_{k\geq 1}$), d'où consommation supplémentaire de mémoire. Ce schéma est relativement coûteux en temps machine car il utilise le produit de mélange dont on remarquera la complexité pour calculer les T_k à chaque itération. Ainsi les calculs effectués ont montré qu'on se heurte très vite à des difficulté de programmation avec un temps de calcul et une place de mémoire importants pour obtenir les ordres élevés dès que la structure des équations devient compliquée. Cela est dû à la complexité croissante exponentiellement du produit de mélange. Le calcul de ce produit est encore mal maîtrisé jusqu'à cette date. Il est ignoré en général par les machines symboliques actuellement sur le marché.

Très récemment, G.Viennot et P. Leroux ont donné une interprétation purement combinatoire de la structure des équations en utilisant des "arbres croissants", des "chemins de Motzkin" et des "histoires" ([VL]).

Il est à noter aussi que cette méthode conduit ([FLL], [L1], [L2]) à des combinaisons linéaires de fractions rationnelles non commutatives de la forme :

$$(c_0z_0)^{*p_0}z_{i_1}(c_1z_0)^{*p_1}z_{i_2}(c_2z_0)^{*p_2}\dots z_{i_{k-1}}(c_{k-1}z_0)^{*p_{k-1}}z_{i_k}(c_kz_0)^{*p_k},$$

où les $p_0,\dots,p_k$ sont des entiers positifs, les $c_0,\dots,c_k$ sont des nombres complexes et les $z_{i_0},\dots,z_{i_k}$ sont des lettres de l'alphabet de codage Z. Ces fractions rationnelles sont complètement intégrables (voir chapitre II). Nous donnerons au chapitre VI une représentation arborescente et une implantation récursive en MACSYMA de la transformation d'Evaluation.

En fait, la transformation d'Evaluation peut contribuer aussi à l'étude de ces équations différentielles en donnant une méthode directe et efficace de programmation, que nous allons montrer dans ce chapitre.

Reprenons donc l'équation symbolique :

$$S = \left[G - \sum_{i=2}^{d} \beta_i S^{ш\, i}\right] z_0^n (1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1}.$$

En prenant l'Evaluation les deux membres on a :

$$\begin{aligned}\mathcal{E}_a(S)(t) &= \mathcal{E}_a(G z_0^n (1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1})(t) \\ &\quad - \mathcal{E}_a\left(\sum_{i=2}^{d} \beta_i S^{ш\, i} z_0^n (1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1}\right)(t) \\ &= \int_0^t \mathcal{E}_a(G)(\tau) h(t-\tau) d\tau - \int_0^t \phi(\mathcal{E}_a(S)(\tau)) h(t-\tau) d\tau,\end{aligned}$$

où $\phi(f)$ est le polynôme $\beta_2 f^2 + \ldots + \beta_d f^d$ et $h(t)$ est l'Evaluation de la série $z_0^{n-1}(1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1}$. On se ramène donc à une équation intégrale du type :

$$s(t) = f(t) + \int_0^t \phi(s(\tau)) h(t-\tau) d\tau$$

dont une méthode de résolution itérative serait :

$$\begin{cases} s_1(t) = f(t) \\ s_k(t) = s_1(t) + \displaystyle\int_0^t \phi(s_{k-1}(\tau)) h(t-\tau) d\tau. \end{cases}$$

A ce schéma itératif correspond un schéma itératif sur les séries génératrices codant cette équation (on peut remarquer la ressemblance avec celui proposé par M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue) :

$$\begin{cases} S_1 = G z_0^n (1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1} \\ S_k = S_1 + \left[\displaystyle\sum_{i=2}^{d} \beta_i S_{k-1}^{ш\, i}\right] z_0^n (1 + \alpha_1 z_0 + \ldots + \alpha_n z_0)^{-1}. \end{cases}$$

On prouvera sa convergence (relativement à la topologie discrète sur les séries formelles), on précisera le nombre de "termes exacts" obtenus à chaque itération et on terminera par l'examination des exemples suivants :

$$\begin{aligned} \dot{y}(t) - y^2(t) &= a^z(t) \quad \text{avec} \quad \dot{y}(0) = y(0) = 0 \\ \dot{y}(t) + y(t) + y^2(t) &= a^z(t) \quad \text{avec} \quad \dot{y}(0) = y(0) = 0 \\ \ddot{y}(t) + \dot{y}(t) + y(t) + y^3(t) &= a^z(t) \quad \text{avec} \quad \ddot{y}(0) = \dot{y}(0) = y(0) = 0. \end{aligned}$$

4.1. Une méthode formelle

Soit $\Phi(S)$ un polynôme commutatif de degré $d \geq 2$ en la série formelle S défini comme suit :

$$\Phi(S) = \beta_2 S^{ш 2} + \ldots + \beta_d S^{ш d},$$

où les $\beta_2, \ldots, \beta_d$ sont des nombres complexes.

Remarque :

(i) Si S est une série formelle propre l'ordre s, puisque l'ordre du polynôme $\Phi(S)$ est 2, alors l'ordre $\Phi(S)$ est supérieure ou égale à $2s$.

(ii) Soient S et T deux séries formelles, nous avons :

$$\Phi(S) - \Phi(T) = \sum_{i=2}^{d} \beta_i (S^{ш i} - T^{ш i}) = (S - T) \sum_{i=2}^{d} \beta_i \sum_{j=0}^{i-1} S^{ш j} T^{ш i-1-j}.$$

Par conséquent :

$$\omega(\Phi(S) - \Phi(T)) \geq \omega(S - T) + \inf(\omega(S), \omega(T)).$$

4.1.1. Une équation algébrique (EQ)

Etant donné une lettre z de Z, on note R une série formelle en une seule lettre z et d'ordre $\rho \geq 1$. Soit G une série formelle d'ordre $\gamma \geq 0$. Les séries R et G sont supposées vérifier les conditions du théorème 2.4.1.

Soit S une série formelle vérifiant l'équation polynômiale suivante :

$$(EQ) \quad S + \alpha_1 S z + \alpha_2 S z^2 + \ldots + \alpha_n S z^n = G + \Phi(S) R,$$

ou encore $SK = G + \Phi(S)R$, où K est la série formelle définie par :

$$K = \sum_{k=0}^{n} \alpha_k z^k \quad \text{avec} \quad \alpha_0 = 1.$$

Puisque le terme constant $< K|\epsilon > = \alpha_0 = 1$ ne s'annule pas, alors la série formelle K^{-1} existe, K^{-1} est une série formelle en une variable commutative z. La série formelle S peut alors s'écrire :

$$S = GK^{-1} + \Phi(S) R K^{-1},$$

c'est une équation algébrique en S.

Supposons que K admette r racines complexes distinctes $\mu_1, \ldots, \mu_r$ d'ordre de multiplicité $m_1, \ldots, m_r$ respectivement $\left(\sum_{l=1}^{r} m_l = n\right)$. On peut décomposer d'une manière unique K^{-1} en éléments simples :

$$K^{-1} = \frac{1}{\prod_{l=1}^{r} (z - \mu_l)^{m_l}} = \sum_{l=1}^{r} \sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} H_{l,k},$$

où pour tout $l \in [1..r]$ et pour tout $k \in [1..m_l]$, $\lambda_{l,k} \in \mathbb{C}$, et chaque $H_{l,k}$ est de la forme $\left(\frac{z}{\mu_l}\right)^{*k}$. Ainsi GK^{-1} s'écrit sous la forme $\sum_{l=1}^{r} \sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} G \left(\frac{z}{\mu_l}\right)^{*k}$.

4.1.2. Un schéma de résolution itératif (IS)

Considérons le schéma itératif suivant pour calculer la solution $\hat{S}$ de l'équation (EQ) :

$$(IS)\quad \begin{cases} S_1 = GK^{-1} \\ S_k = GK^{-1} + \Phi(S_{k-1})RK^{-1}. \end{cases}$$

Lemme 4.1.2.1. : *Nous avons pour tout entier $k \geq 1$:*

$$\omega(S_k) \geq \gamma.$$

<u>*Preuve*</u> :

. Puisque $S_1 = GK^{-1}$, le résultat est immédiat pour $k = 1$.

. Supposons que le résultat soit vrai pour tout κ, $1 \leq \kappa \leq k$.

. Pour $\kappa = k + 1$, puisque $S_{k+1} = GK^{-1} + \Phi(S_k)RK^{-1}$, alors il est clair que :

$$\omega(S_{k+1}) \geq \inf(\gamma, \omega(\Phi(S_k)) + \rho).$$

D'après la remarque (ii), on a $\omega(\Phi(S_k)) \geq 2\omega(S_k)$. Par conséquent, d'après l'hypothèse de récurrence, on a $\omega(S_{k+1}) \geq \gamma$ •

Lemme 4.1.2.2. : *Nous avons pour tout entier $k \geq 2$:*

$$\omega(S_{k+1} - S_k) \geq \omega(S_k - S_{k-1}) + \gamma + \rho.$$

<u>*Preuve*</u> : En effet, puisque $S_{k+1} - S_k = [\Phi(S_k) - \Phi(S_{k-1})]RK^{-1}$, alors d'après la remarque (ii) et le lemme 4.1.2.1., on a :

$$\omega(S_{k+1} - S_k) \geq \omega(S_k - S_{k-1}) + \omega(S_{k-1}) + \rho \geq \omega(S_k - S_{k-1}) + \gamma + \rho \quad \bullet$$

Ce lemme exprime qu'à chaque étape, on ajoute au moins $\gamma + \rho$ termes exacts à l'étape précédente.

Lemme 4.1.2.3. : *Nous avons pour tout entier $k \geq 1$:*

$$\omega(S_{k+1} - S_k) \geq (k+1)\gamma + k\rho.$$

<u>*Preuve*</u> :

. Nous montrons aisément que pour $k = 1$, puisque $\omega(S_1) \geq \gamma$, alors d'après la remarque (i), nous avons $\omega(\Phi(S_1)) \geq 2\gamma$. Par conséquent :

$$\omega(S_2 - S_1) = \omega(\Phi(S_1)RK^{-1}) \geq 2\gamma + \rho.$$

. Supposons que le résultat soit vrai pour tout κ, $1 \leq \kappa \leq k$.

. Pour $\kappa = k + 1$, d'après le lemme 4.1.2.2. et d'après l'hypothèse de récurrence, on a :

$$\omega(S_{k+2} - S_{k+1}) \geq [(k+1)\gamma + k\rho] + \gamma + \rho = (k+2)\gamma + (k+1)\rho \quad \bullet$$

Théorème 4.1.2.1. : *La suite* $\{S_k\}_{k\geq 1}$ *converge vers la solution* $\hat{S}$ *de l'équation* (EQ).

<u>*Preuve*</u> : Pour tous entiers $k, l \geq 1$, nous avons :

$$\begin{aligned}\omega(S_{k+l} - S_k) &= \omega(S_{k+l} - S_{k+l-1} + S_{k+l-1} - \ldots - S_{k+1} + S_{k+1} - S_k)\\ &\geq \inf[\omega(S_{k+l} - S_{k+l-1}), \ldots, \omega(S_{k+1} - S_k)]\\ &\geq \omega(S_{k+1} - S_k) \qquad \text{(lemme 4.1.2.2.)}\\ &\geq (k+1)\gamma + k\rho \qquad \text{(lemme 4.1.2.3.).}\end{aligned}$$

Or $\rho \geq 1$, donc la suite $\{S_k\}_{k\geq 1}$ est une suite de Cauchy. Puisque $A \ll Z \gg$ est complet, alors la suite $\{S_k\}_{k\geq 1}$ converge vers la solution $\hat{S}$ de l'équation (EQ) •

Corollaire 4.1.2.1. : *Pour tout entier* $k \geq 1$, *la série formelle* S_k *coïncide avec la solution* $\hat{S}$ *de l'équation* (EQ) *jusqu'à l'ordre* $(k+1)\gamma + k\rho - 1$:

$$\forall w \in Z^*, \quad |w| \leq (k+1)\gamma + k\rho - 1 \quad \Rightarrow \quad < S_k|w> = < \hat{S}|w>,$$

<u>*Preuve*</u> : Si $\hat{S}$ la solution de l'équation (EQ) alors :

$$\hat{S} = GK^{-1} + \Phi(\hat{S})RK^{-1}.$$

Par conséquent :

$$\omega(\hat{S} - S_k) \geq \omega(\hat{S} - S_{k-1}) + \gamma + \rho.$$

D'où par récurrence :

$$\omega(\hat{S} - S_k) \geq (k+1)\gamma + k\rho \quad \bullet$$

Remarque 4.1.1. : *Comme pour le schéma itératif proposé par M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue, ce schéma conduit à mélanger des combinaisons linéaires des fractions rationnelles non commutatives de la forme :*

$$(c_0 z_{j_0})^{*p_0} z_{i_1} (c_1 z_{j_1})^{*p_1} z_{i_2} (c_2 z_{j_2})^{*p_2} \ldots z_{i_{k-1}} (c_{k-1} z_{j_{k-1}})^{*p_{k-1}} z_{i_k} (c_k z_{j_k})^{*p_k},$$

où les $p_0, \ldots, p_k$ *sont des entiers positifs, les* $c_0, \ldots, c_k$ *sont des nombres complexes et les* $z_{j_0}, \ldots, z_{j_k}$ *et* $z_{i_0}, \ldots, z_{i_k}$ *sont des lettres de l'alphabet de codage* Z. *Mais il est aussi intéressant de remarqer que ce schéma fait intervenir des puissances de mélange de séries formelles. Ainsi pour calculer* $(uz)^{ш n}$ *on aura besoin d'effectuer une seule fois* $[(uz)^{ш n-1} ш u]z$, *puis multiplier par* n *car* $(uz)^{ш n} = n[(uz)^{ш n-1} ш u]z$ *(voir lemme 1.4.3.1., cette technique est en cours d'implantation par N.E. Oussous).*

Pour le schéma (IS), *on connaît exactement le début de la solution de l'équation* (EQ). *On peut dès lors utiliser la technique développée par G. Jacob et C. Hespel qui consiste à prolonger un polynôme par une série rationnelle admettant ce polynôme comme début. Dans cette technique, les auteurs donnent l'automate minimal reconnaissant cette série, et la représentation d'état minimale. A partir de l'automate minimal on peut obtenir une expression rationnelle de la série reconnue, grâce à une résolution du système d'équations linéaires, bien connue en informatique théorique.*

Exemple 4.1.1. : *Reprenons l'équation différentielle de l'exemple 4.0. Alors le schéma suivant*

$$\begin{cases} S_1 = z_1 \\ S_k = z_1 + (S_{k-1} ш S_{k-1}) z_0 \end{cases}$$

donne une suite d'approximants de la solution $\hat{S}$:

$$\begin{aligned} S_1 &= z_1, \\ S_2 &= 2z_1^2 z_0 + z_1, \\ S_3 &= 8(z_1^2 z_0)^2 z_0 + 48 z_1^4 z_0^3 + 24 z_1^2 (z_1 z_0)^2 z_0 \\ &\quad + 12 z_1^3 z_0^2 + 4 z_1^2 z_0 z_1 z_0 + 2 z_1^2 z_0 + z_1, \\ &\vdots \end{aligned}$$

Pour cet exemple, nous avons $\gamma = \rho = 1$. *D'après le corollaire 4.1.2.1., nous savons que les termes jusqu'au degré 6* $(= (3+1)+3-1)$ *de* S_3 *sont les termes de la série* $\hat{S}$, *c'est à dire le polynôme* $12z_1^3 z_0^2 + 4z_1^2 z_0 z_1 z_0 + 2z_1^2 z_0 + z_1$ *figure dans* $\hat{S}$. *Ce polynôme figure dans* F_3 *et dans* F_4 *de l'exemple 4.0. Le support du polynôme* $8(z_1^2 z_0)^2 z_0 + 48 z_1^4 z_0^3 + 24 z_1^2 (z_1 z_0)^2 z_0$ *figurant dans* S_3 *est inclus dans le support de* F_4.

Ainsi pour une équation différentielle

$$\dot{y}(t) = u(t) + y^2(t) \quad \text{avec} \quad \dot{y}(0) = y(0) = 0,$$

les approximants (paramétrés par $\xi_{z_1}(t) = \int_0^t u(\tau) d\tau$*) correspondants sont (voir chapitre II) :*

$$\begin{aligned} y_1(t) &= \xi_{z_1}(t), \\ y_2(t) &= \int_0^t \xi_{z_1}^2(\tau) d\tau + \xi_{z_1}(t), \\ y_3(t) &= 4 \int_0^t \int_0^{\tau_1} \int_0^{\tau_2} \xi_{z_1}^2(\tau_3)(t - \tau_1) d\tau_3 d\xi_{z_1}(\tau_2) d\tau_1 \\ &\quad + \int_0^t \xi_{z_1}^4(\tau)(t-\tau)^2 d\tau + 4 \int_0^t \int_0^{\tau_1} \int_0^{\tau_2} \xi_{z_1}^3(\tau_3) d\tau_3 d\xi_{z_1}(\tau_2) d\tau_1 \\ &\quad + 2 \int_0^t \xi_{z_1}^3(\tau)(t-\tau) d\tau + \int_0^t \int_0^{\tau_1} \int_0^{\tau_2} \xi_{z_1}^2(\tau_3) d\tau_3 d\xi_{z_1}(\tau_2) d\tau_1 \\ &\quad + \int_0^t \xi_{z_1}^2(\tau) d\tau + \xi_{z_1}(t), \\ &\vdots \end{aligned}$$

On peut observer que S_3 *est de degré 7, mais on n'aura besoin en fait que des intégrales triples. Cette diminution du nombre des intégrales à effectuer est dûe au théorème de convolution (voir chapitre II, ce théorème est largement exploité au chapitre VI pour implanter la transformation d'Evaluation).*

4.1.3. Evaluation du schéma itératif (IS)

Puisque le produit de mélange est coûteux en mémoire et en temps machine, nous allons donner une méthode directe pour calculer des approximants de la solution des équations différentielles non linéaires en régime forcé en termes d'Evaluation du schéma itératif (IS). Cette méthode exploite les capacités d'intégration, de sommation et de dérivation formelle du système de calcul formel MACSYMA (voir chapitre VI) :

Soit $\phi(f)$ un polynôme (en l'indéterminée f) :

$$\phi(f) = \beta_2 f^2 + \ldots + \beta_d f^d.$$

Alors d'après les théorèmes 2.4.3. et 2.4.4., nous avons :

$$\mathcal{E}_a(\Phi(S)) = \phi(\mathcal{E}_a(S)).$$

D'après les théorèmes 2.4.2., 2.4.3. et 2.4.4., le schéma itératif (IS) peut s'écrire :

$$(EIS) \quad \begin{cases} \mathcal{E}_a(S_1) = \mathcal{E}_a(GK^{-1}) \\ \mathcal{E}_a(S_k) = \mathcal{E}_a(GK^{-1}) + \mathcal{E}_a(\mathcal{E}_a(\Phi(S_{k-1})); RK^{-1}). \end{cases}$$

Pour tout entier $k \geq 1$, on note y_k l'Evaluation de la série S_k, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ relative à l'alphabet fini Z :

$$y_k = \mathcal{E}_a(S_k).$$

Nous obtenons ainsi :

$$(KIS) \quad \begin{cases} y_1 = \mathcal{E}_a(GK^{-1}) \\ y_k = y_1 + \mathcal{E}_a(\phi(y_{k-1}); RK^{-1}). \end{cases}$$

4.1.3.1. Calcul de y_1

Rappelons que :

$$GK^{-1} = \sum_{l=1}^{r} \sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} G\left(\frac{z}{\mu_l}\right)^{*k}.$$

Notons $h_{l,k}(\xi_z(t))$ l'Evaluation de $\left(\frac{z}{\mu_l}\right)^{*k}$:

$$h_{l,k}(\xi_z(t)) = \exp\left(\frac{\xi_z(t)}{\mu_l}\right) \sum_{j=0}^{k-1} \binom{k-1}{j} \frac{1}{j!} \left(\frac{\xi_z(t)}{\mu_l}\right)^j \qquad \text{(proposition 2.5.2.).}$$

D'après le théorème 2.4.3. et le théorème 2.4.5., nous obtenons :

$$y_1(t) = \sum_{l=1}^{r} \sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} \left[< G|\epsilon > h_{l,k}(\xi_z(t)) + \int_0^t h_{l,k}(\xi_z(t) - \xi_z(\tau)) d\mathcal{E}_a(G)(\tau) \right].$$

Signalons que dans le cas particulier $z = z_0$, le calcul précédent correspond mutatis mutandis à la transformation de Laplace inverse dans l'étude des systèmes dynamiques décrits par une équation différentielle linéaire de degré n.

4.1.3.2. Calcul de y_k

Puisque $\omega(R) = \rho \geq 1$, nous remarquons que $R = zR_1$ ($\omega(R_1) \geq 0$). Nous notons $\theta(\xi_z(t))$ l'Evaluation de la série formelle R_1K^{-1}, pour l'entrée $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m}\,)$ relative à l'alphabet fini Z. Avec le même calcul que précédemment, on a :

$$\theta(\xi_z(t)) = \mathcal{E}_a(R_1K^{-1})(t) = \sum_{l=1}^{r}\sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k}\int_0^t h_{l,k}(\xi_z(t) - \xi_z(\tau))d\mathcal{E}_a(R_1)(\tau).$$

Avec ces notations, d'après le théorème 2.4.2., $\mathcal{E}_a(\phi(y_{k-1}); RK^{-1})$ peut s'écrire :

$$\mathcal{E}_a(\phi(y_{k-1}); RK^{-1}) = \mathcal{E}_a(\mathcal{E}_a(\phi(y_{k-1}); z); R_1K^{-1}).$$

Par conséquent, d'après le théorème de convolution, (KIS) devient :

$$(KIS) \quad \begin{cases} y_1(t) = \mathcal{E}_a(GK^{-1})(t) \\ y_k(t) = y_1(t) + \displaystyle\int_0^t \phi(y_{k-1}(\tau))\theta(\xi_z(t) - \xi_z(\tau))d\xi_z(\tau). \end{cases}$$

Remarque 4.1.2. :

(i) Les noyaux $\{\phi(y_{k-1})\}_{k\geq 0}$ jouent ici un rôle analogue à celui des noyaux de Volterra.

(ii) On peut appliquer les mêmes techniques pour prouver la convergence relativement à la topologie discrète sur les séries formelles du schéma itératif proposé par M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue. En prenant l'Evaluation de ce schéma, on obtient le schéma suivant :

$$\begin{cases} f_1(t) = \displaystyle\int_0^t \mathcal{E}_a(G)(\tau)h(t-\tau)d\tau \\ f_k(t) = -\displaystyle\sum_{i=2}^{d}\beta_i \sum_{\substack{k_1,\ldots,k_i \\ k_1+\ldots+k_i=k}} \int_0^t f_{k_1}(\tau)\ldots f_{k_i}(\tau)h(t-\tau)d\tau, \end{cases}$$

et l'approximant d'ordre k de la sortie est :

$$f_1(t) + \ldots + f_k(t)$$

correspondante ($h(t)$ est l'Evaluation de la série $z_0^{n-1}(1+\alpha_1 z_0+\ldots+\alpha_n z_0)^{-1}$).

(iii) Le schéma précédent et le schéma (KIS) correspondent aussi à des méthodes de résolution des équations intégrales du type :

$$s(t) = f(t) + \int_0^t \phi(s(\tau))\theta(\xi(t) - \xi(\tau))d\xi(\tau).$$

4.2. Applications

Les exemples suivants sont obtenus à l'aide du système de calcul formel MACSYMA version 309.6 sur SUN-3, Release 309.6 (il est possible qu'il y ait des erreurs dues à la transcription en format de T_EX).

Exemple 4.2.1.

Reprenons l'équation différentielle de l'exemple 4.1.1. :

$$\dot{y}(t) = a^z(t) + y^2(t) \quad \text{avec} \quad \dot{y}(0) = y(0) = 0,$$

ou équivalemment :

$$y(t) = \int_0^t a^z(\tau)d\tau + \int_0^t y^2(\tau)d\tau.$$

Si S est une série telle que $\mathcal{E}_a(S)(t) = y(t)$, alors nous avons :

$$S = z + (S \sqcup\!\sqcup S)z_0.$$

D'après ce qui précède, la suite $\{y_k\}_{k\geq 0}$ suivante converge vers la solution de cette équation différentielle :

$$\begin{cases} y_1(t) = \displaystyle\int_0^t a^z(\tau)d\tau \\ y_k(t) = y_1(t) + \displaystyle\int_0^t y_{k-1}^2(\tau)d\tau. \end{cases}$$

Pour $a^z = un$, nous obtenons les résultats suivants :

$$\begin{aligned} y_1(t) &= t, \\ y_2(t) &= \frac{t^3+3t}{3} \\ y_3(t) &= \frac{5t^7+42t^5+105t^3+315t}{315}, \\ y_4(t) &= (715t^{15}+13860t^{13}+109746t^{11}+570570t^9+2297295t^7 \\ &\quad +5675670t^5+14189175t^3+42567525t)/42567525). \end{aligned}$$

Pour $a^z(t) = t$, nous obtenons les résultats suivants :

$$\begin{aligned} y_1(t) &= \frac{t^2}{2}, \\ y_2(t) &= \frac{t^5+10t^2}{20}, \\ y_3(t) &= \frac{2t^{11}+55t^8+440t^5+4400t^2}{8800}, \\ y_4(t) &= (476t^{23}+30107t^{20}+770385t^{17}+12903000^{14}+168599200t^{11} \\ &\quad +1324708000t^8+10597664000t^5+105976640000t^2)/211953280000). \end{aligned}$$

Pour $a^z(t) = \mathrm{e}^t$, nous obtenons les résultats suivants :

$$
\begin{aligned}
y_1(t) &= \mathrm{e}^t - 1,\\
y_2(t) &= \frac{\mathrm{e}^{2t} - 2\mathrm{e}^t + 2t + 1}{2},\\
y_3(t) &= (3\mathrm{e}^{4t} - 16\mathrm{e}^{3t} + (24t + 24)\mathrm{e}^{2t} + (96t - 96)\mathrm{e}^t\\
&\quad + 16t^3 + 24t^2 + 12t - 107)/48,\\
y_4(t) &= (42525\mathrm{e}^{8t} - 518400\mathrm{e}^{7t} + (907200t + 2368800)\mathrm{e}^{6t}\\
&\quad + (580608 - 10160640t)\mathrm{e}^{5t} + (907200t^3 + 6123600t^2 + 37535400t\\
&\quad - 39037950)\mathrm{e}^{4t} + (-6451200t^3 - 61286400t^2 + 36019200t\\
&\quad + 89196800)\mathrm{e}^{3t} + (14515200t^4 + 7257600t^3 + 195955200t^2\\
&\quad - 630504000t + 392364000)\mathrm{e}^{2t} + (-116121600t^4 + 406425600t^3\\
&\quad - 1132185600t^2 + 3128025600t - 3817497600)\mathrm{e}^t + 1382400t^7\\
&\quad + 4838400t^6 + 7257600t^5 - 26913600t^4 - 62899200t^3 - 48535200t^2\\
&\quad + 432772200t + 3372501217)/87091200).
\end{aligned}
$$

Pour $a^z(t) = \cos(t)$, nous obtenons les résultats suivants :

$$
\begin{aligned}
y_1(t) &= \sin(t),\\
y_2(t) &= \frac{\sin(2t) - 4\sin(t) - 2t}{4},\\
y_3(t) &= -(3\sin(4t) - 32\sin(3t) + 120\sin(2t) - 48t\cos(2t) - 672\sin(t)\\
&\quad + 384t\cos(t) - 32t^3 - 204t)/384,\\
y_4(t) &= -(42525\sin(8t) - 1036800\sin(7t) + 11289600\sin(6t)\\
&\quad - 1814400t\cos(6t) - 96671232\sin(5t) + 40642560t\cos(5t)\\
&\quad + (635947200 - 19051200t^2)\sin(4t)\\
&\quad + (-3628800t^3 - 373766400t)\cos(4t)\\
&\quad + (412876800t^2 - 2850444800)\sin(3t)\\
&\quad + (51609600t^3 + 2607360000t)\cos(3t)\\
&\quad + (-116121600t^4 - 2743372800t^2 + 15117580800)\sin(2t)\\
&\quad + (-522547200t^3 - 14818204800t)\cos(2t)\\
&\quad + (1857945600t^4 - 18811699200t^2 + 10980748800)\sin(t)\\
&\quad + (10683187200t^3 - 15734476800t)\cos(t) - 11059200t^7 - 197406720t^5\\
&\quad - 2935699200t^3 - 17653242600t)/11147673600.
\end{aligned}
$$

D'autre part, l'exemple 4.0. nous donne :

$$
\begin{aligned}
F_3 &= 12z_1^3z_0^2 + 4z_1^2z_0z_1z_0 + 2z_1^2z_0 + z_1,\\
F_4 &= 24(z_1^2z_0)^2z_0 + 144z_1^4z_0^3 + 24z_1^3z_0^2z_1z_0 + 24z_1^3z_0z_1z_0^2 + 48z_1^2(z_1z_0)^2z_0\\
&\quad + 8z_1^2(z_0z_1)^2z_0 + 12z_1^3z_0^2 + 4z_1^2z_0z_1z_0 + 2z_1^2z_0 + z_1.
\end{aligned}
$$

Et les Evaluations des séries F_3 et F_4 pour $a^z = un$ donne :

$$\mathcal{E}_a(F_3)(t) = \frac{t(2t^4+5t^2+15)}{15},$$
$$\mathcal{E}_a(F_4)(t) = \frac{t(17t^6+42t^4+105t^2+315)}{315}.$$

Et les Evaluations des séries F_3 et F_4 pour $a^z(t) = t$ donne :

$$\mathcal{E}_a(F_3)(t) = \frac{t^2(t^6+8t^3+80)}{160}$$
$$\mathcal{E}_a(F_4)(t) = \frac{t^2(7t^9+55t^6+440t^3+4400)}{8800}.$$

Et les Evaluations des séries F_3 et F_4 pour $a^z(t) = \mathrm{e}^t$ donne :

$$\begin{aligned}\mathcal{E}_a(F_3)(t) &= \frac{\mathrm{e}^{3t}-6\mathrm{e}^{2t}+6t\mathrm{e}^t+12\mathrm{e}^t-3t^2-6t-7}{3},\\ \mathcal{E}_a(F_4)(t) &= (33\mathrm{e}^{4t}-320\mathrm{e}^{3t}+360t\mathrm{e}^{2t}+1008\mathrm{e}^{2t}-288t^2\mathrm{e}^t-1152t\mathrm{e}^t\\ &\quad -1104\mathrm{e}^t+144t^3+504t^2+852t+383)/144).\end{aligned}$$

Et les Evaluations des séries F_3 et F_4 pour $a^z(t) = \cos(t)$ donne :

$$\begin{aligned}\mathcal{E}_a(F_3)(t) &= (\sin(3t)-2t\cos(3t)-3\sin(2t)+21\sin(t)+8t\cos^3(t)\\ &\quad -18t\cos(t)+6t)/12)\\ \mathcal{E}_a(F_4)(t) &= -(11\sin(4t)-32\sin(3t)+64t\cos(3t)+344\sin(2t)\\ &\quad -240t\cos(2t)-672\sin(t)-256t\cos 3^(t)+576t\cos(t)\\ &\quad -32t^3-492t)/384.\end{aligned}$$

Le tableau suivant donne une idée sur la complexité de deux schémas :

	un	t	e^t	$\cos(t)$
y_2	1.5 s	3 s	7 s	8 s
y_3	6 s	6 s	21 s	31 s
y_4	12 s	13 s	77 s	167 s
$\mathcal{E}_a(F_3)$	17 s	18 s	85 s	97 s
$\mathcal{E}_a(F_4)$	74 s	71 s	670 s	672 s

(Nous n'avons pas comptabiliser le temps pour obtenir les séries F_3 et F_4 qui est de l'ordre 1 et 7 secondes en utilisant les programmes du mélange fournis par N.E. Oussous).

Exemple 4.2.2.

Considérons l'équation différentielle non linéaire en régime forcé suivante :

$$\dot{y}(t) + y(t) + y^2(t) = a^z(t) \quad \text{avec} \quad \dot{y}(0) = y(0) = 0,$$

ou équivalemment :

$$y(t) + \int_0^t y(\tau)d\tau = \int_0^t a^z(\tau)d\tau - \int_0^t y^2(\tau)d\tau.$$

Si S est une série telle que $\mathcal{E}_a(S)(t) = y(t)$, alors nous avons :

$$S(1 + z_0) = z - (S \sqcup\!\sqcup S)z_0.$$

Dans ce cas, nous avons :

$$S = z(-z_0)^* - (S \sqcup\!\sqcup S)z_0(-z_0)^*.$$

D'après ce qui précède, la suite $\{y_k\}_{k \geq 0}$ suivante converge vers la solution de cette équation différentielle :

$$\begin{cases} y_1(t) = \displaystyle\int_0^t a^z(\tau)\exp(\tau - t)d\tau \\ y_k(t) = y_1(t) + \displaystyle\int_0^t [-y_{k-1}^2(\tau)]\exp(\tau - t)d\tau. \end{cases}$$

Pour $a^z = un$, nous obtenons les résultats suivants :

$$\begin{aligned}
&y_1(t) = 1 - \mathrm{e}^{-t}, \\
&y_2(t) = \mathrm{e}^{-2t}((2t - 1)\mathrm{e}^t + 1), \\
&y_3(t) = \frac{\mathrm{e}^{-4t}(3\mathrm{e}^{4t} - 19\mathrm{e}^{3t} + (12t^2 + 12t + 15)\mathrm{e}^{2t} + 6t\mathrm{e}^t + 1)}{3}, \\
&y_4(t) = \mathrm{e}^{-8t}((1436400t - 4354267)\mathrm{e}^{7t} + (907200t^2 + 2721600t + 8404200)\mathrm{e}^{6t} \\
&\quad + (-2872800t^2 - 5518800t - 6350400)\mathrm{e}^{5t} \\
&\quad + (604800t^4 + 2016000t^3 + 4132800t^2 + 3309600t + 2073400)\mathrm{e}^{4t} \\
&\quad + (453600t^3 + 793800t^2 + 963900t + 121275)\mathrm{e}^{3t} \\
&\quad + (151200t^2 + 120960t + 99792)\mathrm{e}^{2t} + (25200t + 4200)\mathrm{e}^t + 1800) \\
&\quad /113400).
\end{aligned}$$

Pour $a^z(t) = t$, nous obtenons les résultats suivants :

$$
\begin{aligned}
y_1(t) &= \mathrm{e}^{-t} + t - 1,\\
y_2(t) &= -\mathrm{e}^{-2t}((t^2 - 5t + 6)\mathrm{e}^{2t} + (t^2 - 2t - 5)\mathrm{e}^{t} - 1)),\\
y_3(t) &= -\mathrm{e}^{-4t}((30t^4 - 420t^3 + 2370t^2 - 6570t + 7650)\mathrm{e}^{4t}\\
&\quad + (12t^5 - 105t^4 + 220t^3 + 390t^2 - 1800t - 6665)\mathrm{e}^{3t}\\
&\quad + (-30t^4 + 240t^2 - 420t - 810)\mathrm{e}^{2t} + (30t^2 - 30t - 165)\mathrm{e}^{t} - 10)/30),\\
y_4(t) &= -\mathrm{e}^{-8t}((4898880000t^8 - 176359680000t^7 + 2968721280000t^6\\
&\quad - 30794359680000t^5 + 217084069440000t^4 - 1072825326720000t^3\\
&\quad + 3650807039040000t^2 - 7848774884160000t + 8167324556160000)\mathrm{e}^{8t}\\
&\quad + (391910400t^{10} - 9906624000t^9 + 107693712000t^8 - 635128128000t^7\\
&\quad + 1969077600000t^6 - 1672804224000t^5 - 6384873600000t^4\\
&\quad - 3579992640000t^3 + 163399692960000t^2 - 555067598400000t\\
&\quad - 7529750204031367)\mathrm{e}^{7t} + (-783820800t^{10} + 5878656000t^9\\
&\quad - 26045712000t^8 - 145006848000t^7 + 98086464000t^6\\
&\quad - 2443561344000t^5 - 21702854880000t^4 - 38907993600000t^3\\
&\quad - 135191125440000t^2 - 423942543360000t - 598284020880000)\mathrm{e}^{6t}\\
&\quad + (1959552000t^9 - 8328096000t^8 - 13063680000t^7 + 177666048000t^6\\
&\quad - 161989632000t^5 - 2391469920000t^4 + 512749440000t^3\\
&\quad + 6887008800000t^2 - 20937813120000t - 32984159040000)\mathrm{e}^{5t}\\
&\quad + (-1632960000t^8 - 5660928000t^7 + 25655616000t^6 - 22607424000t^5\\
&\quad - 310655520000t^4 + 306472320000t^3 + 1540990080000t^2\\
&\quad - 2248935360000t - 5653151840000)\mathrm{e}^{4t} + (2449440000t^6\\
&\quad + 1551312000t^5 - 33985980000t^4 + 25889220000t^3 + 169649235000t^2\\
&\quad - 218905942500t - 599862965625)\mathrm{e}^{3t} + (-1632960000t^4 + 653184000t^3\\
&\quad + 15415142400t^2 - 13756055040t - 50025403008)\mathrm{e}^{2t} + (544320000t^2\\
&\quad - 362880000t - 3054240000)\mathrm{e}^{t} - 77760000)\\
&\quad /4898880000).
\end{aligned}
$$

Pour $a^z(t) = \mathrm{e}^t$, nous obtenons les résultats suivants :

$$
\begin{aligned}
y_1(t) &= \frac{\mathrm{e}^{t} - \mathrm{e}^{-t}}{2},\\
y_2(t) &= \frac{\mathrm{e}^{-2t}(\mathrm{e}^{4t} - 6\mathrm{e}^{3t} - 6\mathrm{e}^{2t} + 14\mathrm{e}^{t} - 3)}{12},\\
y_3(t) &= -\mathrm{e}^{-4t}(\mathrm{e}^{8t} - 15\mathrm{e}^{7t} + 40\mathrm{e}^{6t} - 110\mathrm{e}^{5t} - 690\mathrm{e}^{4t}\\
&\quad + (1739 - 660t)\mathrm{e}^{3t} - 1160\mathrm{e}^{2t} + 210\mathrm{e}^{t} - 15)/720),
\end{aligned}
$$

$$
\begin{aligned}
y_4(t) = -e^{-8t}(&28e^{16t} - 945e^{15t} + 10980e^{14t} - 59640e^{13t} + 177408e^{12t} \\
&+ (989604 - 83160t)e^{11t} + (1663200t - 8751960)e^{10t} \\
&+ (-6652800t - 20898360)e^{9t} + (36590400t - 38004120)e^{8t} \\
&+ (114760800t^2 - 536097240t + 551586977)e^{7t} + (-109771200t^2 \\
&+ 358918560t - 794616732)e^{6t} + (447977880 - 192931200t)e^{5t} \\
&+ (23284800t - 168359520)e^{4t} + (33668460 - 1247400t)e^{3t} \\
&- 3976560e^{2t} + 264600e^{t} - 8100)/130636800).
\end{aligned}
$$

Et pour $a^z(t) = \cos(t)$, nous obtenons les résultats suivants :

$$
\begin{aligned}
y_1(t) &= (\sin(t) + \cos(t) - e^{-t})/2, \\
y_2(t) &= -e^{-2t}(e^{2t}\sin(2t) - 2e^{2t}\cos(2t) + (-10e^{2t} - 10e^{t})\sin(t) \\
&\quad + (10e^{t} - 10e^{2t})\cos(t) + 5e^{2t} + 2e^{t} - 5)/20), \\
y_3(t) &= -e^{-4t}(24e^{4t}\sin(4t) + 57e^{4t}\cos(4t) \\
&\quad + (1020e^{4t} - 340e^{3t})\sin(3t) - 1020e^{3t}\cos(3t) \\
&\quad + (1428e^{4t} - 10608e^{3t} + 3060e^{2t})\sin(2t) \\
&\quad + (-4896e^{4t} - 204e^{3t} + 4080e^{2t})\cos(2t) \\
&\quad + (-31620e^{4t} + 3060e^{3t} + 4080e^{2t} - 6120e^{t})\sin(t) \\
&\quad + (-18360e^{4t} + 15300e^{3t} - 10200e^{2t} + 2040e^{t})\cos(t) \\
&\quad + 13005e^{4t} + (2040t + 5536)e^{3t} - 5508e^{2t} + 1020e^{t} - 850)/40800), \\
y_4(t) &= -e^{-8t}(181165320e^{8t}\sin(8t) - 144323865e^{8t}\cos(8t) \\
&\quad + (-2211028092e^{8t} - 6971710200e^{7t})\sin(7t) \\
&\quad + (6118031400e^{7t} - 8425809756e^{8t})\cos(7t) \\
&\quad + (-132936247080e^{8t} + 95977210420e^{7t} + 82072372200e^{6t})\sin(6t) \\
&\quad + (-16279885440e^{8t} + 99197476560e^{7t} - 114436540400e^{6t})\cos(6t) \\
&\quad + (-599298934500e^{8t} + 3339050802360e^{7t} \\
&\quad - 1916607407400e^{6t} - 157361458800e^{5t})\sin(5t) \\
&\quad + (1293405466500e^{8t} - 105770803320e^{7t} \\
&\quad - 2553561238200e^{6t} + 1001934351600e^{5t})\cos(5t) \\
&\quad + (8828575413840e^{8t} + (56769640200t + 5286829460640)e^{7t} \\
&\quad - 21469680348480e^{6t} + 13867727546400e^{5t} - 2204653956960e^{4t})\sin(4t) \\
&\quad + (8332886818620e^{8t} + (-23903006400t - 13257084823710)e^{7t} \\
&\quad + 11710082835360e^{6t} + 3433069294200e^{5t} - 2738686958280e^{4t})\cos(4t) \\
&\quad + (54309124478700e^{8t} - 141632948602600e^{7t} \\
&\quad + (62989667245960 - 1083602956800t)e^{6t} - 6490125910800e^{5t} \\
&\quad - 13691774860400e^{4t} + 6208141940000e^{3t})\sin(3t)
\end{aligned}
$$

$$
\begin{aligned}
&+ 338625924000\mathrm{e}^{7t}\cos^2(3t) + (-32829285352500\mathrm{e}^{8t} \\
&+ (-1354503696000t - 68585361044200)\mathrm{e}^{7t} \\
&+ (812702217600t + 108226638035880)\mathrm{e}^{6t} - 62647481408400\mathrm{e}^{5t} \\
&+ 25859733062800\mathrm{e}^{4t} - 2370381468000\mathrm{e}^{3t})\cos(3t) \\
&+ (-74923375985640\mathrm{e}^{8t} + (-9752426611200t - 101753405115180)\mathrm{e}^{7t} \\
&+ (8127022176000t + 174509674783080)\mathrm{e}^{6t} \\
&+ (1015877772000t - 144856999385280)\mathrm{e}^{5t} + 52153601925600\mathrm{e}^{4t} \\
&- 17303784716400\mathrm{e}^{3t} + 1389533964000\mathrm{e}^{2t})\sin(2t) \\
&- 25214086301040\mathrm{e}^{7t}\cos^2(2t) + (-329784998699520\mathrm{e}^{8t} \\
&+ (126896678759760 - 2844457761600t)\mathrm{e}^{7t} \\
&+ (17066746569600t - 194640985964880)\mathrm{e}^{6t} \\
&+ (27006015014160 - 7111144404000t)\mathrm{e}^{5t} + 16347817684800\mathrm{e}^{4t} \\
&- 11784182155200\mathrm{e}^{3t} + 5161126152000\mathrm{e}^{2t})\cos(2t) \\
&+ (-1515655275252300\mathrm{e}^{8t} + (320185750521000 - 73143199584000t)\mathrm{e}^{7t} \\
&+ (24381066528000t - 182624944647600)\mathrm{e}^{6t} \\
&+ (-14628639916800t - 22292740535520)\mathrm{e}^{5t} \\
&+ (8127022176000t + 61328341903920)\mathrm{e}^{4t} - 27954565984800\mathrm{e}^{3t} \\
&+ 4454233308000\mathrm{e}^{2t} - 1738889880000\mathrm{e}^{t})\sin(t) \\
&+ 527240563668000\mathrm{e}^{7t}\cos^2(t) + (-530908181212500\mathrm{e}^{8t} \\
&+ (125968843728000t - 31688414776200)\mathrm{e}^{7t} \\
&+ (-36571599792000t - 181903870621200)\mathrm{e}^{6t} \\
&+ (13003235481600t + 121907722940640)\mathrm{e}^{5t} - 55311158426160\mathrm{e}^{4t} \\
&+ 12465417837600\mathrm{e}^{3t} - 3308114796000\mathrm{e}^{2t} + 274561560000\mathrm{e}^{t})\cos(t) \\
&+ 831053377279125\mathrm{e}^{8t} + (25904883186000t^2 - 242320711214400t \\
&- 150867818096054)\mathrm{e}^{7t} + (-4063511088000t^2 - 30181529414400t \\
&- 136252151424480)\mathrm{e}^{6t} + (10971479937600t + 32465660867640)\mathrm{e}^{5t} \\
&+ (-1354503696000t - 34742422227240)\mathrm{e}^{4t} \\
&+ (846564810000t + 16426511182900)\mathrm{e}^{3t} - 6095266632000\mathrm{e}^{2t} \\
&+ 282188270000\mathrm{e}^{t} - 100781525000)/1625404435200000).
\end{aligned}
$$

Le tableau suivant donne une idée sur la complexité du schéma :

	un	t	e^t	$\cos(t)$
y_2	9 s	15 s	13 s	21 s
y_3	27 s	49 s	32 s	124 s
y_4	76 s	215 s	84 s	1060 s

Exemple 4.2.3.

Considérons l'équation différentielle non linéaire en régime forcé suivante :

$$\ddot{y}(t) + \dot{y}(t) + y(t) + y^3(t) = a^z(t) \quad \text{avec} \quad \ddot{y}(0) = \dot{y}(0) = y(0) = 0,$$

ou équivalemment :

$$y(t) + \int_0^t y(\tau)d\tau + \int_0^t \int_0^\tau y(\rho)d\rho d\tau = \int_0^t \int_0^\tau a^z(\rho)d\rho d\tau - \int_0^t \int_0^\tau y^3(\rho)d\rho d\tau.$$

Si S est une série telle que $\mathcal{E}_a(S)(t) = y(t)$, alors nous avons :

$$S(1 + z_0 + z_0^2) = zz_0 - (S \sqcup\!\sqcup S \sqcup\!\sqcup S)z_0^2.$$

L'équation $1 + z_0 + z_0^2 = 0$ admet deux racines complexes distinctes :

$$k_1 = -\frac{i\sqrt{3}+1}{2} \qquad \text{et} \qquad k_2 = \frac{i\sqrt{3}-1}{2}.$$

Par conséquent :

$$\begin{aligned}
\frac{1}{1 + z_0 + z_0^2} &= \frac{i\sqrt{3}}{3}\left(\frac{1}{z_0 - k1} - \frac{1}{z_0 - k2}\right) \\
&= A_1\left(\frac{z_0}{k_1}\right)^* + A_2\left(\frac{z_0}{k_2}\right)^*.
\end{aligned}$$

où $A_1 = -\frac{i\sqrt{3}}{3k_1}$ et $A_2 = \frac{i\sqrt{3}}{3k_2}$. Dans ce cas, nous avons :

$$\begin{aligned}
S &= zz_0\left[A_1\left(\frac{z_0}{k_1}\right)^* + A_2\left(\frac{z_0}{k_2}\right)^*\right] - (S \sqcup\!\sqcup S \sqcup\!\sqcup S)z_0^2\left[A_1\left(\frac{z_0}{k_1}\right)^* + A_2\left(\frac{z_0}{k_2}\right)^*\right] \\
&= z\left[A_1\left(\frac{z_0}{k_1}\right)^* + A_2\left(\frac{z_0}{k_2}\right)^*\right]z_0 - (S \sqcup\!\sqcup S \sqcup\!\sqcup S)z_0\left[A_1\left(\frac{z_0}{k_1}\right)^* + A_2\left(\frac{z_0}{k_2}\right)^*\right]z_0.
\end{aligned}$$

Notons $h(t)$ l'Evaluation de la série $\left[A_1\left(\frac{z_0}{k_1}\right)^* + A_2\left(\frac{z_0}{k_2}\right)^*\right]z_0$:

$$h(t) = A_1 k_1 e^{\frac{t}{k_1}} + A_2 k_2 e^{\frac{t}{k_2}} = \frac{i\sqrt{3}}{3}(e^{\frac{t}{k_2}} - e^{\frac{t}{k_1}}) = \frac{2\sqrt{3}e^{-\frac{t}{2}}}{3}\sin(\frac{\sqrt{3}t}{2}).$$

D'après ce qui précède, la suite $\{y_k\}_{k\geq 0}$ suivante converge vers la solution de cette équation différentielle :

$$\begin{cases}
y_1(t) = \displaystyle\int_0^t a^z(\tau)h(t-\tau)d\tau \\
y_k(t) = y_1(t) + \displaystyle\int_0^t [-y_{k-1}^3(\tau)]h(t-\tau)d\tau
\end{cases}$$

Pour $a^z = un$, nous obtenons les résultats suivants :

$$y_1(t) = 1 - e^{-t},$$
$$y_2(t) = \mathrm{e}^{-\frac{7t}{2}}(((5\sqrt{3} - 20)\mathrm{e}^{\frac{t}{2}} + 64)\mathrm{e}^{3t} + (-5\sqrt{3} - 60)\mathrm{e}^{\frac{5t}{2}} + 20\mathrm{e}^{\frac{3t}{2}} - 4\mathrm{e}^{\frac{t}{2}})$$
$$/(5\sqrt{3})),$$
$$y_3(t) = \mathrm{e}^{-\frac{33t}{2}}((3920716800\sqrt{3} - 9311702400)t\mathrm{e}^{16t}$$
$$+ \mathrm{e}^{t}(57432375\mathrm{e}^{\frac{31t}{2}} + (28920414784 - 10025174720\sqrt{3})\mathrm{e}^{15t}$$
$$+ (18442684260\sqrt{3} - 65806781040)\mathrm{e}^{\frac{29t}{2}}$$
$$+ (-3697113420\sqrt{3} - 61416395040)\mathrm{e}^{\frac{27t}{2}}$$
$$+ (5944786848 - 3102879780\sqrt{3})\mathrm{e}^{\frac{25t}{2}}$$
$$+ (1365249600\sqrt{3} + 3510791856)\mathrm{e}^{\frac{23t}{2}}$$
$$+ (-353953600\sqrt{3} - 2416414000)\mathrm{e}^{\frac{21t}{2}}$$
$$+ (49008960\sqrt{3} + 665333760)\mathrm{e}^{\frac{19t}{2}}$$
$$+ (-3769920\sqrt{3} - 120637440)\mathrm{e}^{\frac{17t}{2}} + 13069056\mathrm{e}^{\frac{15t}{2}}$$
$$- 768768\mathrm{e}^{\frac{13t}{2}}) - 57432375\mathrm{e}^{\frac{31t}{2}}$$
$$+ (70873490688 - 7841433600\sqrt{3})\mathrm{e}^{15t}$$
$$+ (7841433600\sqrt{3} + 28180152000)\mathrm{e}^{14t}$$
$$+ (-1568286720\sqrt{3} - 14637342720)\mathrm{e}^{13t}$$
$$+ (196035840\sqrt{3} + 4312788480)\mathrm{e}^{12t} - 627314688\mathrm{e}^{11t}$$
$$+ ((2552550000 - 1301800500\sqrt{3})\mathrm{e}^{\frac{13t}{2}}$$
$$+ 52276224)\mathrm{e}^{\frac{10t}{2}})/57432375).$$

Pour $a^z(t) = t$, nous obtenons les résultats suivants :

$$y_1(t) = \mathrm{e}^{-t} + t - 1,$$
$$y_2(t) = -\mathrm{e}^{-\frac{7t}{2}}(((60t^3 - 540t^2 + (2340 - 15\sqrt{3}t + 15\sqrt{3} - 4740)\mathrm{e}^{\frac{t}{2}} + 5632)\mathrm{e}^{3t}$$
$$+ (-180t^2 - 360t - 15\sqrt{3} - 900)\mathrm{e}^{\frac{5t}{2}} + (20 - 60t)\mathrm{e}^{\frac{3t}{2}} - 12\mathrm{e}^{\frac{t}{2}})/(15\sqrt{3})),$$
$$y_3(t) = \mathrm{e}^{-\frac{33t}{2}}((1698019241443639994880000000t^7$$
$$- 3565840407031643989248000000t^6$$
$$+ (37797908314535426286028800000$$
$$- 1188613469010547996416000000\sqrt{3}t^5$$
$$+ (14857668362631849955200000000\sqrt{3}$$
$$- 255551895837267819229440000000)t^4$$
$$+ (116610410144116074373387200000$$
$$- 95089077520843839713280000000\sqrt{3}t^3$$

$$
\begin{aligned}
&+ (35064097335811165894272000000\sqrt{3}\\
&- 36623409630469378547070240000000)t^2\\
&+ (741835952512026952338158400000\\
&- 46950232025916645858432000000\sqrt{3}t)e^{16t}\\
&+ e^t((29678403600782725851562 5t\\
&- 296784036007827258515625)e^{\frac{31t}{2}}\\
&+ (77647293422503437087315428832389 12\\
&- 17403830260726573566444998776320\sqrt{3})e^{15t}\\
&+ (75976713218003778180000000t^8\\
&+ (97250192919044836070400000000\\
&- 31656963840834907575000000\sqrt{3}t^6\\
&+ (686829487490754154747200000000 - 18994178304500944545000000 0\sqrt{3}t^5\\
&+ (840155243309261904350576250000\\
&- 21463421484086067335850000000\sqrt{3}t^4\\
&+ (615052510909778557338182460000 0\\
&- 198805732920443219571000000000\sqrt{3}t^3\\
&+ (395668355913128926770329187000 00\\
&- 10380199729795363062939093750 0\sqrt{3})t^2\\
&+ (148173295482134507801150134200000\\
&- 3820278435008969898457242750 00\sqrt{3}t\\
&- 795331270605637282236307087500\sqrt{3}\\
&+ 3163450123763599408899522862500 00)e^{\frac{29t}{2}}\\
&+ (-67534856193781136160000000t^7 - 900464749250415148800000000t^6\\
&+ (211046425605566050500000 0\sqrt{3} - 2588836154094943552800000000)t^5\\
&+ (19627317581317642696500000 0\sqrt{3} - 6367036164491477114640000000 0)t^4\\
&+ (16743016431374906673000000 0\sqrt{3} + 4596864728389087646992500000 0)t^3\\
&+ (159691795374878311545000000 0\sqrt{3}\\
&+ 21391913170855533109093290000 00)t^2\\
&+ (1256590936458029694709687500\sqrt{3}\\
&+ 84747075989034049011971550000 0)t\\
&+ 2407062590750290569610263750 0\sqrt{3}\\
&+ 18132615134929737032806215000 00)e^{\frac{27t}{2}}\\
&+ (16208365486507472678400000t^6 + 51704685901958837844096000 0t^5\\
&+ (15955109775780793417800000\sqrt{3} + 156491768772229648709952000 0)t^4
\end{aligned}
$$

BU LILLE

$$
\begin{aligned}
&+ (10488163166894210445648000\sqrt{3} + 63938760171174678221752320000)t^3\\
&+ (174989565883706301904176000\sqrt{3}\\
&+ 1522258384086936016398717840 0)t^2\\
&+ (615099366030215334408940800\sqrt{3} + 697841231739120889208816227 20)t\\
&+ 43698927613936215710471382 0\sqrt{3}\\
&+ 2107164890331951994574138908 8)e^{\frac{25t}{2}}\\
&+ (2460198332773455674400000 0t^5 + 21990176134117918927200000 0t^4\\
&+ (7054980513100350831000000\sqrt{3} + 38767238927641051027200000 0)t^3\\
&+ (17926024966742161269000000\sqrt{3} + 14147778858001009341120000 00)t^2\\
&+ (23609716459824227733000000\sqrt{3} + 10394951179755227377925000 00)t\\
&+ 20755096002528782895000000\sqrt{3} + 81473123970490896874701900 00)e^{\frac{23t}{2}}\\
&+ (1238139030219320829600000 0t^4 + 53027368566968892096000000t^3\\
&+ (1829069021914905771000000\sqrt{3} + 90880238581754862240000000)t^2\\
&+ (1657105267717777878000000\sqrt{3} + 21898339241544560995200000 0)t\\
&+ 4761137881356843987000000\sqrt{3} + 15044597724131674023325000 0)e^{\frac{21t}{2}}\\
&+ (343813813350158511360000 0t^3 + 65413926825711976512000000t^2\\
&+ (253255710726679260600000\sqrt{3} + 1101496379783668989120000 0)t\\
&- 537209083359622674000000\sqrt{3} + 4644993515336424432000000 0)e^{\frac{19t}{2}}\\
&+ (623398672557979718400000t^2 + 3996145336910126400000000t\\
&+ 194812085174368662000000\sqrt{3} + 1273643244558791568000000 0)e^{\frac{17t}{2}}\\
&+ (6753485619378113616 0000t - 135069712387562272320 00)e^{\frac{15t}{2}}\\
&+ 3972638599634184480000e^{\frac{13t}{2}}) + 296784036007827258515625e^{\frac{31t}{2}}\\
&+ (71316808140632879784960000 00t^5\\
&- 14263361628126575956992000000t^4\\
&+ (12837025465313918361292800000 0\\
&- 11886134690105479964160000 00\sqrt{3}t^3\\
&+ (570534465125063038279680000 00\\
&- 10697521221094931967744000000\sqrt{3}t^2\\
&+ (37753335309447530736163200000 0\\
&- 35658404070316439892480000 00\sqrt{3}t\\
&- 41601471415369179874560000 000\sqrt{3}\\
&- 5929973226896351189633556480000)e^{15t}\\
&+ (-41601471415369179874560000 00t^4
\end{aligned}
$$

$$
\begin{aligned}
&- 40809062436028814543616000000t^3 \\
&+ (-11886134690105479964160000000\sqrt{3} \\
&- 8617447650326472974016000000\mathrm{0})t^2 \\
&+ (-25753291828561873255680000000\sqrt{3} \\
&- 30250212786318446508787200000\mathrm{0})t \\
&- 5844016222635194315712000000\sqrt{3} \\
&- 25378135702405419044311200000\mathrm{0})\mathrm{e}^{14t} \\
&+ (-2218745142153022926643200000t^3 \\
&- 7607126201667507177062400000t^2 \\
&+ (-2377226938021095992832000000\sqrt{3} \\
&- 9191944160348237838950400000)t \\
&+ 2641363264467884436480000\mathrm{0}\sqrt{3} \\
&- 11621998363658691520512000000)\mathrm{e}^{13t} \\
&+ (-65373740795580139802880000\mathrm{0}t^2 - 841934540549138164128000000t \\
&- 2971533672526369991040000\mathrm{0}\sqrt{3} \\
&- 20264208794589550911120000\mathrm{00})\mathrm{e}^{12t} \\
&+ (126785436694458452951040\mathrm{00} - 950890775208438397132800\mathrm{00}t)\mathrm{e}^{11t} \\
&+ ((844185702422264202000000\mathrm{0}t^9 - 3798835660900188909000000000t^8 \\
&+ (9117205586160453381600000000 - 6331392768166981515000000\sqrt{3}t^7 \\
&+ (2089359613495103899950000\mathrm{00}\sqrt{3} \\
&- 15357426298465830362784\mathrm{0000000})t^6 \\
&+ (1997432539045895453588336250000 \\
&- 362788805615968040809500000\mathrm{0}\sqrt{3}t^5 \\
&+ (4273056979235895824473500000\mathrm{0}\sqrt{3} \\
&- 20639173865106824972555861250000)t^4 \\
&+ (1671769912628724163161031125000\mathrm{00} \\
&- 36592324074825870709576968750\mathrm{0}\sqrt{3}t^3 \\
&+ (2253187962779856640577727187500\sqrt{3} \\
&- 10075283431317355247521249725000\mathrm{00})t^2 \\
&+ (40362785266637398962150966712500\mathrm{00} \\
&- 9091281302759145665350487812500\sqrt{3}t \\
&+ 1822207722349646947277243531250\mathrm{0}\sqrt{3} \\
&- 80767195932078670135227468862500\mathrm{00})\mathrm{e}^{\frac{13t}{2}} \\
&- 792408979340365330944000\mathrm{0})\mathrm{e}^{10t})/2967840360078272585156\mathrm{25}).
\end{aligned}
$$

Pour $a^{z}(t) = \mathrm{e}^{t}$, nous obtenons les résultats suivants :

$$
\begin{aligned}
y_1(t) &= (\mathrm{e}^{t} - \mathrm{e}^{-t})/2,\\
y_2(t) &= -\mathrm{e}^{-\frac{7t}{2}}(5\mathrm{e}^{-\frac{13t}{2}} + (-35\sqrt{3} - 35)\mathrm{e}^{\frac{9t}{2}} + 128\mathrm{e}^{3t} + (35\sqrt{3} - 105)\mathrm{e}^{\frac{5t}{2}}\\
&\quad + 7\mathrm{e}^{\frac{t}{2}})/(70\sqrt{3})),\\
&\quad + (-24249225\sqrt{3} - 24249225)\mathrm{e}^{\frac{45t}{2}} + (496006875\sqrt{3} + 826678125)\mathrm{e}^{\frac{41t}{2}}\\
&\quad + (-3637383750\sqrt{3} - 8476836225)\mathrm{e}^{\frac{37t}{2}} + 6810845184\mathrm{e}^{\frac{35t}{2}}\\
&\quad + (-17313946650\sqrt{3} - 25801175400)\mathrm{e}^{\frac{33t}{2}},\\
y_3(t) &= \mathrm{e}^{\frac{33t}{2}}(\mathrm{e}^{t}(911625\mathrm{e}^{\frac{49t}{2}} + (139674088960\sqrt{3} + 367754007040)\mathrm{e}^{15t}\\
&\quad + (98791342650\sqrt{3} - 118142224200)\mathrm{e}^{\frac{29t}{2}}\\
&\quad + (23837958144 0 - 79459860480\sqrt{3}\mathrm{e}^{\frac{27t}{2}}\\
&\quad + (64143080001 - 38498069610\sqrt{3})\mathrm{e}^{\frac{25t}{2}} - 6810845184\mathrm{e}^{\frac{23t}{2}}\\
&\quad + (2455638185\sqrt{3} - 4673633965)\mathrm{e}^{\frac{21t}{2}}\\
&\quad + (164521665 - 54840555\sqrt{3})\mathrm{e}^{\frac{17t}{2}} - 2795793\mathrm{e}^{\frac{13t}{2}})\\
&\quad + 110853600\mathrm{e}^{22t} + (-2327925600\sqrt{3} - 2327925600)\mathrm{e}^{20t}\\
&\quad + (37246809600\sqrt{3} + 51214363200)\mathrm{e}^{\frac{16t}{2}} + 53469541125\mathrm{e}^{\frac{35t}{2}}\\
&\quad + (130363833600\sqrt{3} + 1862340480)t\mathrm{e}^{16t} - 53469541125\mathrm{e}^{\frac{31t}{2}}\\
&\quad - 145298030592\mathrm{e}^{15t} + (104291066880\sqrt{3} - 189027558720)\mathrm{e}^{14t}\\
&\quad + (9777287520 - 3259095840\sqrt{3}\mathrm{e}^{12t}\\
&\quad + ((-238379581440\sqrt{3} - 238379581440)\mathrm{e}^{\frac{13t}{2}}\\
&\quad - 217273056)\mathrm{e}^{10t})/106939082250).
\end{aligned}
$$

Et pour $a^2(t) = \cos(t)$, nous obtenons les résultats suivants :

$$y_1(t) = \sin(t) - (2/\sqrt{3})e^{-\frac{t}{2}}\sin(\frac{\sqrt{3}t}{2}),$$

$$\begin{aligned}
y_2(t) = &-e^{-\frac{3t}{2}}((70093170\sqrt{3} - 280372680)e^{2t}\cos(\frac{(\sqrt{3}+4)t}{2})\\
&+ (106464540\sqrt{3} - 150302880)e^{\frac{3t}{2}}\sin((\sqrt{3}+1)t)\\
&+ (338181480 - 87676680\sqrt{3}e^{\frac{3t}{2}}\cos((\sqrt{3}+1)t)\\
&- 26127920e^{t}\sin(\frac{3\sqrt{3}t}{2}) - 39191880\sqrt{3}e^{t}\cos(\frac{3\sqrt{3}t}{2})\\
&+ e^{t}(303737070\sqrt{3}e^{t} + 347128080)\sin(\frac{\sqrt{3}t}{2})\\
&+ e^{t}(607474140\sqrt{3}e^{t} + 173564040\sqrt{3}\cos(\frac{sqrt3t}{2})\\
&+ (-106464540\sqrt{3} - 150302880)e^{\frac{3t}{2}}\sin((\sqrt{3}-1)t)\\
&+ (-87676680\sqrt{3} - 338181480)e^{\frac{3t}{2}}\cos((\sqrt{3}-1)t)\\
&+ (70093170\sqrt{3} + 280372680)e^{2t}\cos(\frac{(\sqrt{3}-4)t}{2})\\
&- 4104555\sqrt{3}e^{\frac{3t}{2}}\sin(3t) + 24627330\sqrt{3}e^{\frac{3t}{2}}\cos(3t)\\
&+ ((91121121\sqrt{3} - 455605605)e^{\frac{3t}{2}} - 242989656\sqrt{3}e^{\frac{3t}{2}})\sin(t)\\
&+ (-182242242\sqrt{3}e^{\frac{3t}{2}} - 485979312\sqrt{3}e^{\frac{3t}{2}})\cos(t)\\
&- 63085056\sqrt{3}e^{2t})/455605605).
\end{aligned}$$

Le tableau suivant donne une idée sur la complexité du schéma :

	un	t	e^t	$\cos(t)$
y_2	20 s	38 s	17 s	169 s
y_3	110 s	745 s	115 s	-
y_4	2932 s	-	-	-

CHAPITRE 5

DECOUPLAGE DES SYSTEMES DYNAMIQUES NON LINEAIRES

5.0. Introduction

Les systèmes dynamiques ont permis de modéliser de nombreux problèmes techniques ou biomédicaux. Une grande question soulevée durant l'étude de ces systèmes est : comment peut-on rendre l'observation de ces systèmes insensibles aux actions perturbantes ?

Ainsi on peut se poser la question dans le domaine du pilotage automatique :

Que faire pour maintenir la trajectoire d'un engin en dépit des perturbations telles que le vent, le courant, etc ....? ([GM])

et en biomédecine :

Quelles doses de tranquilisants et d'anti-dépresseur afin de supprimer l'état dépressif sans augmenter l'angoisse chez un patient ? ([B3])

En termes de systèmes dynamiques, ces questions reviennent à résoudre de manière effective l'un des problèmes suivants :

- **Le rejet de perturbations** : *toutes les sorties du système* $\{y_s\}_{1\leq s\leq p}$ *sont indépendantes des entrées* $\{a^x\}_{x\in X_1}$.
- **Le découplage k-bloc triangulaire** : *chaque bloc de sorties* $\{y_s\}_{s\in\pi_l}$, $1\leq l\leq k$ *ne dépend que des blocs d'entrées* $\{a^x\}_{x\in X_j}$, $1\leq j\leq l$.
- **Le découplage k-bloc diagonal** : *chaque bloc de sorties* $\{y_s\}_{s\in\pi_l}$ *ne dépend que d'un seul bloc d'entrées* $\{a^x\}_{x\in X_l}$, $1\leq l\leq k$.
- **La commande non interactive** : *chaque sortie ne dépend que d'une seule entrée (c'est une application du cas précédent avec* $k=m=p$*).*

La théorie de la commande non interactive des systèmes linéaires a été largement développée par P.L. Falb et W.A. Wolowich ([FW]). Leur réponse consiste à trouver les matrices F et G de dimensions respectivement $m\times n$ et $m\times m$ pour que la matrice de tranfert $T(s)$ soit **diagonale**, avec

$$T(s)=C[sI-(A+BF)]^{-1}BG.$$

En parallèle avec ces méthodes basées sur l'algèbre matricielle, les caractérisations du **découplage par blocs** des systèmes linéaires à partir des critères géométriques sont données par G. Basile et G. Marro ([BM]), A.S. Morse et W.M. Wonham ([MW]). Soit une partition arbitraire de $Y \simeq I\!R^p$ en somme directe :

$$Y = \bigoplus_{i=1}^{k} Y_i \quad \text{où} \quad Y_i \simeq I\!R^{p_i} \quad \text{et} \quad \sum_{i=1}^{k} p_i = p.$$

La sortie s'écrit alors $y = {}^t(y_1, y_2, \ldots, y_k)$, où pour tout $i \in [1..k]$, $y_i \in Y_i$. Le découplage relatif à cette k-partition par bouclage statique par retour d'état revient donc à trouver des applications linéaires :

$$F : I\!R^n \mapsto I\!R^m, \quad \text{et} \quad \forall i \in [1..k], \quad G_i : I\!R^m \mapsto I\!R^m$$

telles que pour tout $i \in [1..k]$:

$$Y_i = C[(ImB + (A+BF)Im(BG_i) + \ldots + (A+BF)^{n-1}Im(BG_i]).$$

L'extension de la théorie du découplage aux systèmes non linéaires instationnaires a été faite par Freund ([F9], [F10]) suivant une idée originellement due à W.A. Porter.

Quand aux systèmes non linéaires plus généraux, une approche pour traiter ce problème passe par la géométrie différentielle ([H1], [L5]). Dans cette approche la notion de sous-espace vectoriel Y_i est remplacée par celle de **distribution invariante**. Comme l'ont montré par C. Gori-Giorgi, A. Isidori, A.J. Krener et S. Monaco ([GIKM]), le rejet des perturbations peut être assuré par l'existence d'une **carte locale** dans laquelle la dynamique du système se décompose en deux parties dont une est **inobservable**. Cette méthode a permis aussi à H. Nijmeijer et J.M. Schumacher ([NS]) de donner une condition **locale** nécessaire et suffisante de découplage par blocs par bouclage régulier. L'existence des singularités liées à la nature physique des systèmes a limité la portée de cette méthode car elle conduit à ne considérer que les distributions de rang constant, ce qui est difficile à réaliser dans la pratique. Pour une lecture approfondie de cette approche, on peut consulter le récent livre d'A. Isidori ([I]).

On sait que l'algèbre des séries commutatives joue un rôle déterminant en géométrie (voir par ex. [A1], [M1]). De même, l'algèbre des séries formelles non commutatives trouve aussi sa place naturelle en géométrie différentielle. Ainsi en utilisant conjointement les **séries de Fliess et l'algèbre des champs de vecteurs**, D. Claude ([C3]) étudie le rejet de perturbations pour les systèmes non linéaires. Dans cette approche plus algébrique, les algèbres engendrées par les modules remplaçant les distributions invariantes conduisent à une classe des lois de bouclage **génériques**, donc plus large que celle obtenue par les distributions invariantes. **L'existence de la plus grande algèbre de découplage** a été mise en évidence. En introduisant le concept d'**immersion** ([CFI]), qui préserve le comportement d'entrée-sortie, et un choix judicieux de lois de bouclage régulier assurant le découplage, D. Claude donne une condition nécessaire et suffisante pour qu'un système non linéaire puisse, par bouclage, être immergé dans un système linéaire dont la dimension dépend des **nombres caractéristiques**, il généralise ainsi les techniques de W.A. Porter ([P3]).

Ces travaux ont été prolongés dernièrement par C.H. Moog ([M2]) pour résoudre le problème de **découplage par bouclage dynamique** en utilisant les techniques d'algèbre différentielle ([R4]) introduites par M. Fliess ([F6]), ce problème restait non résolu par les techniques de géométrie différentielle et par les séries génératrices.

Comme le soulignait déjà D. Claude ([C4]), la résolution de plusieurs problèmes distincts, mais voisins (le rejet de perturbations [C3], [C4], le découplage k-bloc triangulaire [N1], le découplage k-bloc diagonal ([NS]), la commande non interactive [GH] ...), conduit à dégager une notion commune de ces études. C'est ce que nous proposons en définissant **le découplage d'un sous-ensemble $\{y_s\}_{s\in\pi}$ de l'ensemble des sorties par rapport à un sous-ensemble $\{a^x\}_{x\in X_1}$ de l'ensemble des entrées.**

L'étude de cette nouvelle notion de découplage nous permet ici de proposer un seul traitement pour ces différents problèmes. Ceci nous conduit à réécrire les constructions de D. Claude ([C3]) dans un cadre unifié. Ce cadre d'une part, nous conduira à une approche par sous programme pour implanter les quatre problèmes cités par simple instanciation, évitant ainsi un "bricolage" informatique. Et d'autre part, il nous permettra d'étendre le théorème de caractérisation du découplage de D. Claude ([C3], [C4]) sans trop nous "torturer" au point de vue mathématique pour adapter son étude à d'autres problèmes.

Comme l'a montré par D. Claude ([C3], [C4]), l'obtention d'une **caractérisation algébrique** du découplage nécessite l'étude des systèmes **non initialisés**, et conduit à coder les systèmes dynamiques par leurs séries génératrices non commutatives à coefficients dans $C^\omega(Q)$, anneau des fonctions analytiques sur une variété analytique réelle Q. Ainsi, en utilisant donc systématiquement ces séries à coefficients analytiques, nous allons caractériser le découplage d'un sous-ensemble $\{y_s\}_{s\in\pi}$ de l'ensemble des sorties par rapport à un sous-ensemble $\{a^x\}_{x\in X_1}$ de l'ensemble des entrées. Nous retrouvons très simplement comme cas particulier les résultats classiques concernant les différents problèmes précités (rejet de perturbations, découplage k-bloc diagonal, découplage k-bloc triangulaire, systèmes non interactifs). Enfin, nous construisons les lois de bouclage génériques permettant de découpler chaque fois que c'est possible. Nous montrons ainsi l'avantage de la méthode de D. Claude par rapport aux autres techniques par le fait qu'elle donne effectivement des lois de bouclage. Pour cette construction, nous utilisons les **modules** (et les **algèbres**) **de découplage**, et des calculs d'**annulateurs** dans l'**algèbres de Lie libres**.

5.1. Les nombres caractéristiques

Définition 5.1.1. : *Soit $s \in [1..p]$,* **le nombre caractéristique** Ψ_s *du système (S) est le plus petit entier ν, lorsqu'il est défini, tel qu'il existe $x \in X_0$, $xx_0^\nu * h_s \not\equiv 0$. Si pour tout entier ν, et pour toute lettre $x \in X_0$, on a $xx_0^\nu * h_s \equiv 0$, alors Ψ_s vaut $+\infty$:*

$$\Psi_s = \begin{cases} \inf\{\nu \in \mathbb{N} \mid \ \exists x \in X_0, \quad xx_0^\nu * h_s \not\equiv 0\} \\ +\infty \quad si \quad \forall \nu \geq 0, \quad \forall x \in X_0, \quad xx_0^\nu * h_s \equiv 0. \end{cases}$$

Définition 5.1.2. : *Soit $s \in [1..p]$, tel que Ψ_s soit défini. Soit $x \in X_0$, x est* **une lettre significative pour** h_s *si $xx_0^{\Psi_s} * h_s \not\equiv 0$. Nous notons* $\operatorname{sgn}(h_s)$ **l'ensemble de toutes les lettres significatives pour** h_s *:*

$$\operatorname{sgn}(h_s) = \{x \in X_0 \mid \ xx_0^{\Psi_s} * h_s \not\equiv 0\}.$$

Soit $s \in [1..p]$, par convention, $\operatorname{sgn}(h_s)$ **est vide si et seulement si** $\Psi_s = +\infty$.

Les nombres caractéristiques, lorsqu'ils sont définis, généralisent la notion des "ordres différentiels" (P.L. Falb et W.A. Wolovich [FW]) dans le cadre des systèmes linéaires. Soit $s \in [1..p]$, Ψ_s exprime le plus haut degré de "non dépendance" entre la sortie y_s et les entrées $\{a^x\}_{x\in X}$. Et $\Psi_s + 1$ est le plus petit nombre de "dérivations" nécessaires pour que la sortie y_s soit affectée par les entrées $\{a^x\}_{x\in X}$ pour lesquelles $\operatorname{sgn}(h_s)$ est non vide. En effet, soit dh_s la différentielle de h_s :

$$dh_s = \sum_{k=1}^{N} \frac{\partial h_s}{\partial q_k} dq_k.$$

Puisque d'une part $dq = \sum_{x\in X} a^x(t) A_x(q)dt$, et d'autre part $A_x = \sum_{k=1}^{N} A_x^k(q) \frac{\partial}{\partial q_k}$, on obtient $dq_k = \sum_{x\in X} a^x(t) A_x^k(q)dt$, par conséquent :

$$dh_s = \sum_{x\in X} a^x(t) \left[\sum_{k=1}^{N} A_x^k(q) \frac{\partial h_s}{\partial q_k} \right] dt = \sum_{x\in X} a^x(t) A_x \circ h_s \, dt = \sum_{x\in X} a^x(t) \, x * h_s \, dt.$$

On peut alors écrire les dérivées successives de y_s jusqu'à l'ordre $\Psi_s + 1$ de y_s en tenant compte de la définition de Ψ_s :

$$y_s^{(\Psi_s+1)}(t) = x_0^{\Psi_s+1} * h_s(q(t)) + \sum_{x\in \operatorname{sgn}(h_s)} a^x(t) x x_0^{\Psi_s} * h_s(q(t)).$$

On obtient ainsi une équation différentielle liant y_s et les entrées $\{a^x\}_{x\in\mathrm{sgn}(h_s)}$. On a alors :

$$(I)\quad \begin{cases} y_1^{(\Psi_1+1)} - x_0^{\Psi_1+1} * h_1(q(t)) = \displaystyle\sum_{x\in\mathrm{sgn}(h_1)} a^x(t) x x_0^{\Psi_1} * h_1(q(t)) \\ y_2^{(\Psi_2+1)} - x_0^{\Psi_2+1} * h_2(q(t)) = \displaystyle\sum_{x\in\mathrm{sgn}(h_2)} a^x(t) x x_0^{\Psi_2} * h_2(q(t)) \\ \vdots \\ y_p^{(\Psi_p+1)} - x_0^{\Psi_p+1} * h_p(q(t)) = \displaystyle\sum_{x\in\mathrm{sgn}(h_p)} a^x(t) x x_0^{\Psi_p} * h_p(q(t)). \end{cases}$$

Définition 5.1.3. : *Les nombres caractéristiques* $\{\Psi_s\}_{s\in[1..p]}$ *sont supposés définis. La* **matrice de commande** *de* (S) *est*

$$\Omega = \begin{pmatrix} x_1 * \Delta \\ x_2 * \Delta \\ \vdots \\ x_m * \Delta \end{pmatrix}.$$

Nous appelons aussi **matrice de commande étendue** *de* (S) *la matrice définie comme suit :*

$$\Gamma = \begin{pmatrix} x_0 * \Delta \\ \Omega \end{pmatrix},$$

où

$$\Delta = \begin{pmatrix} x_0^{\Psi_1} * h_1 & x_0^{\Psi_2} * h_2 & \dots & x_0^{\Psi_p} * h_p \end{pmatrix},$$

et pour toute lettre $x \in X$, $x * \Delta$ *repésente la matrice*

$$x * \Delta = \begin{pmatrix} x x_0^{\Psi_1} * h_1 & x x_0^{\Psi_2} * h_2 & \dots & x x_0^{\Psi_p} * h_p \end{pmatrix}.$$

Ainsi, en notation matricielle, le système d'équations (I) s'écrit :

$$\begin{pmatrix} y_1^{(\Psi_1+1)} & y_2^{(\Psi_2+1)} & \dots & y_p^{(\Psi_p+1)} \end{pmatrix} = a\Gamma,$$

c'est-à-dire :

$$\begin{pmatrix} y_1^{(\Psi_1+1)} & y_2^{(\Psi_2+1)} & \dots & y_p^{(\Psi_p+1)} \end{pmatrix} - x_0 * \Delta = \begin{pmatrix} a^{x_1} & a^{x_2} & \dots & a^{x_m} \end{pmatrix}\Omega.$$

Lorsque Ω admet un inverse à droite, alors le système d'équations (I) **permet de reconstituer les entrées** $\{a^x\}_{x\in X_0}$ à partir des dérivées $\{y_s^{(\Psi_s+1)}\}_{s\in[1..p]}$.

Nous donnons, au chapitre VI, les fonctions écrites en MACSYMA permettant de calculer les nombres caractéristiques et de construire l'ensemble des letters significatives, les matrices Ω et Γ.

Lemme 5.1.1. : *Pour toutes lettres x et z de l'alphabet fini X et pour tout entier positif i, nous avons :*

$$ad_z^i x = \sum_{j=0}^{i}(-1)^j \binom{i}{j} z^{i-j} x z^j.$$

<u>*Preuve*</u> :

. Pour $i = 0$, la formule est immédiate.

. Supposons que le résultat soit vrai pour tout entier j, $0 \leq j \leq i$.

. Pour $j = i+1$, puisque $ad_z^{i+1}x = [z, ad_z^i x]$, alors d'après l'hypothèse de récurrence, nous avons :

$$\begin{aligned}
ad_z^{i+1}x &= \sum_{j=0}^{i}(-1)^j \binom{i}{j}[z, z^{i-j}xz^j] \\
&= \sum_{j=0}^{i}(-1)^j \binom{i}{j}(z^{i+1-j}xz^j - z^{i-j}xz^{j+1}) \\
&= \left[z^{i+1}x + \sum_{j=1}^{i}(-1)^j \binom{i}{j} z^{i+1-j}xz^j\right] \\
&\quad - \left[\sum_{j=0}^{i-1}(-1)^j \binom{i}{j} z^{i-j}xz^{j+1} + (-1)^i xz^{i+1}\right] \\
&= z^{i+1}x + \left[\sum_{j=1}^{i}(-1)^j \binom{i}{j} z^{i+1-j}xz^j\right. \\
&\quad \left. - \sum_{j=1}^{i}(-1)^{j-1} \binom{i}{j-1} z^{i-j+1}xz^j\right] + (-1)^{i+1}xz^{i+1} \\
&= z^{i+1}x + \sum_{j=1}^{i}(-1)^j \left[\binom{i}{j} + \binom{i}{j-1}\right] z^{i-j+1}xz^j + (-1)^{i+1}xz^{i+1}.
\end{aligned}$$

En utilisant la formule de la somme des coefficients binômiaux, nous avons le résultat •

Lemme 5.1.2. : *Soit $s \in [1..p]$ tel que Ψ_s soit défini. Soit x une lettre significative pour h_s. Alors pour tous entiers positifs i, k, nous avons :*

$$\begin{cases} (ad_{x_0}^i x)x_0^k \equiv 0 & si \quad i+k < \Psi_s \\ (ad_{x_0}^i x)x_0^k \not\equiv 0 & si \quad i+k = \Psi_s. \end{cases}$$

<u>*Preuve*</u> : D'après le lemme 5.1.1., pour tout entier positif k, nous avons

$$(ad_{x_0}^i x)x_0^k * h_s = \sum_{j=0}^{i}(-1)^j \binom{i}{j} x_0^{i-j} x x_0^{j+k} * h_s.$$

Si $i+k < \Psi_s$ alors $j+k < \Psi_s$, et d'après la définition de Ψ_s, le membre de droite est nul.

Si $i+k = \Psi_s$ alors d'après la définition of Ψ_s, pour tout couple (j,k) tel que $j+k < \Psi_s$, $xx_0^{j+k} * h_s \equiv 0$.

Par concéquent, le membre de droite est réduit donc à un seul terme $(-1)^i xx_0^{\Psi_s} * h_s$ qui n'est pas nul •

Lemme 5.1.3. : *Soit $s \in [1..p]$ tel que Ψ_s soit défini. Soit x une lettre significative pour h_s, alors les polynômes de Lie $\{ad^i_{x_0}x\}_{0\leq i\leq \Psi_s}$ sont linéairement indépendants dans Lie $< X > /Ann^L C^\omega(Q)$ sur $C^\omega(Q)$.*

<u>*Preuve*</u> : Si ce n'était pas le cas, supposons alors qu'il existe $\Psi_s + 1$ fonctions analytiques réelles non toutes nulles $\mu_0, \ldots, \mu_{\Psi_s}$ telles que :

$$\sum_{j=0}^{\Psi_s} \mu_j(ad^j_{x_0}x) * C^\omega(Q) = \{0\}.$$

Soit $m_0 \leq \Psi_s$ le plus grand indice tel que $\mu_{m_0} \not\equiv 0$, alors

$$\sum_{j=0}^{m_0} \mu_j(ad^j_{x_0}x) * C^\omega(Q) = \{0\}.$$

Par conséquent :

$$\sum_{j=0}^{m_0} \mu_j(ad^j_{x_0}x) * (x_0^{\Psi_s - m_0} * h_s) = \sum_{j=0}^{m_0} \mu_j(ad^j_{x_0}x)x_0^{\Psi_s - m_0} * h_s \equiv 0.$$

D'après le lemme 5.1.2., nous avons $\mu_{m_0} xx_0^{\Psi_s} * h_s \equiv 0$. D'après l'hypothèse, $xx_0^{\Psi_s} * h_s \not\equiv 0$, alors nous avons $\mu_{m_0} \equiv 0$, ce qui contredit l'hypothèse faite sur m_0 •

Proposition 5.1.1. : *Lorsque les nombres caractéristiques sont définis, ils sont strictement inférieurs à la dimension de la variété Q.*

<u>*Preuve*</u> : On sait que, pour tout état q de la variété Q, l'espace vectoriel *Lie* $< X > /Ann^L C^\omega(Q)$ est isomorphe à un sous-espace vectoriel de T_qQ (voir 1.9.2.), et d'après le lemme 5.1.3., on a une famille de $\Psi_s + 1$ éléments linéairement indépendants. D'où :

$$\Psi_s < N \quad \bullet$$

Corollaire 5.1.1. : *Soit $s \in [1..p]$ tel que :*

$$\forall \nu \in [1..N[, \quad \forall x \in X_0, \quad xx_0^\nu * h_s \equiv 0,$$

alors $\Psi_s = +\infty$, ou équivalemment $\mathrm{sgn}(h_s) = \emptyset$. *Dans ce cas, on dit que les entrées $\{a^x\}_{x\in X}$ n'affectent pas l'observation h_s (voir 5.2.).*

5.2. Formulation du problème

On suppose que les nombres caractéristiques $\{\Psi_s\}_{s\in[1..p]}$ *de* (S) *sont tous définis, et* $X_1 \subset X_0$ *est supposé fixé.*

Nous posons :

$$\begin{aligned}
H &= \{h_1, \ldots, h_p\},\\
G &= \{x_0^{\Psi_1} * h_1, \ldots, x_0^{\Psi_p} * h_p\}\\
F &= \bigcup_{s=1}^{p} \{x_0^n * h_s, 0 \leq n \leq \Psi_s\}.
\end{aligned}$$

Etant donné une partie de π de $[1..p]$, on pose :

$$\begin{aligned}
H_1 &= \{h_s\}_{s\in\pi},\\
G_1 &= \{x_0^{\Psi_s} * h_s\}_{s\in\pi},\\
F_1 &= \bigcup_{s\in\pi} \{x_0^n * h_s, 0 \leq n \leq \Psi_s\},
\end{aligned}$$

alors nous avons $G_1 \subset F_1$ et $H_1 \subset F_1 \subset (X \setminus X_1)^* * H_1$.

Définition 5.2.1. : *Pour tout état initial* $q(0) \in Q$*, les sorties* $\{y_s\}_{s\in\pi}$ *de* (S) *sont* **découplée par rapport à** *des entrées* $\{a^x\}_{x\in X_1}$ *si et seulement si pour tout* $s \in \pi$*, la sortie* y_s *n'est pas affectée par les entrées* $\{a^x\}_{x\in X_1}$.

Pour tout état initial $q(0) \in Q$ de (S), pour $s \in \pi$, si la sortie y_s n'est pas affectée par les entrées $\{a^x\}_{x\in X_1}$, alors toute variation de a^x, $x \in X_1$, n'affecte pas l'observation h_s. Rappelons que l'Evaluation de la série génératrice en $q(0)$ associée à h_s n'est rien d'autre que la sortie y_s. Nous allons voir dans la proposition suivante que si les entrées a^x, $x \in X_1$, n'affectent pas l'observation h_s, $s \in \pi$, alors les lettres de X_1 n'apparaissent pas dans le support de la série génératrice σh_s, $s \in \pi$. Désormais on dira tout simplement que H_1 est découplé par rapport à X_1 pour signifier que les sorties $\{y_s\}_{s\in\pi}$ sont découplées par rapport à $\{a^x\}_{x\in X_1}$.

Lemme 5.2.1. : *Etant donné* $s \in [1..p]$*, nous avons :*

$$X_1 \subset Ann(x_0^{\Psi_s} * h_s) \iff \mathrm{sgn}(h_s) \subset (X_0 \setminus X_1).$$

<u>*Preuve*</u> : En effet, étant donné $x \in X_1$, alors nous avons :

$$x \in Ann(x_0^{\Psi_s} * h_s) \iff x x_0^{\Psi_s} * h_s \equiv 0.$$

D'après la définition de $\mathrm{sgn}(h_s)$ (définition 5.1.2.), celà revient à dire que $\mathrm{sgn}(h_s) \bigcap X_1 = \emptyset$, ou équivalemment $\mathrm{sgn}(h_s) \subset (X_0 \setminus X_1)$ •

Proposition 5.2.1. : *Les propriétés suivantes sont équivalentes :*
(a) $\forall f \in H_1, \quad \mathrm{supp}(\sigma f) \subset (X \setminus X_1)^*$.
(b) $X_1(X \setminus X_1)^* \subset AnnH_1$.
(c) $\forall x \in X_1, \quad x(X \setminus \{x\})^* \subset AnnH_1$.
(d) $\{y_s\}_{s\in\pi}$ *est découplé par rapport à* $\{a^x\}_{x\in X_1}$.
(e) $\forall f \in H_1, \quad \sigma f = \sum_{w\in(X\setminus X_1)^*} (w * f)w$.

Preuve :

$a \Rightarrow b$: Si pour tout $f \in H_1$, le support de la série génératrice associée à f est inclus dans $(X \setminus X_1)^*$, alors pour tout $f \in H_1$, on a :

$$X^* \setminus (X \setminus X_1)^* \subset Annf.$$

Puisque $X_1(X \setminus X_1)^*$ est inclus dans $X^* \setminus (X \setminus X_1)^*$, alors pour tout $f \in H_1$, on a l'inclusion suivante :

$$X_1(X \setminus X_1)^* \subset Annf.$$

Par conséquent :

$$X_1(X \setminus X_1)^* \subset AnnH_1.$$

$b \Rightarrow c$: Si tous les mots de $X_1(X \setminus X_1)^*$ annulent les éléments de H_1, alors soit x une lettre de X_1 et soit w un mot de $x(X \setminus \{x\})^*$. On peut écrire $w = w_2x_1w_1$ où w_1 est un mot de $(X \setminus X_1)^*$, x_1 est une lettre de X_1 et w_2 est un mot de X^*. D'après l'hypothèse, on a :

$$x_1w_1 \in X_1(X \setminus X_1)^* \subset AnnH_1,$$

le mot $w = w_2x_1w_1$ annule les éléments de H_1, alors pour tout $x \in X_1$,on a l'inclusion suivante :

$$x(X \setminus \{x\})^* \subset AnnH_1.$$

$c \Rightarrow d$: C'est immédiat.

$d \Rightarrow e$: Si y_s, $s \in \pi$, est découplé par rapport à a^x, $x \in X_1$, alors prenons a^x la fonction nulle. Dans ce cas, pour tout mot $w \notin (X \setminus X_1)^*$, nous avons $\mathcal{E}_a(w)(t) = 0$. Par conséquent, si $q(0) \in Q$ est l'état initial de (S) alors nous avons :

$$y_s(t) = \sum_{w\in(X\setminus X_1)^*} (w * h_s)_{|q(0)}\mathcal{E}_a(w)(t).$$

Puisque y_s est l'Evaluation de de la série génératrice $\sigma_{q(0)}h_s$, alors pour tout entrée $a = (\, a^{x_0} \quad a^{x_1} \quad \ldots \quad a^{x_m}\,)$, et d'après le théorème d'unicité des Evaluations, nous avons :

$$\sigma_{q(0)}h_s = \sum_{w\in(X\setminus X_1)^*} (w * h_s)_{|q(0)}w.$$

Par conséquent, pour tout $f \in H_1$, on a :

$$\sigma f = \sum_{w\in(X\setminus X_1)^*} (w * f)w.$$

$e \Rightarrow a$: C'est immédiat •

5.3. Exemple

Dans cette exemple, les calculs ont été vérifiés avec les fonctions écrites en MACSYMA données au chapitre VI.

$$(S) \quad \begin{cases} \dot{q}_1(t) = b(t)q_1(t) \\ \dot{q}_2(t) = [a(t) - b(t)]q_2(t) \\ \dot{q}_3(t) = [c(t) - a(t)]q_3(t) \\ y_1(t) = q_1(t)q_2(t) \\ y_2(t) = q_2(t)q_3(t). \end{cases}$$

Soit $X = \{x_0, x_1, x_2, x_3\}$ un alphabet fini tel que :
. x_0 code la partie autonôme :$A_{x_0} = 0$,
. x_1 code l'entrée $a(t)$:$A_{x_1} = q_2\frac{\partial}{\partial q_2} - q_3\frac{\partial}{\partial q_3}$,
. x_2 code l'entrée $b(t)$:$A_{x_2} = q_1\frac{\partial}{\partial q_1} - q_2\frac{\partial}{\partial q_2}$,
. x_3 code l'entrée $c(t)$:$A_{x_3} = q_3\frac{\partial}{\partial q_3}$.
L'action de X^* sur $h_1 = q_1q_2$:

$$\forall w \in x_1^*, \quad w * h_1 = h_1 \quad \text{et} \quad \forall w \notin x_1^*, \quad w * h_1 = 0.$$

Seule la lettre x_1 est significative pour h_1 :

$$\mathrm{sgn}(h_1) = \{x_1\}, \qquad \text{et} \qquad \Psi_1 = 0.$$

L'action de X^* sur $h_2 = q_2q_3$:

$$\begin{array}{llll} \forall w \in \{x_2, x_3\}^*, & |w|_{x_2} = 2k, & k \in I\!N, & w * h_2 = \ \ h_2, \\ \forall w \in \{x_2, x_3\}^*, & |w|_{x_2} = 2k+1, & k \in I\!N, & w * h_2 = -h_2, \\ \forall w \notin \{x_2, x_3\}^*, & & & w * h_2 = \ \ 0. \end{array}$$

Les lettres x_2, x_3 sont significatives pour h_2 :

$$\mathrm{sgn}(h_2) = \{x_2, x_3\}, \qquad \text{et} \qquad \Psi_2 = 0.$$

La matrice de commande de (S) est :

$$\Omega = \begin{pmatrix} x_1 * \Delta \\ x_2 * \Delta \\ x_3 * \Delta \end{pmatrix} = \begin{pmatrix} x_1 * h_1 & x_1 * h_2 \\ x_2 * h_1 & x_2 * h_2 \\ x_3 * h_1 & x_3 * h_2 \end{pmatrix} = \begin{pmatrix} q_1q_2 & 0 \\ 0 & -q_2q_3 \\ 0 & q_2q_3 \end{pmatrix}.$$

La série génératrice associée à h_1 est

$$\sigma h_1 = \sum_{w \in x_1^*} (w * h_1)w = h_1 \sum_{w \in x_1^*} w.$$

Si $q(0)$ est l'état initial de (S), alors la sortie y_1 ne dépend que de l'entrée a :

$$y_1(t) = q_1(0)q_2(0)\exp(\int_0^t a(\tau)d\tau).$$

La série génératrice associée à h_2 est

$$\sigma h_2 = \sum_{w\in\{x_2,x_3\}^*} (w * h_2)w = h_2 \sum_{w\in\{x_2,x_3\}^*} (-1)^{|w|_{x_2}} w.$$

Si $q(0)$ est l'état initial de (S), alors la sortie y_2 ne dépend que des entrées b, c :

$$y_2(t) = q_2(0)q_3(0)\exp(\int_0^t [c(\tau) - b(\tau)]d\tau).$$

5.4. Modules et algèbres de découplage

Définition 5.4.1. : *Soit D_1 un sous module (resp. une sous algèbre de Lie) de $Lie < X >$. D_1 est un* **module** *(resp. une* **algèbre***)* **de découplage de H_1 par rapport à X_1** *(ou plus brièvement un module (resp. une algèbre) de découplage), si et seulement si D_1 vérifie les propriétés suivantes :*
(a) $X_1 \subset D_1$.
(b) $D_1 \subset AnnH_1$.
(c) $[X \setminus X_1, D_1] \subset D_1$.

Soit D_1 un module de découplage, alors D_1 vérifie (b) et (c). D'après le lemme 1.8.2.4., les éléments de D_1 annulent les éléments de $(X \setminus X_1)^* * H_1$. En particulier, puisque $X_1 \subset D_1$ (a), alors les éléments de X_1 annulent les éléments de $(X \setminus X_1)^* * H_1$.

Le plus petit sous module de $Lie < X >$ qui vérifie les conditions (a) et (c) que l'on note I_1, est engendré par les éléments

$$(ad_{x_{i_1}} \circ ad_{x_{i_2}} \circ \ldots \circ ad_{x_{i_k}})(x),$$

pour tout $x_{i_1}x_{i_2}\ldots x_{i_k} \in (X \setminus X_1)^*$ et pour tout $x \in X_1$.

I_1 est inclus dans tout module de découplage. C'est un module de découplage si et seulement si $I_1 \subset AnnH_1$.

D'après le théorème de l'élimination de M. Lazard ([V1]), **la plus petite sous algèbre de Lie de** $Lie < X >$ qui vérifie (a) et (c) est l'algèbre de Lie $\Im_1$ engendrée par I_1.

$\Im_1$ est inclus dans toute algèbre de découplage. C'est une algèbre de découplage si et seulement si $\Im_1 \subset AnnH_1$.

On peut vérifier aisément que $AnnH_1$ est une sous algèbre de Lie de $Lie < X >$:

$$\begin{aligned}\forall P,\ Q \in AnnH_1,\quad [P,Q] * H_1 &= (PQ) * H_1 - (QP) * H_1\\ &= P * (Q * H_1) - Q * (P * H_1)\\ &= \{0\}.\end{aligned}$$

Toute algèbre de découplage est incluse dans $AnnH_1$.

Proposition 5.4.1. : *Si* $[X \setminus X_1, AnnH_1] \subset AnnH_1$, *alors*

$$Ann((X \setminus X_1)^* * H_1) = AnnF_1 = AnnH_1.$$

Preuve : D'une part nous avons d'après le lemme 2.4.1. :

$$AnnH_1 \subset Ann((X \setminus X_1) * H_1).$$

D'autre part nous avons d'après le corollaire 2.4.2. :

$$Ann((X \setminus X_1)^* * H_1) = AnnH_1.$$

Alors nous avons successivement :

$$\begin{aligned} (X \setminus X_1)^* * H_1 &\supset \quad F_1 \quad \supset H_1, \\ Ann((X \setminus X_1)^* * H_1) &\subset AnnF_1 \subset AnnH_1 \\ Ann((X \setminus X_1)^* * H_1) &= AnnF_1 = AnnH_1 \quad \bullet \end{aligned}$$

Si $AnnH_1$ vérifie (a) et (c), alors $AnnH_1$ sera **la plus grande algèbre de découplage**, par conséquent c'est le **plus grand module de découplage**. Par conséquent, d'après le résultat précédent, $AnnF_1$ sera le plus grand module de découplage.

5.5. Théorème de caractérisation du découplage : (D. Claude, [C2], [C3]) *Les propriétés suivantes sont équivalentes :*

(a) H_1 *est découplé par rapport à* X_1.
(b) $X_1 \subset Ann((X \setminus X_1)^* * H_1)$.
(c) $\Im_1 \subset AnnH_1$.
(d) $I_1 \subset AnnH_1$.
(e) *Il existe un module de découplage.*
(f) $I_1(X \setminus X_1)^* \subset AnnH_1$.

Preuve :
$a \Rightarrow b$: D'après la proposition 5.2.1., nous avons :

$$X_1(X \setminus X_1)^* \subset AnnH_1.$$

Par conséquent :

$$X_1 \subset Ann((X \setminus X_1)^* * H_1).$$

$b \Rightarrow c$: Si $P \in \Im_1$, alors nous avons :

$$\mathrm{supp}(P) \subset X^* X_1 (X \setminus X_1)^*.$$

Si X_1 est inclus dans $Ann((X \setminus X_1)^* * H_1)$ alors nous avons :

$$X_1(X \setminus X_1)^* \subset AnnH_1,$$

par conséquent :

$$X^*X_1(X \setminus X_1)^* \subset AnnH_1,$$

d'où $AnnH_1$ contient le support de P, et finalement on a :

$$\Im_1 \subset AnnH_1.$$

$c \Rightarrow d$: I_1 est un cas particulier de $\Im_1$.

$d \Rightarrow e$: I_1 est le plus petit de découplage.

$e \Rightarrow f$: Si D_1 est un module de découplage, alors d'après la définition de I_1, nous avons :

$$I_1 \subset D_1 \subset AnnH_1,$$

et d'après le lemme 1.8.2.4., nous avons :

$$I_1(X \setminus X_1)^* \subset AnnH_1.$$

$f \Rightarrow a$: Puisque $X_1 \subset I_1$ alors $X_1(X \setminus X_1)^* \subset AnnH_1$, et d'après la proposition 5.2.1., H_1 est découplée par rapport à X_1 •

5.6. Le plus grand module de découplage

Lemme 5.6.1. :

$[X \setminus X_1, AnnF_1] \subset AnnF_1 \iff AnnF_1 \subset Ann((X \setminus X_1) * G_1)$.

Preuve : Puisque pour tout h_s de H_1 et pour tout entier $n \in [1..\Psi_s[$, nous avons :

$$(X \setminus X_1) * (x_0^n * H_1) = \{0\},$$

par conséquent :

$$Ann((X \setminus X_1) * F_1) = Ann(((X \setminus X_1) * G_1) \bigcup \{0\}) = Ann((X \setminus X_1) * G_1),$$

et il suffit d'appliquer le lemme 1.8.2.1. •

Lemme 5.6.2. : *Si D_1 est un module de découplage alors D_1 est inclus dans $AnnF_1$.*

Preuve : Etant donné $s \in \pi$ et $c \in D_1$, montrons par récurrence sur n que $cx_0^n * h_s$ est nulle :

. Pour $n = 0$, le résultat est immédiat car $c \in D_1 \subset AnnH_1$ (D_1 est un module de découplage).

. Nous supposons que pour tout ν, $0 \le \nu < n \le \Psi_s$, $cx_0^\nu * h_s$ est nulle.

. Pour $\nu = n$, nous avons :

$$cx_0^n * h_s = cx_0 * (x_0^{n-1} * h_s) = x_0 c * (x_0^{n-1} * h_s) - [x_0, c] * (x_0^{n-1} * h_s).$$

Puisque D_1 est stable par les crochets de Lie par rapport à $X \setminus X_1$ (D_1 est un module de découplage), alors nous avons $[x_0, c] \in D_1$. D'après l'hypothèse de récurence, nous avons

$$x_0 * (c * (x_0^{n-1} * h_s)) = [x_0, c] * (x_0^{n-1} * h_s) \equiv 0.$$

Par conséquent, D_1 est inclus dans $AnnF_1$ •

Le résultat du lemme 5.6.2. exprime que si $AnnF_1$ est un module de découplage, il sera le plus grand module de découplage. Nous donnons dans ce qui suit une condition nécessaire et suffisante pour qu'il le soit :

Proposition 5.6.1. : *$AnnF_1$ est un module de découplage si et seulement si les deux conditions suivantes sont satisfaites :*

$$. \forall h_s \in H_1, \quad \mathrm{sgn}(h_s) \subset (X_0 \setminus X_1).$$
$$. AnnF_1 \subset Ann((X \setminus X_1) * G_1).$$

Preuve :
Condition nécessaire : Si $AnnF_1$ est un module de découplage alors d'une part X_1 annule les éléments de F_1, et d'après le lemme 5.2.1., nous avons pour tout h_s de H_1, $\mathrm{sgn}(h_s) \subset (X_0 \setminus X_1)$. D'autre part $AnnF_1$ est stable par les crochets de Lie par rapport à $X \setminus X_1$, et lemme 5.6.1. permet de conclure.
Condition suffisante : Nous avons :
(a) $X_1 \subset AnnF_1$ (d'après le lemme 5.2.1.).
(b) $AnnF_1 \subset AnnH_1$ (puisque $H_1 \subset F_1$).
(c) $[X \setminus X_1, AnnF_1] \subset AnnF_1$
(puisque $AnnF_1 \subset Ann((X \setminus X_1) * G_1)$)) •

5.7. Etudes particulières

Nous examinons dans ce paragraphe les problèmes classiques tels que le rejet de perturbations ([C3], [C4]), le découplage k-bloc triangulaire ([N1]), le découplage k-bloc diagonal ([NS]), la commande non interactive ([GH]) :

5.7.1. Rejet de Perturbation

Etant donné $\mathbf{P} \subset X_0$, on pose $\mathbf{A} = X \setminus \mathbf{P}$. Les éléments de $\mathbf{P}$ représentent des **perturbations** et ceux de $\mathbf{A}$ représentent les **commandes admissibles** de H.

Définition 5.7.1.1. : *Le système (S) est* **découplé par rapport aux perturbations** *si et seulement si H est découplé par rapport à* $\mathbf{P}$ *:*

$$\mathbf{PA}^* \subset AnnH.$$

C'est-à-dire, pour toute observation $h_s \in H$, le support de σh_s est un sous langage de $\mathbf{A}^*$:

$$\forall h_s \in H, \quad \sigma h_s = \sum_{w \in \mathbf{A}^*} (w * h_s) w.$$

Par conséquent, pour tout $h_s \in H$, $\mathrm{sgn}(h_s) \subset \mathbf{A} \setminus \{x_0\}$. La matrice de commande Ω de tel système est de la forme :

$$\begin{pmatrix} \left[xx_0^{\Psi_s} * h_s\right]_{x\in\mathbf{A}\setminus\{x_0\}}^{s\in[1..p]} \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \Omega_a \\ \mathbf{0} \end{pmatrix}$$

et la sous matrice Ω_a est est appelée **la matrice de commande admissible.**

AnnF est le plus grand module de découplage de H si et seulement si :

$$\forall s \in [1..m], \quad \mathrm{sgn}(h_s) \subset \mathbf{A} \setminus \{x_0\} \quad \text{et} \quad AnnF \subset Ann(\mathbf{A} * G).$$

5.7.2. Découplage par k-bloc triangulaire

Etant donné $k \le \inf(m,p)$, nous avons une k-partition de X_0 :

$$X_0 = X_1 \oplus X_2 \oplus \ldots \oplus X_k,$$

et nous avons une k-partition de $[1..p]$:

$$[1..p] = \pi_1 \oplus \pi_2 \oplus \ldots \oplus \pi_k.$$

Nous posons :

$$\mathbf{P}_1 = \emptyset,$$
$$\mathbf{P}_j = \mathbf{P}_{j-1} \bigcup X_{j-1}, \; j \in [2..k].$$

Pour tout $j \in [1..k]$, on pose aussi :

$$\mathbf{A}_j = X \setminus \mathbf{P}_j,$$
$$\mathbf{H}_j = \{h_s\}_{s\in\pi_j},$$
$$\mathbf{G}_j = \{x_0^{\Psi_s} * h_s\}_{s\in\pi_j},$$
$$\mathbf{F}_j = \{x_0^n * h_s |\; 0 \le n \le \Psi_s\}_{s\in\pi_j}.$$

Les éléments de $\mathbf{A}_j$ représentent les commandes admissibles et les éléments de $\mathbf{P}_j$ sont des perturbations de $\mathbf{H}_j$.

Définition 5.7.2.1. : *Le système* (S) *est* **découplé k-block triangulaire** *si et seulement si pour tout entier* j *de* $[2..k]$, $\mathbf{H}_j$ *est découplé par rapport à* $\mathbf{P}_j$ *:*

$$\mathbf{P}_j\mathbf{A}_j^* \subset Ann\mathbf{H}_j.$$

C'est-à-dire, pour toute partition π_j, pour toute observation h_s de $\mathbf{H}_j$, le support de σh_s est un sous langage de $\mathbf{A}_j^*$:

$$\forall h_s \in H_j, \quad \sigma h_s = \sum_{w\in\mathbf{A}_j^*} (w * h_s)w.$$

Par conséquent, pour tout $h_s \in \mathbf{H}_j$, $\mathrm{sgn}(h_s) \subset \mathbf{A}_j \setminus \{x_0\}$. La matrice de commande Ω de tel système est de la forme :

$$\begin{pmatrix} [xx_0^{\Psi_s} * h_s]_{x\in\mathbf{A}_1\setminus\{x_0\}}^{s\in\pi_1} & 0 & \cdots & 0 \\ [xx_0^{\Psi_s} * h_s]_{x\in\mathbf{A}_2\setminus\{x_0\}}^{s\in\pi_1} & [xx_0^{\Psi_s} * h_s]_{x\in\mathbf{A}_2\setminus\{x_0\}}^{s\in\pi_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ [xx_0^{\Psi_s} * h_s]_{x\in\mathbf{A}_k\setminus\{x_0\}}^{s\in\pi_1} & [xx_0^{\Psi_s} * h_s]_{x\in\mathbf{A}_k\setminus\{x_0\}}^{s\in\pi_2} & \cdots & [xx_0^{\Psi_s} * h_s]_{x\in\mathbf{A}_k\setminus\{x_0\}}^{s\in\pi_k} \end{pmatrix}$$

Pour tout entier $j \in [2..k]$, $Ann\mathbf{F}_j$ est le plus grand module de découplage de $\mathbf{H}_j$ si et seulement si :

$$\forall s \in \pi_j, \quad \mathrm{sgn}(h_s) \subset \mathbf{A}_j \setminus \{x_0\} \quad \text{et} \quad Ann\mathbf{F}_j \subset Ann(\mathbf{A}_j * \mathbf{G}_j).$$

5.7.3. Découplage par k-bloc diagonal

Etant donné $k \leq \inf(m,p)$, nous avons une k-partition de X_0 :

$$X_0 = X_1 \oplus X_2 \oplus \ldots \oplus X_k,$$

et nous avons une k-partition de $[1..p]$:

$$[1..p] = \pi_1 \oplus \pi_2 \oplus \ldots \oplus \pi_k.$$

Pour tout $j \in [1..k]$, on pose :

$$\begin{aligned} \mathbf{P}_j &= X_0 \setminus X_j, \\ \mathbf{A}_j &= X \setminus \mathbf{P}_j, \\ \mathbf{H}_j &= \{h_s\}_{s\in\pi_j}, \\ \mathbf{G}_j &= \{x_0^{\Psi_s} * h_s\}_{s\in\pi_j}, \\ \mathbf{F}_j &= \{x_0^n * h_s \mid 0 \leq n \leq \Psi_s\}_{s\in\pi_j}. \end{aligned}$$

Les éléments de $\mathbf{A}_j$ représentent les commandes admissibles et ceux de $\mathbf{P}_j$ sont des perturbations de $\mathbf{H}_j$.

Définition 5.7.3.1. : *Le système (S) est k-***block diagonal découplé** *si et seulement si pour tout entier j de $[1..k]$, $\mathbf{H}_j$ est découplé par rapport à $\mathbf{P}_j$:*

$$\mathbf{P}_j\mathbf{A}_j^* \subset Ann\mathbf{H}_j.$$

C'est-à-dire, pour toute partition π_j, pour toute observation h_s de $\mathbf{H}_j$, le support de σh_s est un sous langage de $\mathbf{A}_j^*$:

$$\forall h_s \in H_j, \quad \sigma h_s = \sum_{w\in\mathbf{A}_j^*\setminus\{x_0\}} (w * h_s)w.$$

Par conséquent, pour tout $h_s \in \mathbf{H}_j$, $\mathrm{sgn}(h_s) \subset \mathbf{A}_j \setminus \{x_0\}$. La matrice de commande Ω de tel système est k-block diagonal :

$$\begin{pmatrix} \left[xx_0^{\Psi_s} * h_s\right]_{x\in\mathbf{A}_1\setminus\{x_0\}}^{s\in\pi_1} & 0 & \dots & 0 \\ 0 & \left[xx_0^{\Psi_s} * h_s\right]_{x\in\mathbf{A}_2\setminus\{x_0\}}^{s\in\pi_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \left[xx_0^{\Psi_s} * h_s\right]_{x\in\mathbf{A}_k\setminus\{x_0\}}^{s\in\pi_k} \end{pmatrix}$$

Pour tout entier j de $[1..k]$, $Ann\mathbf{F}_j$ est le plus grand module de découplage de $\mathbf{H}_j$ si et seulement si :

$$\forall s \in \pi_j, \quad \mathrm{sgn}(h_s) \subset \mathbf{A}_j \setminus \{x_0\} \quad \text{et} \quad Ann\mathbf{F}_j \subset Ann(\mathbf{A}_j * \mathbf{G}_j).$$

5.7.4. Système non interactif

C'est une application du cas précédent avec $k = p = m$:

Définition 5.7.4.1. : *Le système* (S) *est non* **système non interactif** *si et seulement si pour tout* $s \in [1..p]$, h_s *est découplé par rapport à* $X_0 \setminus \{x_s\}$:

$$(X_0 \setminus \{x_s\})\{x_0, x_s\}^* \subset Annh_s.$$

C'est-à-dire, pour toute observation $h_s \in H$, le support de σh_s est un sous langage de $\{x_0, x_s\}^*$ (et non pas seulement de x_s^* !) :

$$\forall h_s \in H, \quad \sigma h_s = \sum_{w\in\{x_0,x_s\}^*} (w * h_s)w.$$

Dans ce cas, d'après la définition des nombres caractéristiques (supposé définis), nous avons :

$$\forall s \in [1..p], \quad x_s x_0^{\Psi_s} * h_s \not\equiv 0.$$

Par conséquent, pour tout $s \in [1..p]$, $\mathrm{sgn}(h_s) = \{x_s\}$. Et la matrice de commande Ω de tel système est diagonal :

$$\begin{pmatrix} x_1x_0^{\Psi_1} * h_1 & 0 & \dots & 0 \\ 0 & x_2x_0^{\Psi_2} * h_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & x_mx_0^{\Psi_m} * h_m \end{pmatrix}$$

Les termes diagonaux de la matrice de commande Ω ne s'annulent pas, par conséquent, Ω est de rang maximal : $\mathrm{rank}\Omega = m$.

Pour tout $s \in [1..p]$, $Ann\{x_0^n * h_s | 0 \leq n \leq \Psi_s\}$ est le plus grand module de découplage de h_s si et seulement si :

$$\mathrm{sgn}(h_s) = \{x_s\}$$
$$\text{et} \quad Ann\{x_0^n * h_s| \quad 0 \leq n \leq \Psi_s\} \subset Ann\{x_0x_0^{\Psi_s} * h_s, x_sx_0^{\Psi_s} * h_s\}.$$

Remarque :

Lorsque H_1 est découplé par rapport à à X_1, d'après le théorème de caractérisation du découplage, nous avons $X_1(X \setminus X_1)^* \in AnnH_1$. En particulier, pour tout $s \in \pi$ et pour tout $x \in X_1$, $xx_0^{\Psi_s} * h_s \equiv 0$, la matrice extraite $\left[\Omega_x^s\right]_{x\in X_1}^{s\in\pi}$ est nulle (la réciproque est fausse en général).

Inversement, si $\left[\Omega_x^s\right]_{x\in X_1}^{s\in\pi}$ est identiquement nulle, alors pour tout $s \in \pi$, $\mathrm{sgn}(h_s) \subset X_0 \setminus X_1$. En outre, si nous avons l'inclusion

$$AnnF_1 \subset Ann((X \setminus X_1) * G_1)$$

alors d'après la proposition 5.6.1., $AnnF_1$ sera le plus grand module de découplage. Par conséquent, d'après le théorème de caractérisation du découplage, H_1 est découplé par rapport à X_1.

Proposition 5.7.1. : *Si la matrice de commande Ω d'un système dynamique non linéaire est une matrice constante ($\Omega \in \mathcal{M}_{m,p}(\mathbb{R})$) alors les sorties $\{y_s\}_{s\in\pi}$ sont découplées par rapport aux entrées $\{a^x\}_{x\in X_1}$ si et seulement si la matrice extraite $\left[\Omega_x^s\right]_{x\in X_1}^{s\in\pi}$ est nulle.*

5.8. Découplage par bouclage statique par retour d'état

Soit $X = \{x_0, x_1, \ldots, x_m\}$ (resp. $\hat{X} = \{\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{\hat{m}}\}$) un alphabet fini. Nous posons $X_0 = X \setminus \{x_0\}$ (resp. $\hat{X}_0 = \hat{X} \setminus \{\hat{x}_0\}$).

5.8.1. Bouclage statique par retour d'état

Lorsque (S) ne présente pas les propriétés qu'on souhaite (dans notre étude c'est le découplage de H_1 par rapport à X_1), on lui applique une transformation du type **bouclage statique par retour d'état :**

$$(*) \quad \forall x \in X, \quad a^x = \sum_{\hat{x}\in\hat{X}} a^{\hat{x}} \Phi_{\hat{x}}^x,$$

où :

. Pour tout $x \in X$ et pour tout $\hat{x} \in \hat{X}$, $\Phi_{\hat{x}}^x$ est une application analytique de $C^\omega(Q)$.

. Pour tout $x \in X$, $\Phi_{\hat{x}_0}^x$ est une application constante, et égale à 0 et $\Phi_{\hat{x}_0}^{x_0}$ est égale à 1.

. Pour tout $\hat{x} \in \hat{X}$, $a^{\hat{x}}$ est une application continue par morceau sur $\mathbb{R}_+$. En particulier, $a^{\hat{x}_0}$ est une application constante et égale à 1. On note $\hat{a}$

le vecteur $(a^{\hat{x}_0} \quad a^{\hat{x}_1} \quad \ldots \quad a^{\hat{x}_{\hat{m}}})$. Nous avons :

$$\begin{aligned}
\dot{q}(t) &= \sum_{x\in X} a^x(t)A_x(q) \\
&= \sum_{x\in X}\Big(\sum_{\hat{x}\in\hat{X}} a^{\hat{x}}(t)\Phi_{\hat{x}}^x(q)\Big)A_x(q) \\
&= \sum_{\hat{x}\in\hat{X}} a^{\hat{x}}(t)\Big(\sum_{x\in X}\Phi_{\hat{x}}^x(q)A_x(q)\Big) \\
&= \sum_{\hat{x}\in\hat{X}} a^{\hat{x}}(t)A_{\hat{x}}(q),
\end{aligned}$$

où pour tout lettre $\hat{x}$ de $\hat{X}$,

$$A_{\hat{x}} = \sum_{x\in X}\Phi_{\hat{x}}^x A_x.$$

Le système bouclé obtenu $(\hat{S})$ est encore un système de (NL) :

$$(\hat{S}) \quad \begin{cases} \dot{q}(t) = \hat{a}(t).\hat{A}(q) \\ y(t) = h(q(t)) \end{cases}$$

où $\hat{A}$ et Φ sont les matrices suivantes :

$$\hat{A} = \begin{pmatrix} A_{\hat{x}_0} \\ A_{\hat{x}_1} \\ \vdots \\ A_{\hat{x}_m} \end{pmatrix},$$

et

$$\Phi = \begin{pmatrix} \Phi_{\hat{x}_0}^{x_0} & \Phi_{\hat{x}_0}^{x_1} & \ldots & \Phi_{\hat{x}_0}^{x_m} \\ \Phi_{\hat{x}_1}^{x_0} & \Phi_{\hat{x}_1}^{x_1} & \ldots & \Phi_{\hat{x}_1}^{x_m} \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{\hat{x}_{\hat{m}}}^{x_0} & \Phi_{\hat{x}_{\hat{m}}}^{x_1} & \ldots & \Phi_{\hat{x}_{\hat{m}}}^{x_m} \end{pmatrix} = \begin{pmatrix} 1 & \alpha^{x_1} & \ldots & \alpha^{x_m} \\ 0 & \beta_{\hat{x}_1}^{x_1} & \ldots & \beta_{\hat{x}_1}^{x_m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \beta_{\hat{x}_{\hat{m}}}^{x_1} & \ldots & \beta_{\hat{x}_{\hat{m}}}^{x_m} \end{pmatrix} = \begin{pmatrix} 1 & \alpha \\ 0 & \beta \end{pmatrix},$$

et en notation matricielle, nous avons $a = \hat{a}\Phi$ et $\hat{A} = \Phi A$.

Définition 5.8.1.1. : *On appelle* Φ *la* **matrice de bouclage**.
Le bouclage par retour d'état sera dit **régulier** *si et seulement si la matrice de bouclage* Φ *est de rang maximal, ou équivalemment* β *est de rang maximal.*

Lorsque le bouclage par retour d'état est régulier ($m = \hat{m}$) alors Φ^{-1} existe et vaut

$$\Phi^{-1} = \begin{pmatrix} 1 & -\alpha\beta^{-1} \\ 0 & \beta^{-1} \end{pmatrix}.$$

Nous notons $\{\hat{\Psi}_s\}_{1\leq s\leq p}$ les nombres caractéristiques du système bouclé $(\hat{S})$. Lorsqu'ils sont tous définis, on note $\hat{\Omega}$ la matrice de bouclage de $(\hat{S})$, $\begin{pmatrix} \hat{x}_1 * \hat{\Delta} \\ \hat{x}_2 * \hat{\Delta} \\ \vdots \\ \hat{x}_{\hat{m}} * \hat{\Delta} \end{pmatrix}$, et $\hat{\Gamma}$ la matrice $\begin{pmatrix} \hat{x}_0 * \hat{\Delta} \\ \hat{\Omega} \end{pmatrix}$, où $\hat{\Delta}$ est la matrice

$$\begin{pmatrix} \hat{x}_0^{\hat{\Psi}_1} * h_1 & \hat{x}_0^{\hat{\Psi}_2} * h_2 & \dots & \hat{x}_0^{\hat{\Psi}_p} * h_p \end{pmatrix},$$

et pour tout $\hat{x} \in \hat{X}$, $\hat{x} * \hat{\Delta}$ représentents la matrice

$$\begin{pmatrix} \hat{x}\hat{x}_0^{\hat{\Psi}_1} * h_1 & \hat{x}\hat{x}_0^{\hat{\Psi}_2} * h_2 & \dots & \hat{x}\hat{x}_0^{\hat{\Psi}_p} * h_p \end{pmatrix}.$$

Lemme 5.8.1.1. : *Soit f une application analytique de $C^\omega(Q)$. Pour tout lettre $\hat{x} \in \hat{X}$, $\hat{x} * f = \sum_{x\in X} \Phi_{\hat{x}}^x x * f$.*

<u>*Preuve*</u> : Il suffit de remarquer que

$$\hat{x} * f = A_{\hat{x}} \circ f = \left(\sum_{x\in X} \Phi_{\hat{x}}^x A_x\right) \circ f = \sum_{x\in X} \Phi_{\hat{x}}^x A_x \circ f = \sum_{x\in X} \Phi_{\hat{x}}^x \, x * f \quad \bullet$$

5.8.2. Les nombres caractéristiques et le bouclage statique par retour d'état

Nous allons prouver que **les nombres caractéristiques sont invariants par bouclage statique par retour d'état** :

Lemme 5.8.2.1. : *Si les nombres caractéristiques $\{\Psi_s\}_{1\leq s\leq p}$ de (S) sont tous définis alors :*

(i) $\forall s \in [1..p]$, $\forall n \in [0..\Psi_s]$, $\hat{x}_0^n * h_s = x_0^n * h_s$.

(ii) $\forall s \in [1..p]$, $\forall n \in [0..\Psi_s[$, $\forall \hat{x} \in \hat{X}_0$, $\hat{x}\hat{x}_0^n * h_s = 0$.

<u>*Preuve*</u> :

(i) Le résultat est immédiat pour $n = 0$. Etant donné $n \leq \Psi_s$, supposons que le résultat soit vrai pour tout ν de $[1..n[$. Pour $\nu = n$, d'après l'hypothèse de récurrence, nous avons :

$$\hat{x}_0^n * h_s = \hat{x}_0 * (\hat{x}_0^{n-1} * h_s) = \hat{x}_0 * (x_0^{n-1} * h_s).$$

d'après le lemme 5.8.1.1., on obtient :

$$\hat{x}_0^n * h_s = \sum_{x \in X} \Phi_{\hat{x}_0}^x x x_0^{n-1} * h_s = x_0^n * h_s + \sum_{x \in X_0} \alpha^x x x_0^{n-1} * h_s.$$

Et d'après la définition de Ψ_s, nous avons le résultat.

(ii) Soit $\hat{x}$ une lettre de $\hat{X}_0$. D'après (i), nous avons

$$\hat{x}\hat{x}_0^n * h_s = \hat{x} * (x_0^n * h_s).$$

d'après lemme 5.8.1.1., on obtient :

$$\hat{x} * (x_0^n * h_s) = \sum_{x \in X} \Phi_{\hat{x}}^x x x_0^n * h_s = \sum_{x \in X_0} \beta_{\hat{x}}^x x x_0^n * h_s.$$

Et d'après la définition de Ψ_s, nous avons le résultat •

Corollaire 5.8.2.1. : *Si pour $s \in [1..p]$, Ψ_s n'est pas défini, alors :*

$$(i)\ \forall n \geq 0, \qquad \hat{x}_0^n * h_s = x_0^n * h_s.$$
$$(ii)\ \forall n \geq 0, \quad \forall \hat{x} \in \hat{X}_0, \quad \hat{x}\hat{x}_0^n * h_s = 0.$$

Corollaire 5.8.2.2. : *Les nombres caractéristiques ne peuvent pas décroître par bouclage statique par retour d'état :*

$$\forall s \in [1..p], \quad \Psi_s \leq \hat{\Psi}_s.$$

Ils ne changent pas dans le cas d'un bouclage statique par retour d'état régulier.

Théorème 5.8.2.1. : *Si les nombres caractéristiques $\{\Psi_s\}_{1 \leq s \leq p}$ de (S) sont définis, et si le bouclage par retour d'état est régulier alors $\hat{\Gamma} = \Phi\Gamma$.*

Preuve : Si le bouclage est régulier, d'après le corollaire 5.8.2.2., nous avons :

$$\Delta = \hat{\Delta},$$

et dans ce cas, nous avons aussi pour tout $\hat{x} \in \hat{X}$:

$$\hat{x} * \hat{\Delta} = \hat{x} * \Delta = \sum_{x \in X} (\Phi_{\hat{x}}^x x) * \Delta = \sum_{x \in X} \Phi_{\hat{x}}^x x * \Delta,$$

c'est-à-dire $\hat{\Gamma} = \Phi\Gamma$ •

5.8.3. Découplage par bouclage statique par retour d'état

Si les nombres caractéristiques $\{\Psi_s\}_{1\leq s\leq p}$ de (S) sont définis, et si le bouclage par retour d'état est régulier alors d'après le théorème 5.8.2.1., nous avons :

$$\begin{pmatrix} \hat{x}_0 * \hat{\Delta} \\ \hat{\Omega} \end{pmatrix} = \begin{pmatrix} 1 & \alpha \\ 0 & \beta \end{pmatrix} \begin{pmatrix} x_0 * \Delta \\ \Omega \end{pmatrix}.$$

Par conséquent, lorsque le bouclage est régulier nous avons le système d'équations suivant :

$$(**) \quad \begin{cases} \hat{x}_0 * \hat{\Delta} - x_0 * \Delta = \alpha\Omega \\ \hat{\Omega} = \beta\Omega \end{cases}$$

Définition 5.8.3.1. : *Un système (S) de (NL) est* **découplable par bouclage statique par retour d'état** *si et seulement si il existe un bouclage régulier par retour d'état tel que le système $(\hat{S})$ bouclé admette l'ensemble $\{y_s\}_{s\in\hat{\pi}}$ de sorties découplé par rapport à l'ensemble $\{a^{\hat{x}}\}_{\hat{x}\in\hat{X}_1}$ d'entrées.*

D'après le résultat précédent, nous déduisons le théorème suivant :

Théorème 5.8.3.1. : *Un système (S) de (NL) est* **découplable par bouclage statique par retour d'état** *si et seulement si le système d'équations $(**)$ admet une solution en α et β tel que le système bouclé $(\hat{S})$ admet l'ensemble $\{y_s\}_{s\in\hat{\pi}}$ de sorties découplé par rapport à l'ensemble $\{a^{\hat{x}}\}_{\hat{x}\in\hat{X}_1}$ d'entrées.*

Une telle solution existe si et seulement si Ω admet un inverse à droite noté par Ω^d. Dans cette condition, nous avons :

$$\begin{cases} \alpha = (\hat{x}_0 * \hat{\Delta} - x_0 * \Delta)\Omega^d \\ \beta = \hat{\Omega}\Omega^d \end{cases}$$

CHAPITRE 6

VERS UNE IMPLANTATION EN MACSYMA

6.0. Introduction

Dans les chapitres précédents, nous nous sommes forcés de dégager des outils et des concepts utiles (l'action d'une lettre sur une fonction analytique, annulateurs, transformation d'Evaluation) pour formuler de façon cohérente des questions posées en théorie des systèmes (le calcul symbolique pour les systèmes dynamiques non linéaires, les équations différentielles non linéaires en régime forcé, le découplage). Nous avons montré comment les objets du monoïde libre facilitent grandement des écritures mathématiques. Ils permettent de mettre en évidence la correspondance entre le produit de Cauchy des séries génératrices et certaines convolutions de signaux, et ils raccourcissent de nombreuses démonstrations. Nous avons vu aux chapitres II et IV que les lettres de l'alphabet de codage de M. Fliess jouent le rôle d'opérateurs d'intégrations et qu'elles jouent, aux chapitres I et V, le rôle d'opérateurs différentiels. Chacun de ces rôles est spécifé chaque fois que nous faisons appel à la transformation d'Evaluation de chaque lettre ou à l'action de chaque lettre sur une fonction analytique. Ces deux rôles ont été utilisés conjointement, au chapitre III, pour obtenir le comportement temporel (sous certaine condition de convergence) des systèmes dynamiques non linéaires, par intermédiaire de la formule fondamentale de M. Fliess, puis en déduire les développements de Taylor des noyaux de Volterra grâce à un théorème de graduation des séries de Chen. Ainsi en utilisant le monoïde libre engendré par cet alphabet de codage, nous avons bénéficié des études systématiques de l'algèbre de Lie libre ([F2], [J1], [R2], [V1]). L'utilisation des objets du monoïde libre est aussi nécessaire pour une implantation concise grâce au système de calcul formel, puisque ces écritures se traduisent de façon naturelle en écritures algorithmiques. Elles conduisent aussi à des méthodes élégantes d'implantation (basées alors sur l'analyse syntaxique des expressions) qui utilisent des représentations non numériques et des manipulations algébriques de base (voir les travaux récents de M.P. Delest [D1], de C. Hespel & G. Jacob [HJ2], P.V. Koseleff [K5] et de G. Jacob & N.E. Oussous [JO2]). C'est d'ailleurs le développement de tels outils et de telles implantations qui constitue l'axe principal de recherche de l'équipe $\mathcal{SNCF}$. du $\mathcal{LIFL}$.

Nous allons présenter brièvement le système de calcul formel MACSYMA. Nous indiquerons nos choix de structures de données en MACSYMA, en particulier nos choix de représentation des séries formelles en variables non commutatives par les "arbres binaires". Nous terminons par une description de nos algorithmes récursifs, en particulier l'algorithme d'Evaluation utilisant les "λ−notations" de MACSYMA. Nos programmes (écrits en MACSYMA SUN-3$^{\text{TM}}$, Release 309.6) exploitent les capacités du calcul matriciel, d'intégration, de sommation et de dérivation formelle de ce système de calcul formel, ils peuvent être aussi bien implantés sur d'autres systèmes. Notre choix de MACSYMA est dû à la disponibilité de MACSYMA au $\mathcal{L.I.F.L.}$ lorsque nous commençons ce travail.

6.1. Brève présentation de MACSYMA

MACSYMA est actuellement le plus complet des systèmes de calcul formel, il peut effectuer notamment des intégrations, sommations et dérivations formelles, intégration des équations différentielles linéaires, calcul matriciel. Il a été créé au Massachusetts Institue of Technology (MIT) par C. Engelman, R.J. Fateman, W. Martin, J. Moses ... La première version est apparue en 1971. Depuis il s'est amélioré et enrichi continuellement. Il comprend plus de cinq cents opérateurs ([M3]), cela représente environ 3.000 subroutines de LISP (Mac-LISP) ou encore 300.000 lignes de codes LISP compilés (équivalent à 100 homme-années de développement et de tests). MACSYMA est diffusé sur les gros ordinateurs comme DPS 8 avec le système Multics, VAX avec les systèmes VMS et Unix, ..., et les stations de travail comme SUN. L'apparition de Common-LISP et la création d'une norme ISO (pour règler le problème des implantations très disparates de LISP avec de nombreux dialectes comme Mac-LISP, Le-LISP, Franz LISP, ... qui limitent donc la diffusion de MACSYMA) va peut être rendre MACSYMA plus disponible sur d'autres machine (version Maxima).

Puisque MACSYMA est écrit en LISP, langage bien adapté à la manipulation d'expressions symboliques, l'utilisation de MACSYMA revient donc à faire tourner un gros programmme LISP muni d'une interface plus conviviale pour l'utilisateur. Lorsque MACSYMA (en mode interactif) attend une commande de l'utilisateur, il se trouve dans une boucle

DISPLAY(MEVAL(PARSE(READ(.))))

dont on notera une ressemblance avec la fameuse boucle

PRINT(EVAL(READ(.)))

de LISP. En effet, la fonction READ est la même dans les deux cas, la fonction PARSE affectue l'analyse syntaxique d'une expression puis la traduit en LISP, la fonction MEVAL évalue une expression LISP par l'évaluateur MACSYMA, et la fonction DISPLAY affiche le résultat, en deux dimensions s'il y a lieu.

Ainsi, il est conseillé de programmer en MACSYMA comme en LISP, c'est-à-dire de considérer MACSYMA comme un langage fonctionnel en utilisant, quand c'est possible, les définitions récursives de fonction. Mais MACSYMA possède aussi des structures de contrôle du type ALGOL; c'est pour cette raison, pour éviter l'"utilisation anarchique des parenthèses", nous adoptons la manière de programmer suivante :

Penser en LISP, écrire en ALGOL.

De cette manière, nous gagnons notablement sur la lisibilité, le nombre de lignes de codes en MACSYMA et le temps d'exécution. Notons que l'on peut aussi générer du FORTRAN à partir de MACSYMA. Ainsi on peut passer du calcul algébrique au calcul numérique en considérant MACSYMA comme un préprocesseur pour le calcul numérique. C'est le cas lorsqu'il faut effectuer un grand nombre de manipulations symboliques (intégrations, sommations et dérivations, ...) d'avant d'écrire le code en FORTRAN.

Les principales possibilités de MACSYMA sont les mêmes que celles des autres systèmes de calcul formel (quelque fois il est plus performant, [F1]). Au lieu de citer les qualités de ce système (car il existe déjà d'exellents ouvrages pour mieux comprendre le fonctionnement et les principaux algorithmes de MACSYMA : [DST], [F1], [PW]), nous allons montrer quelques points d'insatisfaction (peut être minimes devant les qualités) relevant de nos expériences avec MACSYMA.

Tout d'abord, MACSYMA sur SUN 3 est très gourmand en mémoire, nous arrivons très vite à saturer notre machine (voir exemple 4.2.3.). Il semble très lent, vu la capacité intrinsèque de cette machine. Ceci est justifié par la faible place disponible dans la configuration utilisée pour la pile LISP ([MR1]). Aussi MACSYMA passe beaucoup de temps pour effectuer le "garbage collector".

Nous sommes souvent génés par le manque de possibilités pour définir des types en MACSYMA, car MACSYMA n'est pas un langage typé contrairement à des langages plus récents comme SCRATCHPAD (écrit en LISP et installé sous VM-CMS d'IBM et AIX du IBM PC/RT, [S7]) par exemple. Ainsi, le manque d'objet du type "fonction" nous conduit à utiliser la λ−notation de MACSYMA pour représenter les fonctions et pour forcer nos fonctions à retourner un résultat de type "fonction" puis passer ce résultat en paramètre pour d'autres fonctions (des fonctionnelles). Il n'est pas possible non plus de fixer un domaine de définition pour une variable ou une fonction.

La connaissance de la représentation interne des fonctions et des expressions (les λ−notations, les listes, les arbres n-aires, ...) est nécessaire pour obtenir des algorithmes performants. L'utilisation abusive de cette possibilité peut pousser très vite aux bricolages (c'est ce que nous allons faire mais pour la bonne cause !).

On peut espérer que le système SCRATCHPAD en cours de réalisation chez IBM à Yorktown permettra de pallier ces inconvénients. Car dans ce système, les auteurs ont voulu éviter le "mélange" entre les "structures mathématiques" et les " représentations informatiques" en introduisant les "types abstraits" (un type abstrait est un ensemble de valeurs et de fonctions pouvant être effectuées sur ces valeurs) et des "fonctions génériques" (une fonction est appelée de façon générique si un même nom peut opérer sur des objets de types différents, l'opération à effectuer est alors déterminée par les types des arguments et des résultats).

6.2. Choix de la structure de données

Comme LISP, les structures principales de données de MACSYMA sont les "atomes" (qui correspondent aux nombres et aux "identificateurs" des langages du type ALGOL) et des "listes" que l'ont peut considérer comme des "tableaux dynamiques" dont la taille resterait indéterminée et varierait au cours de l'exécution (le gestionnaire de la mémoire dynamique se chargera de l'attribution et de la récupération des places de la mémoire). Une liste peut être vide, représentée par "[]", elle peut contenir des atomes et aussi bien d'autres listes. De cette façon, on peut obtenir des structures de plus en plus compliquées comme celle des arbres n-aires :

$$\begin{array}{ccccc} & & \phi & & \\ & \swarrow & \downarrow & \searrow & \\ op_1 & \ldots & op_i & \ldots & op_n. \end{array}$$

Ces arbres n-aires sont utilisés pour représenter les fonctions (l'opérateur pricipal) à plusieur arguments (les opérandes), soit :

$$\phi(op_1, \ldots, op_i, \ldots, op_n).$$

Chaque argument op_i peut être aussi une autre fonction à plusieurs arguments. On peut accéder à l'opérateur pricipal (ou la racine d'un arbre) et l'un des opérandes (ou des sous arbres) par la fonction "**inpart**" de MACSYMA. De cette structure d'arbres n-aires, nous allons déduire une structure d'arbres binaires pour représenter des séries formelles en variables non commutatives.

Pour représenter une lettre d'un alphabet Z, nous pouvons utiliser la fonction "**concate**" de MACSYMA : si i est un entier alors **concate**($'z, i$) est l'atome zi représentant la lettre z_i.

On peut utiliser les lettres z_i de l'alphabet Z pour indicer les tableaux. Les exemples traités au paragraphe 6.3. et au paragraphe 6.8. montrent comment nous indiçons les champs de vecteurs et les entrées (en ALGOL, cela revient à considérer l'alphabet Z comme un interval de type énuméré).

Pour une représentation interne, une série formelle non commutative S peut être un élément d'un anneau unitaire commutatif A (par exemple $A = \mathbb{R}$), ou bien une lettre d'un alphabet Z, ou bien un élément de la forme

$$S_1 \ \phi \ S_2,$$

où ϕ est l'un des "opérateurs binaires"

$$\{*,\ +,\ .,\ \wedge\wedge,\ \wedge *\}.$$

. Pour $\phi =$ "$*$" alors S_1 est un élément de A et S_2 est une série formelle (cas d'une multiplication par un scalaire).

. Pour $\phi =$ " $+$ " alors S_1 et S_2 sont des séries formelles (cas d'une addition de deux séries formelles).

. Pour $\phi =$ "." alors S_1 et S_2 sont des séries formelles (cas d'une multiplication non commutative de deux séries formelles).

. Pour $\phi =$ "$\wedge\wedge$" alors S_2 est un entier et S_1 est une série formelle (cas d'une puissance non commutative).

. Pour $\phi =$ "$\wedge *$ " alors S_2 est un entier et S_1 est une série formelle propre (cas d'une d'une fraction rationnelle en variables non commutatives).

Remarque : *Pour que MACSYMA puisse reconnaître "$\wedge *$ " comme un opérateur binaire il est nécessaire de déclarer*

nary("$\wedge *$ ").

Une telle expression peut être représentée par un arbre binaire :

$$\begin{array}{ccc} & \phi & \\ \swarrow & & \searrow \\ S_1 & & S_2 \end{array}$$

L'analyse (récursive) de ces arbres, comme de leurs sous arbres, peut être décrite par les fonctions **Root** (pour la "racine"), **Left** et **Right** (pour les deux "opérandes" gauche et droit).

Examinons l'exemple suivant :

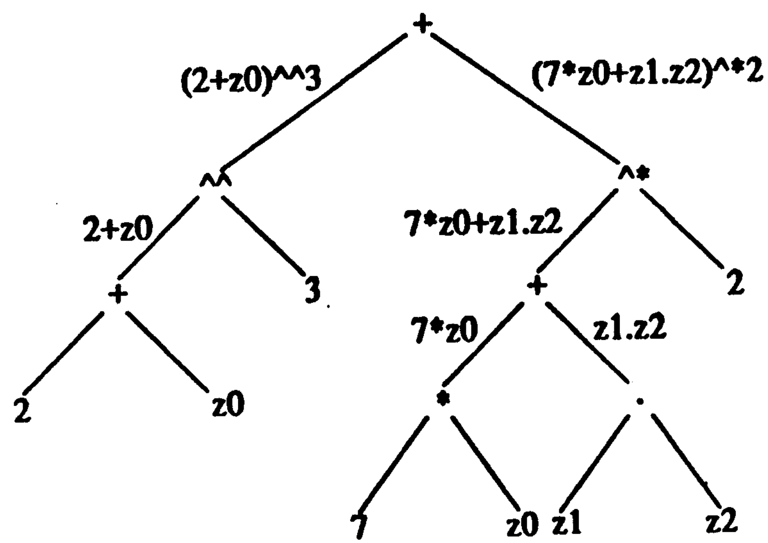
$$(2 + z_0)^3 + (7z_0 + z_1 z_2)^{*2}$$

dont l'arbre binaire corespondant est :

ANALYSE ARBORESCENTE

DE LA FRACTION RATIONNELLE

(2+z0)^^3+(7*z0+z1.z2)^*2




```
(c5) S:(2+z0)^^3+(7*z0+z1 . z2)^*2;
                                                    <3>
(d5)          (z1 . z2 + 7 z0) ^* 2 + (z0 + 2)

(c6) Root(S);

(d6)                                  +

(c7) Left(S);
                                         <3>
(d7)                             (z0 + 2)

(c8) Right(S);

(d8)                      (z1 . z2 + 7 z0) ^* 2

(c9) Root(Left(S));

(d9)                                  ^^

(c10) Left(Left(S));

(d10)                               z0 + 2

(c11) Right(Left(S));


(d11)                                 3
```

6.3. Application à l'étude du découplage

Nous avons utilisé la représentation arborescente des séries formelles non commutatives pour écrire la fonction "**Action(P,f)**" calculant l'action (voir 1.8.1.) d'un polynôme **P** sur la fonction analytique **f**. Ensuite, grâce à cette fonction, nous avons écrit la procédure "**Control_matrix()**" contruisant la matrice de commande Ω et la matrice Γ d'un système dynamique (voir définition 5.1.3.). Cette procédure utilise la liste **Psi** contenant les nombres caractéristiques $\{\psi_s\}_{1\leq s\leq p}$ d'un système dynamique (voir définition 5.1.2.), et aussi p listes **sgn** de lettres significatives des fonctions d'observation $\{h_s\}_{1\leq s\leq p}$, et enfin la liste **Gamma** représentant la première ligne de la matrice Γ, c'est-à-dire $\left(z_0^{\Psi_1+1} * h_1 \quad z_0^{\Psi_2+1} * h_2 \quad \ldots \quad z_0^{\Psi_p+1} * h_p\right)$, tout cela étant fourni par la procédure "**Characteristic_numbers()**". Cette dernière procédure calcule (comme son nom l'indique) les nombres caractéristiques d'un système dynamique (voir définition 5.1.1.). Elle utilise la condition d'arrêt liée à la proposition 5.1.1.

<u>*Exemple*</u> : Soit S le système dynamique suivant :

$$(S)\quad\begin{cases}\dot q_1(t) = q_2(t) + 2q_2(t)q_3(t) + 2q_2(t)a(t) + 2q_2(t)b(t)\\ \dot q_2(t) = q_3(t) + a(t) + b(t)\\ \dot q_3(t) = q_4(t) - q_3^2(t) - q_5^2(t) + a(t)\\ \dot q_4(t) = q_5(t) + 2q_3(t)(q_4(t) - q_3^2(t) - q_5^2(t)) + 2q_3(t)a(t) + 2q_5(t)b(t)\\ \dot q_5(t) = b(t)\\ y_1(t) = h_1(q) = q_1(t) - q_2^2(t)\\ y_2(t) = h_2(q) = q_3(t).\end{cases}$$

Soit $Z = \{z_0, z_1, z_2\}$ l'alphabet de codage :

- z_0 code la partie autonome :

$$Y_{z_0} = (q_2+2q_2q_3)\frac{\partial}{\partial q_1} + q_3\frac{\partial}{\partial q_2} + (q_4 - q_3^2 - q_5^2)\frac{\partial}{\partial q_3} + (q_5 + 2q_3(q_4 - q_3^2 - q_5^2))\frac{\partial}{\partial q_4},$$

- z_1 code l'entrée $a(t)$:

$$Y_{z_1} = 2q_2\frac{\partial}{\partial q_1} + \frac{\partial}{\partial q_2} + \frac{\partial}{\partial q_3} + 2q_3\frac{\partial}{\partial q_4}$$

- z_2 code l'entrée $b(t)$:

$$Y_{z_2} = 2q_2\frac{\partial}{\partial q_1} + \frac{\partial}{\partial q_2} + 2q_5\frac{\partial}{\partial q_4} + \frac{\partial}{\partial q_5}.$$

On souhaite de rendre ce système non interactif par bouclage par retour d'état (voir 5.8.3.). Et on souhaite aussi que le système non interactif obtenu (voir 5.7.4.) ait la matrice de commande étendue $\hat\Gamma$ sous la forme $\begin{pmatrix}0 & 0\\ k_1 & 0\\ 0 & k_2\end{pmatrix}$ (k_1 et k_2 sont des constantes arbitraires). Pour cela, nous allons calculer les lois de bouclage Φ correspondantes (voir définition 5.8.1.1. et théorème 5.8.3.1.) à l'aide de MACSYMA et des fonctions données au paragraphe 6.4.

```
(c18) p:2;

(d18)                                  2

(c19) m:2;

(d19)                                  2

(c20) N:5;

(d20)                                  5

(c21) depends(h,[q1,q2,q2,q3,q4,q5]);

(d21)                        [h(q1, q2, q2, q3, q4, q5)]

(c22) Y[z0](h):=(q2+2*q2*q3)*diff(h,q1)+q3*diff(h,q2)+(q4-q3^2-q5^2)*diff(h,q3)
         +(q5+2*q3*(q4-q3^2-q5^2)*diff(h,q4))$

(c23) Y[z0];

                      dh               2             2     dh          2             2
(d23) lambda([h], 2 --- q3 (- q5  + q4 - q3 ) + --- (- q5  + q4 - q3 ) + q5
                     dq4                                dq3

                                                      dh                          dh
                                                  + --- (2 q2 q3 + q2) + 2 --- q3)
                                                     dq1                         dq2
```

```
(c24) Y[z1](h):=2*q2*diff(h,q1)+diff(h,q2)+diff(h,q3)+2*q3*diff(h,q4)$

(c25) Y[z1];

                           dh          dh          dh        dh
(d25)        lambda([h], 2 --- q3 + 2 --- q2 + --- + 2 ---)
                           dq4         dq1         dq3       dq2

(c26) Y[z2](h):=2*q2*diff(h,q1)+diff(h,q2)+2*q5*diff(h,q4)+diff(h,q5)$

(c27) Y[z2];

                           dh          dh          dh        dh
(d27)        lambda([h], 2 --- q5 + 2 --- q2 + --- + 2 ---)
                           dq4         dq1         dq5       dq2

(c28) h[1]:q1-q2^2;

                                         2
(d28)                              q1 - q2

(c29) h[2]:q3;

(d29)                                 q3
```

```
(c30) Characteristic_numbers()$

(c31) sgn[1];

(d31)                               [z1, z2]

(c32) psi[1];

(d32)                                  0

(c33) sgn[2];

(d33)                                [z1]

(c34) psi[2];

(d34)                                  0

(c35) Control_matrix();

/usr1/macsyma_309/das/mstuff.o being loaded.

                               [ - 2 q2  1 ]
(d35)                          [           ]
                               [ - 2 q2  0 ]
```

```
(c36) Omega_chapeau:entermatrix(2,2);

Is the matrix  1. Diagonal  2. Symmetric  3. Antisymmetric  4. General
Answer 1, 2, 3 or 4
1;

Row 1 Column 1:  k1;

Row 2 Column 2:  k2;

Matrix entered.

                                  [ k1   0  ]
(d36)                             [         ]
                                  [ 0    k2 ]

(c37) Alpha:-submatrix(2,3,Gamma) . Omega^^-1;

      [   2               2      2    q5 - 2 q2 q3 + q2                2 ]
(d37) [ q5  - q5 - q4 + q3   - q5  + ----------------- + q5 + q4 - q3  ]
      [                                     2 q2                        ]

(c38) Beta:Omega_chapeau . Omega^^-1;

                                [          k1  ]
                                [ 0    - ----  ]
(d38)                           [        2 q2  ]
                                [              ]
                                [ k2    - k2   ]
```

```
(c39) Phi:addcol(ematrix(3,1,1,1,1),addrow(Alpha,Beta));

       [        2               2       2    q5  - 2 q2 q3 + q2             2 ]
       [ 1  q5   - q5 - q4 + q3   - q5  + ----------------- + q5 + q4 - q3   ]
       [                                        2 q2                          ]
       [                                                                      ]
(d39)  [                                              k1                      ]
       [ 0           0                           - ----                       ]
       [                                             2 q2                     ]
       [                                                                      ]
       [ 0           k2                            - k2                       ]

(c41)
```

6.4. Première liste de fonctions

```
Root(P):=inpart(P,0)$


Left(P):=inpart(P,1)$


Right(P):=if inpart(P,0)="." or inpart(P,0)="+" or inpart(P,0)="*"
          then inpart(P,allbut(1))
          else inpart(P,2)$


Action(P,f):=if atom(P) then ev(A[P](f),diff)
             else Root(P)="." then Action(Left(P),Action(Right(P),f))
                  else if Root(P)="+" then Action(Left(P),f)+Action(Right(P),f)
                       else if Root(P)="*" then Action(Right(P),f)*Left(P)
                            else if Root(P)="^^" then
                                   Action(Left(P),Action(Left(P)^^(Right(P)-1),f))
                                 else print("Error in polynomial :",P)$


Characteristic_numbers():=block([z,i,j,s,hs,nu],
Gamma:[],
for s:1 thru p do
   (hs:h[s],sgn[s]:[],nu:-1,
    while (nu<N-2) and (sgn[s]=[]) do
          (nu:nu+1,i:0,G[s]:hs,
           while (i<m) and (sgn[s]=[]) do
                 (i:i+1,z:concat('z,i),
                  if Action(z,hs)#0 then (sgn[s]:[z],Psi[s]:nu)),
           hs:Action(z0,hs)),
    if sgn[s]=[] then Psi[s]:INF
    else for j:i+1 thru m do
         (z:concat('z,j),if Action(z,G[s])#0 then sgn[s]:endcons(z,sgn[s])),
    Gamma:endcons(hs,Gamma)))$


Control_matrix():=block([z],
Gamma:matrix(Gamma),
for i:1 thru m do
   (z:concat('z,i),
    Gamma:addrow(Gamma,makelist(if member(z,sgn[s]) then Action(z,G[s])
                                else 0,s,1,p))),
Omega:submatrix(1,Gamma))$
```

6.5. Utilisation des λ–notations en MACSYMA

La λ–notation est souvent présentée comme un moyen adéquat permettant de définir et de programmer les fonctionnelles, c'est-à-dire des fonctions dont les arguments sont des fonctions. Elle consiste à représenter une fonction $t \longmapsto f(t)$ par la "Curryfication" de cette fonction, notée :

$$\lambda[t].f(t).$$

La λ–notation indique la correspondance entre la liste des paramètres de la fonction (ici la liste en question est $[t]$) et le corps de la fonction (ici le corps en question est $f(t)$). Ainsi à partir des fonctions $\lambda[t].\sin(t)$ et $\lambda[t].\cos(t)$, on peut fabriquer, par exemple, la fonction qui au réel t, fait correspondre le réel $\sin(t)+\cos(t)$ dont la λ–notation est :

$$\lambda[t].(\sin(t)+\cos(t)).$$

La λ–notation permet aussi de représenter les fonctionnelles. Elle permet donc de traiter naturellement les fonctions, les fonctionnelles et les fonctionnelles itérées comme des objets de programmation. Ce mécanisme sera utilisé récursivement dans l'implantation de la transformation d'Evaluation. Il est utile et élégant pour passer des fonctions en paramètre.

Les λ–expressions sont à la base de la programmation en LISP. En MACSYMA aussi on a accès à la description d'un objet fonctionnel par une λ–notation ([M3]). Par exemple on peut définir la fonction f en MACSYMA de la manière suivante :

$$f : lambda([t], \sin(t))\$$$

f est alors un élément de la liste "**values**" de MACSYMA. Par contre, en utilisant l'instruction suivante :

$$f(t) := \sin(t)\$$$

on définit une fonction f appartenant à la liste "**functions**" de MACSYMA (qui ne peut pas être employée seule comme expression, à l'intérieur d'une autre expression par exemple).

Toutefois la manipulation du calcul algébrique en termes de λ–notation n'est pas directement donnée en MACSYMA, et demande une programmation spécifique. Ainsi à partir des λ–notations $s : \lambda.[t]\sin(t)$ et $c : \lambda.[t]\cos(t)$, l'appel de $s+c$ renvoie :

$$lambda([t], \sin(t)) + lambda([t], \cos(t))$$

et non pas

$$lambda([t], \sin(t)+\cos(t)).$$

On peut construire des λ–expressions en utilisant des tableaux de fonctions. Par exemple, si on défini $f[5](t) := \sin(t)$, l'appel de $f[5]$ renvoie la λ–notation $\lambda[t].\sin(t)$. Mais cette façon de définir ne peut pas coder les fonctionnelles itérées. Ceci nous amène à écrire les fonctions :

. "**Lamb-exp(expression)**" prend en argument une expression explicite d'une fonction de t (c'est-à-dire **expression**), et rend comme résultat la $\lambda-$notation de la fonction qui à t associe **expression**.

. "**Somme(f1,f2)**" prend en argument deux $\lambda-$notations **f1** et **f2**, et rend comme résultat la $\lambda-$notation de la fonction qui à t associe la somme "**f1**(t)+**f2**(t)".

. "**Produit(r,f)**" prend en argument un réel **r** et une $\lambda-$notation **f**, et rend comme résultat la $\lambda-$notation de la fonction qui à t associe le produit "**r*f**(t)".

. "**Prodfonc(f1,f2)**" prend en argument deux $\lambda-$notations **f1** et **f2**, et rend comme résultat la $\lambda-$notation de la fonction qui à t associe le produit "**f1**(t)***f2**(t)".

. "**Composition(f1,f2)**" prend en argument deux $\lambda-$notations **f1** et **f2**, et rend comme résultat la $\lambda-$notation de la fonction qui à t associe la composition "**f1**o**f2**(t)".

. "**Stieltjes(f,z)**" prend en argument une $\lambda-$notation **f** et une lettre **z**, et rend comme résultat la $\lambda-$notation de la fonction qui à t associe l'intégrale de Stieltjes "$\int_0^t \mathbf{f}(\tau)d\xi_{\mathbf{z}}(\tau)$".

. "**Conv(f,z,n)**" prend en argument une $\lambda-$notation **f** et une lettre **z**, et rend comme résultat la $\lambda-$notation de la fonction qui à t associe l'Evaluation du mot $\mathbf{z}^{\mathbf{n}}$: "$\mathcal{E}_a(\mathbf{f};\mathbf{z}^{\mathbf{n}})(t)$". Son implantation est liée au théorème 2.4.5.

. "**Star-Jacob(f,S,n)**" prend en argument une $\lambda-$notation **f** et une série de la forme $(cz)^{*p}$ (p est un entier, c est un nombre complexe et z est une lettre), et rend comme résultat la $\lambda-$notation de la fonction qui à t associe l'Evaluation de la fraction rationnelle $\mathbf{S}^{*\mathbf{n}}$:"$\mathcal{E}_a(\mathbf{f};\mathbf{S}^{*\mathbf{n}})(t)$". Son implantation est liée aux propositions 2.5.1., 1.5.2. et au théorème 2.5.1.

. "**Jacob(f,S)**" prend en argument une $\lambda-$notation **f** et une série **S** de la forme

$$(c_0 z_{j_0})^{*p_0} z_{i_1}(c_1 z_{j_1})^{*p_1} z_{i_2}(c_2 z_{j_2})^{*p_2} \ldots z_{i_{k-1}}(c_{k-1} z_{j_{k-1}})^{*p_{k-1}} z_{i_k}(c_k z_{j_k})^{*p_k},$$

(les $p_0,\ldots,p_k$ sont des entiers, les $c_0,\ldots,c_k$ sont des nombres complexes, les $z_{j_0},\ldots,z_{j_k}$ et les $z_{i_0},\ldots,z_{i_k}$ sont des lettres), et rend comme résultat la $\lambda-$notation de la fonction qui à t associe l'Evaluation de la série **S** : "$\mathcal{E}_a(\mathbf{f};\mathbf{S})(t)$".

6.6. Présentation de l'algorithme d'Evaluation

Nous avons établi, au chapitre II, les propriétés fondamentales de la tranformation d'Evaluation (théorèmes 2.4.2., 2.4.3., 2.4.4., 2.4.5. et 2.5.1.). Nous allons appliquer ces propriétés pour calculer récursivement l'Evaluation des combinaisons linéaires de fractions rationnelles non commutatives de la forme $(c_0z_{j_0})^{*p_0}z_{i_1}(c_1z_{j_1})^{*p_1}z_{i_2}(c_2z_{j_2})^{*p_2}..z_{i_{k-1}}(c_{k-1}z_{j_{k-1}})^{*p_{k-1}}z_{i_k}(c_kz_{j_k})^{*p_k}$, où les $p_0,\ldots,p_k$ sont des entiers positifs, les $z_{j_0},\ldots,z_{j_k}$ et $z_{i_0},\ldots,z_{i_k}$ sont des lettres. L'algorithme est décrit comme suit :

si S est un réel **alors** il s'agit d'une multiplication par un scalaire
de l'Evaluation du mot vide, on applique
la définition 2.2.1. et le théorème 2.3.1.
sinon si S est une lettre **alors** c'est l'intégrale de Stieltjes de f
par rapport à la primitive indexée par S
sinon si Root(S)="*" **alors** il s'agit d'une multiplication par un scalaire,
on applique le théorème 2.4.1.
sinon si Root(S)="+" **alors** il s'agit d'une somme,
on applique le théorème 2.4.1.
sinon si Root(S)="." **alors** il s'agit d'une multiplication non
commutative, on applique le
théorème 2.4.2.
sinon si Root(S)="^^" **alors** il s'agit d'une puissance
non comutative,
si Left(S) est une lettre
alors ils'agit d'une répétition
d'une même lettre,
on applique le
théorème 2.4.5.
sinon ils'agit d'une répétition
d'une même série,
on applique le
théorème 2.4.2.
sinon si Root(S)="^*"
alors ils'agit d'une fraction
non commutative,
on applique
le théorème 2.5.1.
sinon on ne peut actuellement
traiter d'autre cas.

Plus généralement, on peut modifier cet algorithme pour les séries rationnelles de la forme $G_0z_{i_1}G_1z_{i_2}\ldots z_{i_{k-1}}G_{k-1}z_{i_k}G_k$, où les séries $G_0,\ldots,G_k$ sont échangeables et $z_{i_0},\ldots,z_{i_k}$ sont des lettres (voir 2.5.), sous la seule condition que l'Evaluation de chacune des séries échangeables G_i puisse être calculée par la transformation de Laplace-Borel multidimensionelle inverse. Il serait donc possible d'inclure (dans notre futur développement) dans l'algorithme d'Evaluation toute librairie de transformation de Laplace.

6.7. Deuxième liste de fonctions

```
Lamb_exp(f):=block(i:i+1,l[i](t):=factor(ratsimp(f)),l[i])$


Somme:lambda([f1,f2],Lamb_exp(f1(t)+f2(t)))$


Produit:lambda([r,f],Lamb_exp(r*f(t)))$


Prodfonc:lambda([f1,f2],Lamb_exp(f1(t)*f2(t)))$


Composition:lambda([r,f],Lamb_exp(f1(f2(t))))$


Stieltjes:lambda([f,z],Lamb_exp(integrate(f(tau)*a[z](tau),tau,0,t)))$


Conv:lambda([f,z,n],Lamb_exp(
if (f=un) then Xi[z](t)^n/n!
else integrate(f(tau)*((Xi[z](t)-Xi[z](tau))^(n-1)/(n-1)!)*a[z](tau),tau,0,t)))$


Star_Jacob:lambda([f,S,n],g:acob(un,S)(t),Lamb_exp(
if (f=un) then %e^(g(t))*sum(binomial(n-1,j)*(g(t))^j/j!,j,0,n-1)
else sum(binomial(n-1,j)
        *integrate(diff(f(tau),tau)*%e^(g(t)-g(tau))*(g(t)-g(tau))^j/j!,tau,0,t),
        j,0,n-1)))$


Jacob:lambda([f,S],
if numberp(S) then Produit(S,f)
else if atom(S) then Stieltjes(f,S)
    else if Root(S)="*" then Produit(Left(S),Jacob(f,Right(S)))
        else if Root(S)="+" then Somme(Jacob(f,Left(S),t),Jacob(f,Right(S)))
            else if Root(S)="." then Jacob(Jacob(f,Left(S)),Right(S))
                else if Root(S)="^^" then if atom(Left(S)) then
                                                Conv(f,Left(S),Right(S))
                                        else Jacob(Jacob(f,Left(S)),
                                                Left(S)^^(Right(S)-1))
                  else if Root(S)="^*" then Star_Jacob(f,Left(S),Right(S))
                      else print("Error in S :", S))$
```

6.8. Exemples d'exécution

```
(c40) un:lambda([t],1);

Time= 0 msec.

(d40)                              lambda([t], 1)

(c41) a[z0]:lambda([t],1);

Time= 0 msec.

(d41)                              lambda([t], 1)

(c42) a[z1]:lambda([t],t);

Time= 0 msec.

(d42)                              lambda([t], t)

(c43) a[z2]:lambda([t],%e^t);

Time= 0 msec.

                                                 t
(d43)                              lambda([t], %e )

(c44) a[z3]:lambda([t],sin(t));

Time= 0 msec.

(d44)                              lambda([t], sin(t))

(c45) a[z4]:lambda([t],1/(1+t));

Time= 0 msec.

                                                 1
(d45)                              lambda([t], -----)
                                               1 + t
```

```
(c46) Jacob(un,z0);

Time= 650 msec.

(d46)                          lambda([t], t)

(c47) Jacob(un,z1);

Time= 1433 msec.

                                            2
                                           t
(d47)                          lambda([t], --)
                                           2

(c48) Jacob(un,z2);

Time= 4500 msec.

                                          t
(d48)                       lambda([t], %e  - 1)

Time= 7016 msec.

(d49)                    lambda([t], - (cos(t) - 1))

(c50) Jacob(un,z4);

Time= 4983 msec.

(d50)                      lambda([t], log(t + 1))

(c51) Jacob(un,z3+z4);

Time= 11233 msec.

(d51)             lambda([t], log(t + 1) - cos(t) + 1)

(c52) Jacob(un,3*z3+4*z4);

Time= 13183 msec.

(d52)         lambda([t], 4 log(t + 1) - 3 cos(t) + 3)
```

```
(c53) Jacob(un,z0^^5);

Time= 83 msec.

                                                  5
                                                 t
(d53)                              lambda([t], ---)
                                                120

(c54) Jacob(un,z1^^5);

Time= 1900 msec.

                                                  10
                                                 t
(d54)                              lambda([t], ----)
                                                3840

(c55) Jacob(un,z2^^5);

Time= 366 msec.

                                                   t     5
                                                (%e  - 1)
(d55)                           lambda([t], ----------)
                                                  120

(c56) Jacob(un,z3^^5);

Time= 316 msec.

                                                          5
                                               (cos(t) - 1)
(d56)                        lambda([t], - -------------)
                                                  120

(c57) Jacob(un,z4^^5);

Time= 116 msec.

                                                5
                                             log (t + 1)
(d57)                          lambda([t], -----------)
                                                 120
```

```
(c58) Jacob(un,z0^*3);

Time= 366 msec.
                                          2           t
                                      (t   + 4 t + 2) %e
 (d58)                  lambda([t], -----------------)
                                              2

(c59) Jacob(un,z1^*3);

Time= 2216 msec.
                                                          2
                                                         t
                                                         --
                                    4       2             2
                                 (t   + 8 t   + 8) %e
 (d59)               lambda([t], -------------------)
                                             8

 (c60) Jacob(un,z2^*3);

 Time= 533 msec.
                                                       t
                              2 t        t          %e  - 1
                          (%e    + 2 %e   - 1) %e
 (d60)       lambda([t], -----------------------------)
                                      2

 (c61) Jacob(un,z3^*3);

 Time= 616 msec.
                    1 - cos(t)      2
                  %e             (cos (t) - 6 cos(t) + 7)
 (d61)lambda([t], -------------------------------------)
                                  2

 (c62) Jacob(un,z4^*3);

 Time= 400 msec.
                                 2
                (t + 1) (log (t + 1) + 4 log(t + 1) + 2)
(d62)lambda([t], ---------------------------------------)
                                 2
```

6.9. Exemple d'utilisation

La série suivante est tirée d'un exemple de l'article de M. Fliess, M. Lamnabhi et F. Lamnabhi-Lagarrigue ([FLL]) :

$$
\begin{aligned}
S = {} & (-k_1 z_0)^* z_1 - 2k_2(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_1(-k_1 z_0)^* z_1 \\
& + 4k_2^2(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_1(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_1(-k_1 z_0)^* z_1 \\
& + 12k_2^2(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_0(-3k_1 z_0)^* z_1(-2k_1 z_0)^* z_1(-k_1 z_0)^* z_1 \\
& - 8k_2^3(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_1(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_1(-k_1 z_0)^* z_0 \\
& \quad (-2k_1 z_0)^* z_1(-k_1 z_0)^* z_1 \\
& - 24k_2^3(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_0(-3k_1 z_0)^* z_1(-2k_1 z_0)^* z_1(-k_1 z_0)^* z_0 \\
& \quad (-2k_1 z_0)^* z_1(-k_1 z_0)^* z_1 \\
& - 72k_2^3(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_0(-3k_1 z_0)^* z_1(-2k_1 z_0)^* z_0(-3k_1 z_0)^* z_1 \\
& \quad ((-2k_1 z_0)^* z_1(-k_1 z_0)^* z_1 \\
& - 24k_2^3(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_1(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_0(-3k_1 z_0)^* z_1 \\
& \quad (-2k_1 z_0)^* z_1(-k_1 z_0)^* z_1 \\
& - 144k_2^3(-k_1 z_0)^* z_0(-2k_1 z_0)^* z_0(-3k_1 z_0)^* z_0(-4k_1 z_0)^* z_1(-3k_1 z_0)^* z_1 \\
& \quad (-2k_1 z_0)^* z_1(-k_1 z_0)^* z_1
\end{aligned}
$$

(il est possible qu'il y ait des erreurs dues à la transcription).

Pour $a^{z_1}(t) = t$, nous obtenons l'Evaluation de S :

$$
\begin{aligned}
& - e^{-4k_1 t}((6630k_2^3 - 360k_1^3k_2^2 + 30k_1^6k_2 - 6k_1^9)e^{4k_1 t} \\
& + (-k_1^7k_2^3t^7 - 11k_1^6k_2^3t^6 + (3k_1^8k_2^2 - 60k_1^5k_2^3)t^5 \\
& + (19k_1^7k_2^2 - 268k_1^4k_2^3)t^4 + (-1078k_1^3k_2^3 + 58k_1^6k_2^2 - 6k_1^9k_2)t^3 \\
& + (-3234k_1^2k_2^3 + 174k_1^5k_2^2 - 18k_1^8k_2)t^2 \\
& + (-5784k_1k_2^3 + 306k_1^4k_2^2 - 24k_1^7k_2 + 6k_1^{10})t - 5784k_2^3 \cdot \\
& + 306k_1^3k_2^2 - 24k_1^6k_2 + 6k_1^9)e^{3k_1 t} \\
& + (-12k_1^6k_2^3t^{-}680k_1^5k_2^3t^5 + (12k_1^7k_2^2 - 256k_1^4k_2^3)t^4 \\
& + (48k_1^6k_2^2 - 656k_1^3k_2^3)t^3 + (-1260k_1^2k_2^3 + 84k_1^5k_2^2 - 6k_1^8k_2)t^2 \\
& + (-1536k_1k_2^3 + 96k_1^4k_2^2 - 12k_1^7k_2)t - 768k_2^3 + 48k_1^3k_2^2 \\
& - 6k_1^6k_2)e^{2k_1 t} + (-18k_1^5k_2^3t^5 - 90k_1^4k_2^3t^4 \\
& + (6k_1^6k_2^2 - 198k_1^3k_2^3)t^3 + (18k_1^5k_2^2 - 270k_1^2k_2^3)t^2 \\
& + (18k_1^4k_2^2 - 216k_1k_2^3)t - 72k_2^3 + 6k_1^3k_2^2)e^{k_1 t} - 6k_1^4k_2^3t^4 \\
& - 24k_1^3k_2^3t^3 - 36k_1^2k_2^3t^2 - 24k_1k_2^3t - 6k_2^3)/(6k_1^{11})).
\end{aligned}
$$

Pour $a^{z_1}(t) = e^t$, nous obtenons l'Evaluation de S :

$$
\begin{aligned}
&- e^{-4k_1t}(((15k_1^6 + 12k_1^5 - 69k_1^4 - 66k_1^3)k_2^3 \\
&+ (-6k_1^8 + 24k_1^7 + 30k_1^6 - 120k_1^5 - 24k_1^4 + 96k_1^3)k_2^2 \\
&+ (3k_1^{10} - 24k_1^9 + 42k_1^8 + 84k_1^7 - 273k_1^6 + 84k_1^5 + 228k_1^4 - 144k_1^3)k_2 \\
&- 3k_1^{12} + 33k_1^{11} - 120k_1^{10} + 90k_1^9 + 441k_1^8 - 1071k_1^7 + 570k_1^6 \\
&+ 660k_1^5 - 888k_1^4 + 288k_1^3)e^{4k_1t} \\
&+ ((-4k_1^9 + 26k_1^8 - 26k_1^7 - 148k_1^6 + 304k_1^5 + 386k_1^4 - 1834k_1^3 \\
&+ 2328k_1^2 - 1320k_1 + 288)k_2^3e^{4t} + ((12k_1^9 - 90k_1^{+}8138k_1^7 + 366k_1^6 \\
&- 1158k_1^5 + 420k_1^4 + 1512k_1^3 - 1776k_1^2 + 576k_1)k_2^3 \\
&+ (6k_1^{10} - 51k_1^9 + 114k_1^8 + 114k_1^7 - 762k_1^6 + 789k_1^5 + 546k_1^4 \\
&- 1644k_1^3 + 1176k_1^2 - 288k_1)k_2^2)e^{3t} \\
&+ ((-12k_1^9+102k_1^8-204k_1^7-324k_1^6+1284k_1^5-498k_1^4-1068k_1^3+720k_1^2)k_2^3 \\
&+ (-12k_1^{10} + 108k_1^9 - 264k_1^8 - 168k_1^7 + 1428k_1^6 \\
&- 1428k_1^5 - 576k_1^4 + 1488k_1^3 - 576k_1^2)k_2^2 \\
&+ (-6k_1^{11} + 60k_1^{10} - 186k_1^9 + 48k_1^8 + 798k_1^7 - 1428k_1^6 + 426k_1^5 \\
&+ 1032k_1^4 - 1032k_1^3 + 288k_1^2)k_2)e^{2t} \\
&+ ((4k_1^9 - 38k_1^8 + 92k_1^7 + 118k_1^6 - 616k_1^5 + 136k_1^4 + 736k_1^3)k_2^3 \\
&+ (6k_1^{10} - 57k_1^9 + 147k_1^8 + 105k_1^7 - 861k_1^6 + 672k_1^5 + 708k_1^4 - 720k_1^3)k_2^2 \\
&+ (6k_1^{11} - 60k_1^1 + 180k_1^9 - 882k_1^7 + 1260k_1^6 + 120k_1^5 \\
&- 1200k_1^4 + 576k_1^3)k_2 + 3k_1^{12} - 33k_1^{11} + 120k_1^{10} - 90k_1^9 - 441k_1^8 \\
&+ 1071k_1^7 - 570k_1^6 - 660k_1^5 + 888k_1^4 - 288k_1^3)e^t)e^{3k_1t} \\
&+ ((-24k_1^8 + 186k_1^7 - 318k_1^6 - 630k_1^5 + 2226k_1^4 - 1140k_1^3 - 2100k_1^2 \\
&+ 2664k_1 - 864)k_2^3e^{4t} + ((48k_1^8 - 408k_1^{+}7888k_1^6 + 792k_1^5 - 4488k_1^4 \\
&+ 3648k_1^3 + 672k_1^2 - 1152k_1)k_2^3 + (12k_1^9 - 108k_1^8 + 264k_1^7 + 168k_1^6 \\
&- 1428k_1^5 + 1428k_1^4 + 576k_1^3 - 1488k_1^2 + 576k_1)k_2^2)e^{3t} \\
&+ ((-24k_1^8 + 222k_1^7 - 582k_1^6 - 102k_1^5 + 2082k_1^4 - 1524k_1^3 - 720k_1^2)k_2^3 \\
&+ (-12k_1^9 + 114k_1^8 - 312k_1^7 - 66k_1^6 + 1452k_1^5 - 1704k_1^4 - 48k_1^3 + 576k_1^2)k_2^2 \\
&+ (-3k_1^{10}+30k_1^9-90k_1^8+441k_1^6-630k_1^5-60k_1^5+600k_1^3-288k_1^2)k_2)e^{2t})e^{2k_1t} \\
&+ ((-18k_1^7+144k_1^6-252k_1^5-504k_1^4+1638k_1^3-504k_1^2-1368k_1+864)k_2^3e^{4t} \\
&+ ((18k_1^7 - 156k_1^6 + 342k_1^5 + 348k_1^4 - 1800k_1^3 + 1104k_1^2 + 576k_1)k_2^3 \\
&+ (3k_1^8 - 27k_1^7 + 63k_1^6 + 63k_1^5 - 378k_1^4 + 252k_1^3 + 312k_1^2 - 288k_1)k_2^2)e^{3t})e^{k_1t} \\
&+ (-3k_1^6 + 24k_1^5 - 39k_1^4 - 102k_1^3 + 276k_1^2 + 24k_1 - 288)k_2^3e^{4t}) \\
&/(3k_1^{13} - 36k_1^{12} + 153k_1^{11} - 210k_1^{10} - 351k_1^9 + 1512k_1^8 - 1641k_1^7 \\
&- 90k_1^6 + 1548k_1^5 - 1176k_1^4 + 288k_1^3)).
\end{aligned}
$$

Pour $a^{z_1} = un$, nous obtenons l'Evaluation de S :

$$\begin{aligned}
&- \mathrm{e}^{-4k_1t}((15k_2^3 - 6k_1^2k_2^2 + 3k_1^4k_2 - 3k_1^6)\mathrm{e}^{4k_1t} \\
&+ (-4k_1^3k_2^3t^3 + (6k_1^4k_2^2 - 12k_1^2k_2^3)t^2 \\
&+ (-6k_1k_2^3 + 6k_1^3k_2^2 - 6k_1^5k_2)t + 12k_2^3 - 3k_1^2k_2^2 + 3k_1^6)\mathrm{e}^{3k_1t} \\
&+ (-24k_1^2k_2^3t^2 + (12k_1^3k_2^2 - 36k_1k_2^3)t - 12k_2 3^+ 6k_1^2k_2^2 - 3k_1^4k_2)\mathrm{e}^{2k_1t} \\
&+ (-18k_1k_2^3t - 12k_2^3 + 3k_1^2k_2^2)\mathrm{e}^{k_1t} - 3k_2^3)/(3k_1^7)).
\end{aligned}$$

Le tableau suivant donne une idée sur la complexité de l'algorithme :

	un	t	e^t
$\mathcal{E}_a(S)$	1762 s	2800 s	10477 s

Note : *Pour cette série S, l'exploration de l'arbre et les appels récursifs nécessitent seulement* 12 s.

BIBLIOGRAPHIE

[A1] **S.S. Abhyankar**, *Local Analytic Geometry*, Academic Press, New York & London, 1964.

[A2] **A.G. Akritas**, *Elements of Computer Algebra with Applications*, John Wiley & Sons, 1989.

[B1] **S. Barnett**, *Polynomials and Linear Control Systems*, Marcel Dekker, Inc., New York and Basel, 1983.

[B2] **J.F. Barrett**, *The use of functional in the analysis on non linear physical systems*, J. Electronics Control, vol 15, pp. 567-615, 1963.

[BM] **G. Basile & G. Marro**, *A state space approach to non-interacting controls*, Ricerche di Automatica, 1, pp. 68-77, 1970.

[BR1] **E. Bedrosian & S.Rice**, *The output properties of Volterra systems driven by harmonic and Gaussian inputs*, Proc. IEEE, vol. 59, pp. 1688-1707,1971.

[BR2] **J. Berstel & C. Reutenauer**, *Rational Series and their languages*, Springer-Verlag, 1988.

[B3] **E. Bernard-Weil**, *Formalisation et contrôle du système endocrinien surréno-posthypophysaire par le modèle mathématique de la régularisation des couples ago-antagonistes*, Thèse de Doctorat ès sciences, Université de Paris VI, 1979.

[B4] **J.A. Borrie**, *Modern Control Systems, A Manual of Design Methods*, Prentice Hall, International, 1986.

[B5] **N. Bourbaki**, *Groupes et Algèbres de Lie*, Chapitre 2, Hermann, 1972.

[B6] **R.W. Brockett**, *Volterra Series and Geometric Control Theory*, Automatica, 12, pp. 167-176, 1976.

[BCL] **B. Buchberger G.E. Colin & R. Loos (éditeurs)**, *Symbolic & Algebraic Computation. Computing Supplementum 4*, Springer-Verlag, 1982.

[BDK] **C. Bruni, G. Di Pillo et G. Koch**, *On the mathematical model of bilinear systems*, Ricerche di Automatica.

[C1] **K.T. Chen**, *Iterated path integrals*, Bull. Amer. Math. Soc., vol 83, 1977, pp. 831-879.

[C2] **K.T. Chen**, *Iterated integrals and exponential homomorphisms*, Proc. London Math Soc. (3) 4, 1954.

[C3] **R. Cori**, *Un code pour les graphes planaires et ses applications*, Astérique, 27, 1975.

[CS] **N. Chomsky & M.P. Schützenberger**, *The algebraic theory of context-free langage in computer programing and formal systems*, North Hollang Publishing Company, Amsterdam, 1963..

[CN1] **L.O. Chua & C.Y. Ng**, *Frequency domain analysis of nonlinear systems : general theory*, IEEE Elec. Cir. & Syst., vol. 3, N 4, July 1979.

[CN2] **L.O. Chua & C.Y. Ng**, *Nonlinear Oscillation Via Volterra Series*, IEEE Elec. Cir. & Syst., vol. 3, March 1982.

[C3] **D. Claude**, *Découplage et Linéarisation des systèmes non linéaires par bouclages statiques*, Thèse d'état, Université de Paris Sud, Centre d'Orsay, 1986.

[CFI] **D. Claude, M. Fliess & A. Isidori**, *Immersion, directe et par bouclage, d'un système non linéaire dans un système linéaire*, C.R. Acad. Sci. Paris, 296, série I, 1983, pp. 237-240.

[DST] **Davenport, Y. Siret & E. Tournier**, *Calcul Formel. Systèmes et algorithmes de manipulations algébriques*, Masson, Collection E.R.I, 1987.

[D1] **M.P. Delest**, *Utilisation de MACSYMA dans l'Enumération des polyominos*, Premières Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles Non Commutatives", Lille, December 1988.

[D2] **S. Diop**, *Théorie de l'élimination et principe du modèle interne en Automatique*, Thèse, Université Paris Sud, Centre Orsay, 1989.

[E] **S. Eilenberg**, *Automata, Langages and Machines*, Academic press, 1976.

[F1] **R.J. Fateman**, *A review of MACSYMA*, IEEE Transaction on knowledge and data engineering, Vol 1, 1, March 1989.

[F2] **Ph. Flajolet**, *Mathematical Methods in the Analysis of Algorithms and Data Structures*, Lecture Notes for A Graduate Course on Computation Theory, Udine (Italy) September-October 1984.

[FSZ] **Ph. Flajolet, B. Salvy & P. Zimmermann** , *LAMBDA-UPSILON-OMEGA : an assistant algorithm analyser*, Premières Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles Non Commutatives", Lille, December 1988.

[F3] **M. Fliess**, *Sur certaines familles de séries formelles*, Thèse d'état, Université Paris VII, 1972.

[F4] **M. Fliess**, *Sur divers produit des séries formelles*, Bull. Soc. Math France, 102, 1974, pp. 181-191.

[F5] **M. Fliess**, *Fonctionnelles causales non linéaires et indéterminées non commutative*, Bull. Soc. Math France, 109, 1981, pp. 3-40.

[F6] **M. Fliess**, *Réalisation locale des systèmes nonlinéaires, algèbres de Lie filtrées transitives et séries génératrices non commutatives*, Invent. Math., 71, pp. 521-537. 1983.

[F7] **M. Fliess**, *Automatique et corps différentiels*, Forum Math., 1, 1989.

[F8] **D. Foata**, *La série génératrice exponentielle dans les problèmes d'énumération*, Séminaire de Math. Sup., Presses de L'Université de Montréal, Montréal, 1974.

[F9] **E. Freund**, *Decoupling and pole assignement in nonlinear systems*, Electronics Letters, Vol 9, N 16, pp. 373-374, 08 1973.

[F10] **E. Freund**, *The Structure of Decoupled nonlinear systems*, Inter J. Control, Vol 11, N 3, pp. 443-450, 03 1975.

[FLL] **M. Fliess, M. Lamnabhi & F. Lamnabhi-Lagarrigue**, *An algebraic approach to nonlinear functional expansions*, IEEE Trans. Circ. Syst., CAS-30, 1983, pp. 554-570.

[FW] **P.L. Falb & W.A. Wolovich**, *Decoupling in the Design and Synthesis of Multivariable Control Systems*, IEEE Trans. Aut. Cont., Vol. AC-12, N 6, 12, pp. 651-659, 1967.

[G1] **J.P. Gauthier**, *Structure des Systèmes Non Linéaires*, Edition du C.N.R.S., Paris, 1984.

[G2] **R.P. Gilbert**, *Functinal Theiric Methods in Partial Differential Equation*, MIT RLE technical repport, 335, 1959.

[G3] **D. George**, *Continuous nonlinear systems*, MIT RLE technical repport, 335, 1959.

[G4] **E.G. Gilbert**, *Functinal expansion for reponse of nonlinear differential systems*, IEEE Trans. Aut. Cont., Vol. AC-22, pp. 909-921, 1977.

[GH] **E.G. Gilbert & I.J. Ha**, *A complete Charaterization of Decoupling Control Laws for General Class of Nonlinear Systems*, IEEE Trans. Aut. Cont., Vol. AC-31, N 9, 12, pp. 823-830, 1986.

[GIKM] **C. Gori-Giorgi, A. Isidori, A.J. Krener & S. Monaco**, *Nonlinear decoupling via feedback : a differential geometric approach*, IEEE Trans. Aut. Cont., Vol. AC-26, pp.331-345, 1981.

[GM] **A. Glumineau & R. Mezencev**, *Hybrid simulation of a tanker moored at a single point subjected to the effects of wind, current and waves*, 10^{th} IMACS World Congress on system simulation and scientific computation, Montréal, Canada, 1982, pp 98-100.

[H1] **H. Hermann**, *On the accessibility problem in control theory*, in International Symposium on Nonlinear Differential Equations and Nonlinear Mechanic, Academic press, New York, pp. 325-332, 1963.

[H2] **V. Hoang Ngoc Minh**, *Eléments d'un calcul symbolique pour les systèmes dynamiques non linéaires*, Premières Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles non commutatives", Lille, December 1988.

[HG] **I.J. Ha & E.G. Gilbert**, *A Complete Characterization of Decoupling Control laws for a General Class of Nonlinear Systems*, IEEE Trans., Vol AC-31, N 9, 09-1986..

[HJ1] **C. Hespel & G. Jacob**, *Approximation of Non Linear Systems by Bilinear ones*, in "Algebraic and Geometric Methods in Nonlinear Control Theory", M. Fliess & M. Hazewinkel (eds), pp. 551-519.

[HJ2] **C. Hespel & G. Jacob**, *Calcul des approximations locales bilinéaires des systèmes analytiques*, RAIRO APII, vol. 23, N 4, pp. 331-350, 1989.

[HJ3] **C. Hespel & G. Jacob**, *Approximation des systèmes dynamiques non linéaire par des séries rationnelles*, Premières Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles Non Commutatives", Lille, December 1988.

[HJ4] **V. Hoang Ngoc Minh & G. Jacob**, *Symbolic calculus and Volterra series*, IFAC Symposium "Nonlinear Control Systems Design", Capri, Juin 1989.

[HJ5] **V. Hoang Ngoc Minh & G. Jacob**, *Decoupling for Nonlinear Control Systems*, LIFL technical repport, IT 162, 1989.

[HJ6] **V. Hoang Ngoc Minh & G. Jacob**, *Transformation Evaluation and Symbolic Calculus for Nonlinear Control Systems*, "Neuvième Conférence Internationale Analyse et Optimisation des Systèmes", Antibes, Juin 1990.

[HJ7] **V. Hoang Ngoc Minh & G. Jacob**, *Transformation Evaluation and its implementation in MACSYMA*, "New Trends in Systems Theory", Genoa, Juillet, 1990.

[HJ8] **V. Hoang Ngoc Minh & G. Jacob**, *Transformation Evaluation with Kernel Function*, à paraître dans Calsyf.

[HJ8] **V. Hoang Ngoc Minh, G. Jacob & N.E. Oussous**, *Comportement entree/sortie des systemes analytiques non lineaires : approximations rationnelles, approximations structurelles nilpotentes*, "Analysis of Contolled Dynamical Systems", Lyon, July 1990.

[HMS1] **T. Huillet, A. Monin & G. Salut**, *Représentations exponentielles en commande non linéaire*, RAIRO APII (Automatique), 21, pp. 419-448, 1987.

[HMS2] **T. Huillet, A. Monin & G. Salut**, *Lie Algebraic Canonical Representations*, Mathematical Systems Theory, 20, pp. 193-213, 1987.

[HMS3] **T. Huillet, A. Monin & G. Salut**, *Représentations exponentielles des systèmes dynamiques généraux*, RAIRO APII (Automatique), 24, pp. 37-56, 1990.

[I] **A. Isidori**, *Nonlinear Control Systems : an Introduction*, Springer-Verlag, 1989.

[J1] **G. Jacob**, *Représentations and substitutions matricielles dans la théorie algébrique des transductions*, Thèse d'état, Université Paris VII, 1975.

[J2] **G. Jacob**, *Séries non commutatives, calcul formel et systèmes dynamiques non linéaires*, collogue "Calcul Formel en Théorie des systèmes" Gif sur Yvette, 1989.

[J3] **B. Jakcubzyk**, *Existence and uniqueness of realization of non linear systems*, SIAM, J. Cont. & Opt., Vol. 18, 1980, pp. 455-471.

[JO1] **G. Jacob & N.E. Oussous**, *Sur un résultat de Ree : Séries de Lie et Algèbres de mélange*, LIFL technical repport, IT 103, 1987.

[JO2] **G. Jacob & N.E. Oussous**, *Local and minimal realization of nonlinear dynamical systems*, IFAC Symposium "Nonlinear Control Systems Design", Capri, Juin 1989.

[K1] **T. Kailath**, *Linear systems*, Prentice Hall, 1980.

[K2] **S.C. Kleene**, *Representation of Events in Nerve Nets and Finite Automata*, Automata studies, Priceton, N.J. 1956, pp. 3-40.

[K3] **D. Knuth**, *The Art of Computing Programing - Semi-numerical Algorithms*, Addison-Wesley, 1969.

[K4] **D. Knuth**, *The Art of Computing Programing - Fundamental Algorithms*, Addison-Wesley, 1973.

[K5] **P.V. Koseleff**, *Jeux de mots dans les algèbres de Lie libres : quelques bases et formulles*, Premières Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles Non Commutatives", Lille, December 1988.

[K6] **A.J. Krener**, *(Adf, g), (adf, g) and Locally (adf, g)-Invariant and Controllability Distributions*, SIAM, J. Cont. and Opt., Vol. 23, 1985, pp. 523-549.

[K7] **W. Kuich**, *An Algebraic Characterization of Some Principal Regulated Rational Cones*, J. Comp. Syst. Sc., vol 25, 1982, pp. 377-401.

[KS] **W. Kuich & A. Salomaa**, *Semiring, Automata, Languages*, Springer-Verlag, 1986.

[L1] **F. Lamnabhi-Lagarrigue**, *Séries de Volterra et Commande optimale singulière*, Thèse d'état, Université de Paris Sud, Centre d'Orsay, 1985.

[L2] **M. Lamnabhi**, *Analyse des systèmes non linéaires par les méthodes de développement fonctionnelles*, Thèse d'état, Université de Paris Sud, Centre d'Orsay, 1986.

[LK] **C. Lesiak & A.J. Krener**, *The existence and Uniqueness of Volterra Series for Nonlinear Systems*, IEEE Trans. Aut. Cont., AC-23, N 6, December, pp. 1090-1095, 1978.

[L3] **J.D. Lipson**, *Elements of Algebra and Algebraic Computing*, The Benjamin Comming Buplisshing Compagny, Inc., 1981.

[L4] **C. Lobry**, *Contrôlabilité des systèmes non linéaires*, SIAM, J. Cont. and Opt., Vol. 8, N 4, pp. 573-605, 1970.

[L5] **D. G. Luenberger**, *Introduction to Dynamic Systems*, John Wiley & Sons, 1979.

[M1] **M.P. Malliavin**, *Algèbre commutative. Applications en Géométrie et Théorie des Nombres*, Masson, collection Maitrise de Mathématiques Pures, 1985.

[MR1] **J. Marchand & R. Rioboo**, *A propo de l'utilisation de Common-LISP en calcul formel*, Puplication du LITP Paris, 1989.

[MR2] **G. Melançon & C. Reutenauer**, *Lyndon words, free algebras and shuffles*, Puplication du LITP Paris et UQAM Montréal, Québec, 87-63, 1987.

[MW] **A.S. Morse & W.M. Wonham**, *Decoupling and pole assignement in linear multivariable system: a geometric approach*, SIAM J. Control, pp. 1-18, 8 1970.

[M2] **C. Moog**, *Inversion, Découplage, Poursuite de modèle des Systèmes Non Linéaires*, Thèse, Université Nantes, 1987.

[M3] **MACSYMA**, *Reference Manual Version 11*, Symbolics.

[N1] **H. Nejmeijer**, *The triangular decoupling for nonlinear control systems*, Nonlinear Anal. Appl., 8, pp. 273-279, 1984.

[NS] **H. Nejmeijer & J.M. Schumacher**, *The regular local non interacting control problem for nonlinear control systems*, SIAM J. Control Opt., pp. 1232-1245, 1987.

[N2] **M. Nivat**, *Transductions des langages de Chomsky*, Ann. Ins. Fourier, vol. 18, 1968, pp. 448-483.

[P1] **J.F. Pommaret**, *Diefferential Galois Theory*, Gordon and Beach, New York, 1983.

[P2] **J.F. Pommaret**, *Lie pseudogroups and mechanics*, Gordon and Beach, New York, 1988.

[P3] **W.A. Porter**, *Diagonalization and inverses for nonlinear systems*, Int. J. Cont., vol 10, pp. 252-264, 1970.

[PW] **R. Pavelle & P.S. Wang**, *MACSYMA from F to G*, J. Symbolic Computation 1, 69-100, 1985.

[R1] **R. Ree**, *Lie elements and an algebra associated with shuffle*, A. on Math, V. 68, pp 210-220, 1958.

[R2] **C. Reutenauer**, *Séries rationnelles et Algèbres syntaxiques*, Thèse d'état, Université Paris VI, 1980.

[R3] **C. Reutenauer**, *The local realization of generating series of finite Lie rang*, in "Algebraic and Geometric Methods in Nonlinear Control Theory", M. Fliess & M. Hazewinkel (eds), pp. 33-43.

[R4] **J.F. Ritt**, *Differential Algebra*, Amer. Math. Soc., 1950.

[R5] **W.J. Rugh**, *Nonlinear system theory*, Baltimore : The Johns Hopkins Univ. Press, 1981.

[SS] **A. Salomaa & M. Soittola**, *Automata-Theoretic Aspects of formal power series*, Springer-Verlag, 1978.

[S1] **M. Schetzen**, *The Volterra and Wiener theories of non linear systems*, Wiley, New York, 1980.

[S2] **M.P. Schützenberger**, *Un problème de la théorie des automates*, Séminaire Dubreil-Pisot, 13^{th} year (1959-1960), n° 3, Inst. H. Poincaré, Paris, 1960.

[S3] **M.P. Schützenberger**, *On the Definition of Family of Automata*, Infor. Conr., Vol 4, 1961, pp. 245-270..

[S4] **M.P. Schützenberger**, *Finite Counting Automata*, Infor. Conr., Vol 5, 1962, pp. 91-107..

[S5] **H.J. Susmann**, *Lie brackets and local controlability : A sufficient condition for scalar-input systems*, SIAM, J. Cont. and Opt., Vol. 21, N 5, pp. 686-713, Sept. 1983.

[S6] **H.J. Susmann**, *A product expansion for Chen Series*, in "Theory and Applications of Nonlinear Control Systems", C.I. Byrns and Lindquist (eds). 323-335, 1986.

[S7] **SCRATCHPAD**, *The Scratchpad II Computer Algebra System, Interative Environnement Users Guide*, IBM Research Division, Yorktown Heights, NY 10598, USA, Septembre 1988.

[V1] **G. Viennot**, *Algèbres de Lie libres et Monoïdes libres*, Lecture Notes in Mathematics, 691, 1978.

[VL] **G. Viennot & P. Leroux**, *A combinatorial approach to nonlinear functionnal expensions : an introduction with example*, Premières Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles Non Commutatives", Lille, December 1988.

[V2] **V. Volterra**, *Leçon sur les fonctions de lignes*, Gauthier-Villars, Paris, 1913.

[VP] **V. Volterra & J. Pérès**, *Théorie générale des fonctionnelles*, Gauthier-Villars, Paris, 1936.

[W] **W.M. Wonham**, *Linear Multivariable Control : A geometric approach*, Springer-Verlag, 1979.

ANNEXE A

EVALUATION TRANSFORM

(A paraître dans
"Algebraic and Computing Treatment of Noncommutative power Series",
ed. G. Jacob & C. Reutenauer, Theoret. Comput. Sci.)

EVALUATION TRANSFORM

V. Hoang Ngoc Minh
L.I.F.L. - U.A. 369 C.N.R.S.
Université Lille I
59650 Villeneuve d'Ascq Cedex

Abstract

Given a nonlinear control system, one can view its output function as a signal, parametrized by the primitives of the input functions. This signal can be formally described by its M. Fliess' power series, that is a formal power series on noncommuting variables. The temporal behaviour of the system can be derived from this symbolic description by a transform, that we call "Evaluation transform" and that generalizes inverse Laplace transform to the nonlinear area. We develop here the basic tools of that symbolic calculus. We prove a correspondence theorem between certain convolutions of signals and Cauchy products of generating power series.

1. Introduction

Let $Z = \{z_0, z_1, \ldots, z_m\}$ be a finite alphabet. Let w be a word of Z^* :
. if $w = \epsilon$, then A_ϵ is the identity,
. if $w = w_1 z$, then A_w is the differential operator $A_{w_1} A_z$.

Let (S) be a nonlinear control system described in the following form

$$(S) \quad \begin{cases} \dot{q}(t) = \sum_{z \in Z} a^z(t) A_z(q) \\ y(t) = h(q(t)) \end{cases}.$$

where :

- q is an element of the real analytic manifold Q of dimension N,
- $\forall z \in Z$, A_z is an analytic vector field over Q. We note A the vector ${}^t(A_{z_0} \quad A_{z_1} \quad \ldots \quad A_{z_m})$,
- $\forall z \in Z$, a^z is a continuous piecewise mapping from $\mathbb{R}_+$ to $\mathbb{R}$. In particular a^{z_0} is a constant mapping and equal to 1. We note a the vector $(a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$,
- the observation $h = {}^t(h_1 \quad h_2 \quad \ldots \quad h_p)$ is an analytic mapping from the real analytic manifold Q to $\mathbb{R}^p$.

We can associate to the observation h its generating series :

$$\sigma h = \sum_{w \in Z^*} < \sigma h | w > w,$$

that is a formal power series in noncommuting variables belonging to the finite alphabet Z. By the fundamental formula of M. Fliess (also called "Peano-Baker formula", [3]), the output $y(t)$ defined by the observation h is obtained by the replacement in σh of each word w by the associated "iterated integral" $\int_0^t \delta_a w$ relative to the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ defined over $[0,t]$, $t \geq 0$ (the input $a^{z_0}(t) \equiv 1$ encodes the autonomous part of the system). Here we will call "Evaluation" of the word w, this associated iterated integral, $\mathcal{E}_a(w) = \int_0^t \delta_a w$, and we will call "Evaluation" of the power series S (submitted to some convergence conditions) "the output" $\mathcal{E}_a(S)$, obtained by replacing each word w in S by its Evaluation $\mathcal{E}_a(w)$. Then the Evaluation of S can be viewed as a signal, depending on the time t, and on the m independent parameters $\xi_z(t) = \int_0^t a^z(\tau)d\tau$, $z \in Z$.

This point of view leads naturally to develop a noncommuting "symbolic calculus" in the nonlinear area, that generalizes the Heaviside calculus ([4]). So, the notions of "transfer function" (generating series on one variable) and "impulsive response", coding signals produced by linear or multilinear systems, can be generalized to generating series on $m+1$ variables and Volterra series, coding signals produced by nonlinear control systems. The Evaluation function $\mathcal{E}_a$ corresponds to the "inverse Laplace transform".

Our goal is to develop here the basic tools of this symbolic calculus on noncommuting variables. We prove a correspondence theorem between certain convolutions of signals and Cauchy products of generating power series. We give also some applications of this theorem. This Evaluation transform allows to obtain a simple calculation of the Taylor expansion of Volterra kernels ([6]). This systematic treatment has been used in [7] (via some kernel function) to give a concise implementation in the computer algebraic system MACSYMA, allowing a particularly quick computation.

Recall that $Z = \{z_0, z_1, \ldots, z_m\}$ is a finite alphabet. An element of Z is called letter. A word is a finite sequence w of letters $w = z_{j_1} z_{j_2} \ldots z_{j_k}$. The length of w, noted $|w|$, is its length as sequence of letters. The empty word ϵ is the empty sequence of letters ($|\epsilon| = 0$). We note Z^* the set of all words over Z. The concatenation product of $u = z_{j_1} z_{j_2} \ldots z_{j_k}$ and $v = z_{i_1} z_{i_2} \ldots z_{i_l}$ is the "juxtaposition" of u and v. Thus we have $uv = z_{j_1} z_{j_2} \ldots z_{j_k} z_{i_1} z_{i_2} \ldots z_{i_l}$. This product is associative, and admits ϵ as the identity element. It is easy to verify that Z^* is the free monoid generated by the aphabet Z. Any subset of Z^* is called language.

A formal power series on the associative variables $z \in Z$ (noncommuting if card $Z \geq 2$) with coefficients in A ([1]), is any mapping

$$\begin{aligned} S : Z^* &\longrightarrow A \\ w &\longmapsto \langle S|w \rangle, \end{aligned}$$

and the set of all formal power series over Z is denoted by $A \ll Z \gg$.

A formal power series S will be written as a formal sum :

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

where $< S|w >$ is the coefficient of the word w in S. A formal power series $S \in A \ll Z \gg$ will be said quasiregular if and only if its constant term vanishes. For any quasiregular formal power series on noncommutative variables S in $A \ll Z \gg$, S^* represents classically the formal power series $\sum_{n\geq 0} S^n$. In commutative variables, it coincides with the rational fraction $1/(1-S)$, and in this case we have $S^{*n} = \left(1/(1-S)\right)^n$.

Let S be a formal power series in $A \ll Z \gg$. The support of S is the language over Z defined by :

$$\mathrm{supp}(S) = \{w \in Z^* \mid < S|w > \neq 0\}.$$

A formal power series P with finite support is called **polynomial.**

The **sum** of two formal power series S, T in $A \ll Z \gg$ is the formal power series $S+T$ defined by :

$$\forall w \in Z^*, \quad < S+T|w > = < S|w > + < T|w > .$$

The **Cauchy product**, noted by ".", of two formal power series S, T in $A \ll Z \gg$ is the formal power series $S.T$ defined by :

$$\forall w \in Z^*, \quad < S.T|w > = \sum_{u,v \in Z^*, uv=w} < S|u >< T|v > .$$

The symbol "." will be omitted when there is no ambiguity.

The **shuffle product**, noted by "ш", of two formal power series S, T in $A \ll Z \gg$ is the formal power series $S ш T$ defined by :

$$S ш T = \sum_{u,v \in Z^*} < S|u >< T|v > \, u ш v,$$

where $u ш v$ is the polynomial defined as follows :

$$\textit{for any word } u: \quad u ш \epsilon = \epsilon ш u = u$$

$$\textit{for any words } u, v, \textit{for any letters } x, y : ux ш vy = [(ux) ш v]y + [u ш (vy)]x.$$

The coefficients of the polynomial $u ш v$ are positive integers.

We note $\mathrm{som}(P)$ the sum of the coefficients of the polynomial P :

$$\mathrm{som}(P) = \sum_{w \in \mathrm{supp}(P)} < P|w > .$$

Let u, v be two words in Z^*. The sum of the coefficients of the polynomial $u ш v$ is

$$\mathrm{som}(u ш v) = \binom{|u| + |v|}{|u|} = \frac{(|u| + |v|)!}{|u|!\,|v|!}.$$

In particular, given a letter z in Z, one has :

$$\forall \lambda, \ \mu \geq 0, \quad z^\lambda ш z^\mu = \binom{\lambda + \mu}{\mu} z^{\lambda+\mu}.$$

2. Calculus of the formal power series Evaluation

2.1. Evaluation of formal power series

Let $Z = \{z_0, z_1, \ldots, z_m\}$ be an finite alphabet.

Definition 2.1.1. : *We call* **input related to** Z *the given of a vector* $a = (a^{z_0} \;\; a^{z_1} \;\; \ldots \;\; a^{z_m})$ *of piecewise continous real valued functions defined over* $[0,t], (t \geq 0)$. *Conventionally the 0_component of any input is* $a^{z_0} \equiv 1$.

Following K.T. Chen ([2]), we call **path associated to the input** $a = (a^{z_0} \;\; a^{z_1} \;\; \ldots \;\; a^{z_m})$, *the time dependent vector* $\xi = {}^t(\xi_{z_0} \;\; \xi_{z_1} \;\; \ldots \;\; \xi_{z_m})$ *defined by*

$$\forall z \in Z, \quad \xi_z(\tau) = \int_0^\tau d\xi_z(\rho) = \int_0^\tau a^z(\rho)d\rho.$$

Thus we have $\xi_{z_0}(\tau) = \displaystyle\int_0^\tau d\rho = \tau$, *and for any letter* $z \in Z$, $\xi_z(0) = 0$.

Any formal (resp. analytical) control system can be viewed as a functional defined on the inputs, whose value is generally called the **output function** of the system. More specifically, following M. Fliess ([3]), we will call **causal analytical** any functional of the entry that can be expressed as a convergent Peano-Baker series $\displaystyle\sum_{w \in Z^*} < S|w > \int_0^t \delta_a w$, where occur the iterated integrals of the path ξ, defined as follows :

$$\int_0^t \delta_a \epsilon = 1 \quad \text{and} \quad \int_0^t \delta_a(vz) = \int_0^t \left(\int_0^\tau \delta_a v\right) d\xi_z(\tau)$$

(we use the symetric order of Fliess notations for some practical programing opportunity [7]).

In other words, any analytical functional can be encoded by some non-commutative power series, and its output can be obtained by the **Evaluation** procedure described below.

Here we shall call **Evaluation of** S **for the input** a **at time** t, the value of the functional encoded by S for the input a and the time t, also called **output function of** S. Hence the Evaluation of any formal power series in noncommuting variables can be interpreted as a signal depending of the independent parameters ξ_z, $z \in Z^*$. In fact, **Evaluation functions** can be viewed as a generalization of the **inverse Laplace transform**, as already pointed out by M. Fliess and al. ([3], [4]). Namely, the **Evaluation function** is defined as follows :

Definition 2.1.2. : *We will call* **Evaluation of the word** w **in** Z^*, *for the input* $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ *related to the finite alphabet* Z, *the iterated integral, noted* $\int_0^t \delta_a w$, *where this notation is defined by induction on the length of* w :

$$\mathcal{E}_a(w)(t) = \int_0^t \delta_a w = \begin{cases} 1 & \text{if} \quad w = \epsilon, \\ \displaystyle\int_0^t \mathcal{E}_a(v)(\tau)\, d\xi_z(\tau) & \text{if} \quad w = vz. \end{cases}$$

This definition is extended to $A \ll Z \gg$ in the following way :

Definition 2.1.3. : *We will call* **Evaluation of the formal power series** S **in** $A \ll Z \gg$, *for the input* $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ *related to the finite alphabet* Z, *when it is defined, the function* :

$$\mathcal{E}_a(S)(t) = \sum_{w \in Z^*} < S|w > \mathcal{E}_a(w)(t).$$

Theorem 2.1.1. : (uniqueness of Evaluation, M. Fliess [3]) *Given two formal power series* S *and* T *in* $A \ll Z \gg$. *For any input* $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ *related to the finite alphabet* Z *such that the series* $\sum_{w \in Z^*} < S|w > \mathcal{E}_a(w)$ *and* $\sum_{w \in Z^*} < T|w > \mathcal{E}_a(w)$ *are normaly convergent, then we have* :

$$\mathcal{E}_a(S) = \mathcal{E}_a(T) \qquad \Longleftrightarrow \qquad S = T.$$

2.2. Shuffle product and product of Evaluations

Let a be an input related to the finite alphabet $Z = \{z_0, z_1, \ldots, z_m\}$.

Lemma 2.2.1. : *Let* u *and* v *be two words in* Z^*. *Then the Evaluation of the polynomial* $u \sqcup\!\sqcup v$, *for the input* $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ *related to the finite alphabet* Z, *is given by* $\mathcal{E}_a(u \sqcup\!\sqcup v) = \mathcal{E}_a(u)\mathcal{E}_a(v)$.

Proof :

. The result is immediate for $|u| = 0$ or $|v| = 0$ because $u \sqcup\!\sqcup \epsilon = u$, $\epsilon \sqcup\!\sqcup v = v$ and $\mathcal{E}_a(\epsilon) = 1$.

. Now, suppose the result is true for any words u, v in Z^* that satisfy $|uv| \leq n$.

. If $|uv| = n+1$, then we can write $u = u_1 x$ and $v = v_1 y$ with $x,\ y \in Z$, $u_1,\ v_1 \in Z^*$ and $|u_1 v| = |uv_1| = n$. Using the fact that $u \sqcup\!\sqcup v = (u \sqcup\!\sqcup v_1)y + (u_1 \sqcup\!\sqcup v)x$, we have :

$$\begin{aligned}
\mathcal{E}_a(u \sqcup\!\sqcup v)(t) &= \mathcal{E}_a((u \sqcup\!\sqcup v_1)y)(t) + \mathcal{E}_a((u_1 \sqcup\!\sqcup v)x)(t) \\
&= \int_0^t \mathcal{E}_a(u \sqcup\!\sqcup v_1)(\tau)\ d\xi_y(\tau) + \int_0^t \mathcal{E}_a(u_1 \sqcup\!\sqcup v)(\tau)\ d\xi_x(\tau) \\
&= \int_0^t \Big[\mathcal{E}_a(u)(\tau)\mathcal{E}_a(v_1)(\tau)\Big] d\xi_y(\tau) + \int_0^t \Big[\mathcal{E}_a(u_1)(\tau)\mathcal{E}_a(v)(\tau)\Big] d\xi_x(\tau) \\
&\qquad \text{(by induction hypothesis)} \\
&= \int_0^t \mathcal{E}_a(u)(\tau) d[\mathcal{E}_a(v_1 y)(\tau)] + \int_0^t \mathcal{E}_a(v)(\tau) d[\mathcal{E}_a(u_1 x)(\tau)] \\
&= \Big[\mathcal{E}_a(u_1 x)(\tau)\mathcal{E}_a(v_1 y)(\tau)\Big]_0^t \quad \text{(integration by parts)} \\
&= \mathcal{E}_a(u)(t)\mathcal{E}_a(v)(t) \quad \bullet
\end{aligned}$$

Corollary 2.2.1. : *Let z be a letter in Z. For any positive integer n, we have $\mathcal{E}_a(z^n) = \frac{\xi_z^n}{n!}$. In particular case, $\mathcal{E}_a(z_0^n) = \frac{t^n}{n!}$.*

Proof : The result is immediate for $n = 0$. We suppose that the result is true for all ν, $0 \leq \nu \leq n$. For $\nu = n+1$, since $z^n \sqcup\!\sqcup z = (n+1)z^{n+1}$ (see 1.), and by the lemma 2.2.1., we have :

$$\mathcal{E}_a(z^{n+1}) = \frac{1}{n+1}\mathcal{E}_a(z^n \sqcup\!\sqcup z) = \frac{1}{n+1}\mathcal{E}_a(z^n)\mathcal{E}_a(z) = \frac{1}{n+1}\frac{\xi_z^n}{n!}\xi_z = \frac{\xi_z^{n+1}}{(n+1)!}.$$

The expression corresponding to the particular case of $z = z_0$ is obtained using the fact that $\xi_{z_0}(t) = t$ $\bullet$

Theorem 2.2.1. : (M. Fliess, [3]) *Let S and T be two formal power series in $A \ll Z \gg$. Then the Evaluation of the shuffle product $S \sqcup\!\sqcup T$, for the input $a = (a^{z_0}\quad a^{z_1}\quad \ldots\quad a^{z_m})$ related to the finite alphabet Z, is the product of the Evaluations of S and T :*

$$\mathcal{E}_a(S \sqcup\!\sqcup T) = \mathcal{E}_a(S)\mathcal{E}_a(T).$$

Proof : By the definition of $S \sqcup\!\sqcup T$ and the lemma 2.2.1., we have :

$$\begin{aligned}
\mathcal{E}_a(S \sqcup\!\sqcup T) &= \sum_{u,v \in Z^*} < S|u >< T|v > \mathcal{E}_a(u \sqcup\!\sqcup v) \\
&= \sum_{u \in Z^*}\sum_{v \in Z^*} < S|u >< T|v > \mathcal{E}_a(u)\mathcal{E}_a(v) \\
&= \Big(\sum_{u \in Z^*} < S|u > \mathcal{E}_a(u)\Big)\Big(\sum_{v \in Z^*} < T|v > \mathcal{E}_a(v)\Big) \\
&= \mathcal{E}_a(S)\mathcal{E}_a(T) \quad \bullet
\end{aligned}$$

2.3. Cauchy product and convolution

Let a be an input related to the finite alphabet $Z = \{z_0, z_1, \ldots, z_m\}$.

Lemma 2.3.1. : *Let $\mathcal{E}_a(S)$ be the Evaluation of the formal power series S, for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ related to the finite alphabet Z, and let z be a letter. For any integer n greater than or equal to 1, the Evaluation of the formal power series Sz^n is :*

$$\mathcal{E}_a(Sz^n)(t) = \int_0^t \mathcal{E}_a(S)(\tau)\frac{(\xi_z(t) - \xi_z(\tau))^{n-1}}{(n-1)!}d\xi_z(\tau).$$

In particular, for $z = z_0$, we have :

$$\mathcal{E}_a(Sz_0^n)(t) = \int_0^t \mathcal{E}_a(S)(\tau)\frac{(t-\tau)^{n-1}}{(n-1)!}d\tau.$$

<u>*Proof*</u> : For $n = 1$, we have :

$$\begin{aligned}
\mathcal{E}_a(Sz)(t) &= \sum_{w \in Z^*} < S|w > \mathcal{E}_a(wz)(t) \\
&= \sum_{w \in Z^*} < S|w > \int_0^t \mathcal{E}_a(w)(\tau)\, d\xi_z(\tau) \\
&= \int_0^t \Big(\sum_{w \in Z^*} < S|w > \mathcal{E}_a(w)(\tau)\Big) d\xi_z(\tau) \\
&= \int_0^t \mathcal{E}_a(S)(\tau)\, d\xi_z(\tau).
\end{aligned}$$

We suppose that the result is true for any ν, $1 \leq \nu \leq n$. For $\nu = n+1$, by induction hypothesis, we have :

$$\begin{aligned}
\mathcal{E}_a((Sz)z^n)(t) &= \int_0^t \mathcal{E}_a(Sz)(\tau)\frac{(\xi_z(t) - \xi_z(\tau))^{n-1}}{(n-1)!}d\xi_z(\tau) \\
&= \int_0^t \mathcal{E}_a(Sz)(\tau)\, d\Big[-\frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}\Big] \\
&= -\Big[\mathcal{E}_a(Sz)(\tau)\frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}\Big]_0^t \\
&\quad + \int_0^t \mathcal{E}_a(Sz)(\tau)\frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}d[\mathcal{E}_a(S\ z)(\tau)]
\end{aligned}$$

the first term is vanishing (after one integration by parts), then we have :

$$\mathcal{E}_a(Sz^{n+1})(t) = \int_0^t \mathcal{E}_a(S)(\tau)\frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}d\xi_z(\tau).$$

The particular form obtained for $z = z_0$ follows, since we have $\xi_0(t) = t$ •

Theorem 2.3.1. : *Given* $G \in A \ll Z \gg$ **a quasiregular** *formal power series,* $H \in A \ll z \gg$ *a formal power series, we set*

$$\psi(\xi(t)) = \frac{d}{dt}\mathcal{E}_a(G)(t) \qquad h(\xi_z(t)) = \mathcal{E}_a(H)(t).$$

With these notations, the Evaluation of the formal power series GH*, for the input* $a = (\,a^{z_0} \quad a^{z_1} \quad \dots \quad a^{z_m}\,)$ *related to the finite alphabet* Z*, is :*

$$\mathcal{E}_a(GH)(t) = \int_0^t \psi(\xi(\tau))h(\xi_z(t) - \xi_z(\tau))d\tau.$$

In particular, if H *is a formal power series in* $A \ll z_0 \gg$*, then :*

$$\mathcal{E}_a(GH)(t) = \int_0^t \psi(\xi(\tau))h(t - \tau)d\tau.$$

<u>*Proof*</u> :

<u>(*i*) First case</u> : $H = z^n, \ n \geq 0$:

. For $n = 0$, the result is immediate.

. If $n > 0$ then we have :

$$\begin{aligned}
\mathcal{E}_a(Gz^n)(t) &= \int_0^t \mathcal{E}_a(G)(\tau)\frac{(\xi_z(t) - \xi_z(\tau))^{n-1}}{(n-1)!}d\xi_z(\tau) \quad \text{(by the lemma 2.3.1.)}\\
&= \int_0^t \mathcal{E}_a(G)(\tau)d\left[-\frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}\right]\\
&= -\left[\frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}\mathcal{E}_a(G)(\tau)\right]_0^t\\
&\quad + \int_0^t \frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}d[\mathcal{E}_a(G)(\tau)] \quad \text{(integration by parts)}\\
&= \int_0^t \psi(\xi(\tau))\frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}d\tau \quad \text{(the first term vanishes)}\\
&= \int_0^t \psi(\xi(\tau))h(\xi_z(t) - \xi_z(\tau))d\tau.
\end{aligned}$$

(*ii*) General case : $H = \sum_{n\geq 0} H_n z^n$ and $h(\xi_z(t)) = \sum_{n\geq 0} H_n \frac{\xi_z^n(t)}{n!}$:

$$\begin{aligned}
\mathcal{E}_a(GH)(t) &= \sum_{n\geq 0} H_n \mathcal{E}_a(Gz^n)(t)\\
&= \sum_{n\geq 0} H_n \int_0^t \psi(\xi(\tau))\frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}d\tau \quad \text{(see (i))}\\
&= \int_0^t \psi(\xi(\tau))\sum_{n\geq 0} H_n \frac{(\xi_z(t) - \xi_z(\tau))^n}{n!}d\tau\\
&= \int_0^t \psi(\xi(\tau))h(\xi_z(t) - \xi_z(\tau))d\tau.
\end{aligned}$$

In particular, if $z = z_0$ then $\xi_0(t) = t$, hence we have the expected result •

2.4. Evaluations of some usual formal power series

Let z be a letter in Z. If the support of the formal power series S is a subset of z^*, then we have, for the input $a = (a^{z_0} \quad a^{z_1} \quad \dots \quad a^{z_m})$ related to the finite alphabet Z, the following Evaluations :

S	$\mathcal{E}_a(S)$
ϵ	1
z^n	$\frac{\xi_z^n(t)}{n!}$
$z^{*n}, n \geq 1$	$\exp(\xi_z(t)) \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\xi_z(t))^j}{j!}$
$\sum_{n\geq 0} c_n z^n$	$\sum_{n\geq 0} c_n \frac{\xi_z^n(t)}{n!}$
then, in particular	
$z^* = \sum_{n\geq 0} z^n$	$\exp(\xi_z(t))$
$\sum_{n\geq 0} nz^n = z^* z z^*$	$\xi_z(t)\exp(\xi_z(t))$
$\sum_{n\geq 0} (-1)^n z^{2n}$	$\cos(\xi_z(t))$
$\sum_{n\geq 0} (-1)^n z^{2n+1}$	$\sin(\xi_z(t))$

So the Evaluation of the generating series $S = \sum_{n\geq 0} c_n z^n$ is the associated "exponential generating series" $\mathcal{E}_a(S) = \sum_{n\geq 0} c_n \frac{\xi_z^n}{n!}$, and thus the Evaluation function is reduced to the usual exponential tranform as studied by D. Foata in [5]. In particular, if $z = z_0$ then we have $\xi_{z_0}(t) = t$, in this case we obtain the inverse Laplace-Borel transform.

Let G be a quasiregular formal power series. Its Evaluation, for the input $a = (a^{z_0} \quad a^{z_1} \quad \dots \quad a^{z_m})$ related to the finite alphabet Z, can be described by :

$$\mathcal{E}_a(G)(t) = \int_0^t \psi(\xi(\tau))d\tau.$$

Let z be a letter in Z. We have, for the input $a = (a^{z_0} \quad a^{z_1} \quad \dots \quad a^{z_m})$ related to the finite alphabet Z, the following Evaluation (see the convolution theorem) :

$$\mathcal{E}_a\left(G\sum_{n\geq 0} c_n z^n\right)(t) = \int_0^t \psi(\xi(\tau)) \sum_{n\geq 0} c_n \frac{[\xi_z(t) - \xi_z(\tau)]^n}{n!} d\tau,$$

consequently :

S	$\mathcal{E}_a(S)$
Gz^n	$\int_0^t \psi(\xi(\tau))\frac{(\xi_z(t)-\xi_z(\tau))^n}{n!}d\tau$
$Gz^* = G\sum_{n\geq 0} z^n$	$\int_0^t \psi(\xi(\tau))\exp(\xi_z(t)-\xi_z(\tau))d\tau$
$G\sum_{n\geq 0} nz^n$	$\int_0^t \psi(\xi(\tau))(\xi_z(t)-\xi_z(\tau)) \cdot \exp(\xi_z(t)-\xi_z(\tau))d\tau$
$G\sum_{n\geq 0}(-1)^n z^{2n}$	$\int_0^t \psi(\xi(\tau))\cos(\xi_z(t)-\xi_z(\tau))d\tau$
$G\sum_{n\geq 0}(-1)^n z^{2n+1}$	$\int_0^t \psi(\xi(\tau))\sin(\xi_z(t)-\xi_z(\tau))d\tau$

3. Computing examples

In the following examples $a = (a^{z_0} \quad a^{z_1} \quad \dots \quad a^{z_m})$ is an input related to the given finite alphabet Z.

Example 3.1.

This example gives the computation of the Evaluation of the formal power series z^{*n}, for any letter and for any z positive integer n.

Lemma 3.1.1. : *For any integer $n \geq 1$, we have :*

$$\mathcal{E}_a(z^{*n})(t) = \exp(\xi_z(t))g_n(\xi_z(t)),$$

where the g_n are polynomials in $\xi_z(t)$, and verifies the following inductive equations :

$$g_n(\xi_z(t)) = \begin{cases} 1 & \text{if} \quad n=1 \\ g_{n-1}(\xi_z(t)) + \int_0^t g_{n-1}(\xi_z(\tau))d\xi_z(\tau) & \text{if} \quad n>1. \end{cases}$$

<u>*Proof*</u> :
. Since $\mathcal{E}_a(z^*)(t) = \exp(\xi_z(t))$, we can write $g_1(\xi_z(t)) = 1$.
. We suppose that the result is true for any ν, $1 \leq \nu < n$.
. For $\nu = n$, recall the following identities :

$$\forall n \geq 1, \quad z^{*n} = z^{*n-1}z^* = z^{*n-1}(1+zz^*) = z^{*n-1} + z^{*n-1}zz^*.$$

Recall also that for any formal power series $S = \sum_{w \in Z^*} < S|w > w$ in $\mathbb{R} \ll Z \gg$, we have :

$$\begin{aligned}
\mathcal{E}_a(Sz)(t) &= \sum_{w \in Z^*} < S|w > \int_0^t \mathcal{E}_a(w)(\tau) d\xi_z(\tau) \\
&= \int_0^t \sum_{w \in Z^*} < S|w > \mathcal{E}_a(w)(\tau) d\xi_z(\tau) \\
&= \int_0^t \mathcal{E}_a(S)(\tau) d\xi_z(\tau) \\
&= \int_0^t \mathcal{E}_a(S)(\tau) a^z(\tau) d\tau.
\end{aligned}$$

Thus for $S = z^{*n-1}$, by induction hypothesis, we have :

$$\mathcal{E}_a(z^{*n-1}z)(t) = \int_0^t \exp(\xi_z(\tau)) g_{n-1}(\xi_z(\tau)) a^z(\tau) d\tau.$$

Using the convolution theorem, we obtain :

$$\begin{aligned}
\mathcal{E}_a(z^{*n-1}zz^*)(t) &= \int_0^t \exp(\xi_z(\tau)) g_{n-1}(\xi_z(\tau)) a^z(\tau) \exp(\xi_z(t) - \xi_z(\tau)) d\tau \\
&= \exp(\xi_z(t)) \int_0^t g_{n-1}(\xi_z(\tau)) d\xi_z(\tau).
\end{aligned}$$

Now we have, for any integer $n \geq 1$, the following equalities :

$$\begin{aligned}
\mathcal{E}_a(z^{*n})(t) &= \mathcal{E}_a(z^{*n-1})(t) + \mathcal{E}_a(z^{*n-1}zz^*)(t) \\
&= \exp(\xi_z(t)) \left[g_{n-1}(\xi_z(t)) + \int_0^t g_{n-1}(\xi_z(\tau)) d\xi_z(\tau) \right].
\end{aligned}$$

Hence the family $(g_n)_{n \geq 1}$ is the unique solution of the inductive equations

$$g_n(\xi_z(t)) = \begin{cases} 1 & \text{if} \quad n = 1 \\ g_{n-1}(\xi_z(t)) + \int_0^t g_{n-1}(\xi_z(\tau)) d\xi_z(\tau) & \text{if} \quad n > 1 \end{cases} \bullet$$

Lemma 3.1.2. : *The family* $g_n(\xi_z(t)) = \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{\xi_z^j(t)}{j!}$, *for any integer* $n \geq 1$, *is the unique solution of the inductive equations*

$$g_n(\xi_z(t)) = \begin{cases} 1 & \text{if} \quad n = 1 \\ g_{n-1}(\xi_z(t)) + \int_0^t g_{n-1}(\xi_z(\tau)) d\xi_z(\tau) & \text{if} \quad n > 1. \end{cases}$$

Proof : Let $G_1 = 1 \in \mathbb{R} \ll Z \gg$. We have clearly :

$$g_1(\xi_z(t)) = \mathcal{E}_a(G_1)(t).$$

Suppose that for any integer $n \geq 1$, $g_n(\xi_z(t)) = \mathcal{E}_a(G_n)(t)$ where G_n is a formal power series in $\mathbb{R} \ll Z \gg$. Thus we have :

$$\mathcal{E}_a(G_n)(t) = \mathcal{E}_a(G_{n-1})(t) + \mathcal{E}_a(G_{n-1}z)(t).$$

This equation is true if G_n satisfies $G_n = G_{n-1}(1+z)$. In other terms :

$$G_n = (1+z)^{n-1} = \sum_{j=0}^{n-1} \binom{n-1}{j} z^j.$$

Since $\mathcal{E}_a(z^j)(t) = \frac{\xi_z^j(t)}{j!}$, $j \geq 0$ (corollary 2.2.1.), we have :

$$\forall n \geq 1, \quad g_n(\xi_z(t)) = \mathcal{E}_a(G_n)(t) = \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{\xi_z^j(t)}{j!} \quad \bullet$$

Finally we obtain, as corollaries, the two following propositions :

Proposition 3.1.1. : *For any positive integer n, we have :*

$$\mathcal{E}_a(z^{*n})(t) = \begin{cases} 1 & \text{if} \quad n = 0 \\ \exp(\xi_z(t)) \displaystyle\sum_{j=0}^{n-1} \binom{n-1}{j} \frac{\xi_z^j(t)}{j!} & \text{if} \quad n > 0. \end{cases}$$

In the same manner, we have the following proposition :

Proposition 3.1.2. : *For any positive integer n, for any complex number α, we have :*

$$\mathcal{E}_a((\alpha z)^{*n})(t) = \begin{cases} 1 & \text{if} \quad n = 0 \\ \exp(\alpha\xi_z(t)) \displaystyle\sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\alpha\xi_z(t))^j}{j!} & \text{if} \quad n > 0. \end{cases}$$

Example 3.2.

Let G be a quasiregular formal power series in $\mathbb{R} \ll Z \gg$ such that its evaluation is :

$$\mathcal{E}_a(G)(t) = \int_0^t \psi(\xi(\tau))d\tau.$$

Let z be a letter in Z. Let S be a formal power series in $\mathbb{R} \ll Z \gg$ that verifies the following polynomial equation :

$$S + \beta_1 Sz + \beta_2 Sz^2 + \ldots \beta_n Sz^n = G,$$

that is $SK \doteq G$, where K is the formal power series in $\mathbb{R} \ll Z \gg$ defined by

$$K = \sum_{k=0}^{n} \beta_k z^k \quad \text{with} \quad \beta_0 = 1.$$

Since the constant term $< K|\epsilon >= \beta_0 = 1$ does not vanish, then the formal power series K^{-1} exists and it is a formal power series in the single commutative variable z. Suppose that K admits r complex distinguished roots $\mu_1, \ldots, \mu_r$ of respective multiplicity order $m_1, \ldots, m_r$ $\left(\sum_{l=1}^{r} m_l = n\right)$. One can express unically $K^{-1} = \dfrac{1}{\prod_{l=1}^{r}(z-\mu_l)^{m_l}}$ under partial fraction decomposition form :

$$K^{-1} = \sum_{l=1}^{r}\sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} H_{l,k},$$

where for any $l \in [1..r]$ and for any $k \in [1..m_l]$, $\lambda_{l,k} \in \mathbb{C}$ and each $H_{l,k}$ is of the following form :

$$H_{l,k} = \left(\frac{z}{\mu_l}\right)^{*k}.$$

So $S = GK^{-1}$ can be expressed as :

$$S = \sum_{l=1}^{r}\sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} GH_{l,k},$$

and by the convolution theorem, we obtain the Evaluation of S :

$$\mathcal{E}_a(S)(t) = \sum_{l=1}^{r}\sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} \int_0^t \psi(\xi(\tau)) h_{l,k}(\xi_z(t) - \xi_z(\tau))d\tau,$$

where $h_{l,k}(\xi_z(t))$ is the Evaluation of $H_{l,k}$:

$$h_{l,k}(\xi_z(t)) = \exp\left(\frac{\xi_z(t)}{\mu_l}\right) \sum_{j=0}^{k-1} \binom{k-1}{j} \frac{1}{j!} \left(\frac{\xi_z(t)}{\mu_l}\right)^j \qquad \text{(proposition 3.1.2.).}$$

Let us indicate that if $z = z_0$ then the above calculation corresponds mutadis mutandis to the "inverse Laplace transform" in the study of the systems discribed by one linear differential equation in the inputs and outputs.

Example 3.3.

Theorem 3.3.1. : *For any positive integer k, G_k is supposed a formal power series on only one letter z_{j_k} in Z and we note $g_k(\xi_{z_{j_k}}(t))$ its Evaluation. We set for any positive integer k, $S_k = G_0 z_{i_1} G_1 \ldots z_{i_k} G_k$, where $z_{i_1}, z_{i_2}, \ldots, z_{i_k}$ are k letters in Z. With these notations, for any integer $k > 0$, we have :*

$$\mathcal{E}_a(S_k)(t) = \int_0^t \int_0^{\tau_k} \cdots \int_0^{\tau_2} g_0(\xi_{z_{j_0}}(\tau_1)) g_1(\xi_{z_{j_1}}(\tau_2) - \xi_{z_{j_1}}(\tau_1)) \ldots$$
$$g_k(\xi_{z_{j_k}}(t) - \xi_{z_{j_k}}(\tau_k)) d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k).$$

Proof :
. For $k = 1$, by the lemma 2.3.1., we have :

$$\mathcal{E}_a(G_0 z_{i_1})(t) = \int_0^t g_0(\xi_{z_{j_0}}(t)) a^{z_{i_1}}(\tau) d\tau,$$

and by the convolution theorem, we have :

$$\begin{aligned}\mathcal{E}_a(G_0 z_{i_1} G_1)(t) &= \int_0^t g_0(\xi_{z_{j_0}}(\tau)) a^{z_{i_1}}(\tau) g_1(\xi_{z_{j_1}}(t) - \xi_{z_{j_1}}(\tau)) d\tau \\ &= \int_0^t g_0(\xi_{z_{j_0}}(\tau)) g_1(\xi_{z_{j_1}}(t) - \xi_{z_{j_1}}(\tau)) d\xi_{z_{i_1}}(\tau).\end{aligned}$$

. The result is supposed true for any ν, $1 \leq \nu \leq k-1$.
. For $\nu = k$, by the lemma 2.3.1., we have :

$$\mathcal{E}_a(S_{k-1} z_{i_k})(t) = \int_0^t \int_0^{\tau_k} \cdots \int_0^{\tau_2} g_0(\xi_{z_{j_0}}(\tau_1)) g_1(\xi_{z_{j_1}}(\tau_2) - \xi_{z_{j_1}}(\tau_1)) \ldots$$
$$g_{k-1}(\xi_{z_{j_{k-1}}}(\tau_k) - \xi_{z_{j_{k-1}}}(\tau_{k-1})) d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k).$$

Finally, by the convolution theorem, we have :

$$\mathcal{E}_a(S_k)(t) = \int_0^t \int_0^{\tau_k} \cdots \int_0^{\tau_2} g_0(\xi_{z_{j_0}}(\tau_1)) g_1(\xi_{z_{j_1}}(\tau_2) - \xi_{z_{j_1}}(\tau_1)) \ldots$$
$$g_k(\xi_{z_{j_k}}(t) - \xi_{z_{j_k}}(\tau_k)) d\xi_{z_{i_1}}(\tau_1) \ldots d\xi_{z_{i_k}}(\tau_k) \quad \bullet$$

We deduce the two following results :

Corollary 3.3.1. : *Let $p_0, p_1, \ldots, p_k$ be $k+1$ positive integers. Let $z_{i_1}, z_{i_2}\ldots, z_{i_k}$, be letters in $Z_0 = Z\setminus\{z_0\}$. Then the Evaluation of the word $z_0^{p_0} z_{i_1} z_0^{p_1} \ldots z_{i_k} z_0^{p_k}$ is (see the corollary 2.2.1.) :*

$$\int_0^t \int_0^{\tau_k} \int_0^{\tau_{k-1}} \cdots \int_0^{\tau_2} \frac{\tau_1^{p_0}(\tau_2-\tau_1)^{p_1}\ldots(t-\tau_k)^{p_k}}{p_0!p_1!\ldots p_k!} d\xi_{z_{i_1}}(\tau_1)\ldots d\xi_{z_{i_k}}(\tau_k).$$

Corollary 3.3.2. : *Let $c_0, c_1, \ldots, c_k$ be $k+1$ complex numbers. Let $p_0, p_1, \ldots, p_k$ be $k+1$ positive integers. Let $z_{i_1}, z_{i_2}\ldots, z_{i_k}$, be letters in $Z_0 = Z\setminus\{z_0\}$. Then the Evaluation of the rational fraction $(c_0z_0)^{*p_0} z_{i_1}(c_1z_0)^{*p_1}\ldots z_{i_k}(c_kz_0)^{*p_k}$ is :*

$$\int_0^t \int_0^{\tau_k} \int_0^{\tau_{k-1}} \cdots \int_0^{\tau_2} f_0(\tau_1)f_1(\tau_2-\tau_1)\ldots f_k(t-\tau_k)d\xi_{z_{i_1}}(\tau_1)\ldots d\xi_{z_{i_k}}(\tau_k),$$

with (see the proposition 3.1.2.) :

$$\forall n\in[0..k],\quad f_n(t) = \begin{cases} 1 & \text{if} \quad p_n = 0 \\ \exp(c_nt)\sum_{j=0}^{p_n-1}\binom{p_n-1}{j}\frac{(c_nt)^j}{j!} & \text{if} \quad p_n > 0. \end{cases}$$

The first result is used in [6] to give the Taylor expansion of Volterra kernels. The second result is also presented in ([4]), it allows to compute iteratively the Volterra kernels of the solution of certain nonlinear differential equations with forcing terms. In [7], we give a concise MACSYMA program, allowing a particularly quick computation of the Evaluation of these series S_k.

4. References

[1] **J. Berstel and C. Reutenauer**, *Rational Series and thèir languages*, Springer-Verlag, 1988.

[2] **K.T. Chen**, *Iterated path integrals*, Bull. Amer. Math. Soc., vol 83, 1977, pp. 831-879.

[3] **M. Fliess**, *Fonctionnelles causales non linéaires et indéterminées non commutatives*, Bull. Soc. Math France, N°109, 1981, pp. 3-40..

[4] **M. Fliess, M. Lamnabhi and F. Lamnabhi-Lagarrigue**, *An Algebraic Approach to Nonlinear Functional Expansions*, IEEE Trans. Circ. Syst., CAS-30, 1983, pp. 554-570.

[5] **D. Foata**, *La série génératrice exponentielle dans les problèmes d'énumération*, Séminaire de Math. Sup., Presses de L'Université de Montréal, Montréal, 1974.

[6] **V. Hoang Ngoc Minh and G. Jacob**, *Symbolic Calculus and Volterra Series*, IFAC Symposium "Non Linear Control Systems Design", Capri, June 1989.

[7] **V. Hoang Ngoc Minh and G. Jacob**, *Evaluation transform with kernel function*, to appear in Calsyf.

ANNEXE B

SYMBOLIC CALCULUS AND VOLTERRA SERIES

(IFAC Symposium
"Non Linear Control Systems Design",
Capri-Italy, June 1989)

SYMBOLIC CALCULUS AND VOLTERRA SERIES

V. Hoang Ngoc Minh and G. Jacob
L.I.F.L. - U.A. 369 C.N.R.S.
Université Lille I
59655 Villeneuve d'Ascq Cedex
France

__Abstract__ : *Given a nonlinear analytical dynamic system (affine with respect to the input), its output function can be viewed as a signal parametrized by the primitives of the input functions. This signal can be formally described by its generating series. Hence we obtain a symbolic transform that generalizes Laplace transform of signals depend only on the time. We develop here the basic tools of that symbolic calculus. We prove a correspondence theorem between certain convolutions of signals and Cauchy products of generating series. Finally the Taylor expansion of triangular Volterra kernels is simply deduced.*

__Keywords__ : *Algebraic system theory, generating series, Laplace transforms, nonlinear systems, system analysis, transfer functions*

0. INTRODUCTION

Let (S) be a nonlinear dynamic system described by :

$$(S)\quad \begin{cases} \dot{q}(t) = \sum_{i=0}^{m} a^i(t) A_i(q) \\ y(t) = h(q(t)) \end{cases}$$

We can associate to the observation h its generating series $\sigma h = \sum_{w \in Z^*} < \sigma h | w > w$, that is a formal power series in noncommuting variables belonging to a finite alphabet $Z = \{z_0, \ldots, z_m\}$. By the fundamental formula of M. Fliess ([4]), (also called "Peano-Baker formula"), the output $y(t)$ defined by the observation h is obtained by the replacement in σh of each word w by the associated "iterated integral" $\int_0^t \delta_a w$ related to the input $a = (a^0 \ \ldots \ a^m)$ defined over $[0, t]$, $t \geq 0$ (the input $a^0(t) \equiv 1$ encodes the autonomous part of the system). Here we will call "Evaluation" of the word w, this associated iterated integral, $\mathcal{E}_a(w) = \int_0^t \delta_a w$, and we will call "Evaluation" of the power series S (submitted to some convergence condition) "the output", $\mathcal{E}_a(S)$, obtained by replacing each word w in S by its Evaluation $\mathcal{E}_a(w)$. Then the Evaluation of S can be viewed as a signal, depending on the time t, and on the m independent parameters $\xi_i(t) = \int_0^t a^i(\tau) d\tau$, $i \in [1..m]$.

This point of view leads naturally to develop a noncommuting "symbolic calculus" in the nonlinear area, that generalizes the Heaviside calculus ([5]). So, the notions of "transfer functions" (generating series in one variable) and "impulsive responses", coding signals produced by linear systems, can be generalized to generating series in $m + 1$ variables and Volterra series, coding signals produced by nonlinear analytical systems. The Evaluation function $\mathcal{E}_a$ corresponds to the "inverse Laplace transform".

Our goal is to develop here the basic tools of this symbolic calculus on noncommuting variables. We prove a correspondence theorem between certain convolutions of signals and Cauchy products of generating power series. Finally the Taylor expansion of Volterra kernels is simply deduced. These kernels have been well studied elsewhere in [2], [4] and [10]. These results are written in a natural manner in the noncommutative formal power series area.

1. WORDS, LANGUAGES AND FORMAL POWER SERIES ([1])

By an **alphabet**, Z, we mean a finite non-empty set. The elements of Z are called **letters**. A **word** over Z is a finite sequence w of letters in $Z : w = z_{j_1} z_{j_2} \ldots z_{j_k}$. The length of the word w, denoted by $|w|$, is its length as sequence of letters. The **empty word** ϵ is the empty sequence of letters ($|\epsilon| = 0$). We note Z^* the set of all words over Z. We have $Z^* = \bigcup_{k \geq 0} Z^k$, where Z^k, $k \geq 0$, is the set of all words over Z of length exactly k.

The **concatenation product** of two words u, v in Z^* is obtained by "juxtaposition" of u and v. Thus if we have $u = z_{j_1} z_{j_2} \ldots z_{j_k}$ and $v = z_{i_1} z_{i_2} \ldots z_{i_l}$, then uv is the word $z_{j_1} z_{j_2} \ldots z_{j_k} z_{i_1} z_{i_2} \ldots z_{i_l}$ in Z^*. This product is associative, its identity element is ϵ. One can verify easily that Z^* is **the free monoid generated** by the aphabet Z ([16]), with respect to the concatenation product.

Any subset $L \subset Z^*$ is called a **language**. Given two languages L_1, L_2, their product, noted by ".", is the language $L_1.L_2$ defined by extension of the product of words :

$$L_1.L_2 = \{w \in Z^*,\ w = w_1 w_2,\ w_1 \in L_1,\ w_2 \in L_2\}.$$

This product is associative, its identity element is $\{\epsilon\}$. The symbol "." will be omitted when there is no ambiguity.

We call **formal power series on the associative variables** $z \in Z$ **(noncommuting if** $\operatorname{card} Z \geq 2$**)** with coefficients in an unitary commutative ring A any mapping :

$$S : Z^* \longrightarrow A$$
$$w \longmapsto < S | w > .$$

We denote by $A \ll Z \gg$ the set of all formal power series. A formal power series S will be written as the formal sum $\sum_{w \in Z^*} < S | w > w$, where $< S | w >$ is the coefficient of w.
A formal power series $S \in A \ll Z \gg$ will be said **quasi-regular** if and only if we have $< S | \epsilon > = 0$.

The **characteristic series** of any language L is the formal power series $\operatorname{car}(L)$ defined by :

$$\forall w \in Z^*, \quad < \operatorname{car}(L) | w > = \begin{cases} 1 & \text{if} \quad w \in L \\ 0 & \text{if} \quad w \notin L \end{cases}$$

The support of $S \in A \ll Z \gg$ is the language defined by :

$$\operatorname{supp}(S) = \{w \in Z^* \mid < S | w > \neq 0\}.$$

Let L be a language over Z. Then $\mathrm{supp}[\mathrm{car}(L)] = L$. Let S be a formal power series in $A \ll Z \gg$. Then :

$\mathrm{car}[\mathrm{supp}(S)] = S \iff \forall w \in Z^*, < S|w > \in \{0,1\}$.

Any formal power series with coefficients in $\{0,1\}$ can be identified with its support. So, if L is a language over Z, we denote again its characteristic series by L.

A formal power series P with finite support is called polynomial. The degree of P is defined by :

$$\deg(P) = \begin{cases} -\infty & \text{if} \quad P = 0, \\ \max\{|w|, \quad w \in \mathrm{supp}(P)\} & \text{if} \quad P \neq 0. \end{cases}$$

We denote by $A < Z >$ the set of all polynomials in associative variables $z \in Z$ with coefficients in A. A polynomial is homogenous of degree $k \in \mathbb{N}$ if $\mathrm{supp}(Q) \subset Z^k$.

The sum of two formal power series $S, T \in A \ll Z \gg$ is the formal power series $S + T$ defined by :

$$\forall w \in Z^*, \quad < S+T|w > = < S|w > + < T|w > .$$

This operation is associative and commutative.

The Cauchy product, noted by ".", of two formal power series S,T in $A \ll Z \gg$ is the formal power series $S.T$ in $A \ll Z \gg$ defined by :

$$\forall w \in Z^*, \quad < S.T|w > = \sum_{u,v \in Z^*, uv=w} < S|u >< T|v > .$$

The symbol "." will be omitted when there is no ambiguity.

The set $A \ll Z \gg$ is a (non commutative) ring, with respect to the sum and to the Cauchy product. Any $a \in A$ is identified with the series $a\epsilon$. Then A is identified with an unitary subring of $A \ll Z \gg$. In this way $A \ll Z \gg$ is an A-algebra. In the same manner, for the Cauchy product, $A < Z >$ is a subalgebra of $A \ll Z \gg$.

Let z_0 be a distinguished letter. We set $Z_0 = Z \setminus \{z_0\}$. By ([13]), we obtain $Z^* = (z_0^* Z_0)^* z_0^* = \sum_{k \geq 0} (z_0^* Z_0)^k z_0^*$, with $z_0^* = \sum_{n \geq 0} z_0^n$ and $(z_0^* Z_0)^k = \sum_{\substack{w_1,\ldots,w_k \in z_0^* Z_0 \\ w_1 \ldots w_k = w}} w$. Hence :

$$\forall S \in A \ll Z \gg, \quad S = \sum_{k \geq 0} \Big[\sum_{w \in (z_0^* Z_0)^k z_0^*} < S|w > w \Big] = \sum_{k \geq 0} V_k(S).$$

The shuffle product of two formal power series S, T in $A \ll Z \gg$, noted again by "$\sqcup\!\sqcup$", is the formal power series $S \sqcup\!\sqcup T$ in $A \ll Z \gg$ defined by :

$$S \sqcup\!\sqcup T = \sum_{u,v \in Z^*} < T|u >< S|v > u \sqcup\!\sqcup v,$$

where $u \sqcup\!\sqcup v$ is the homogenous polynomial of degree $|u|+|v|$, defined recursively in the following way :

$$\forall u \in Z^* : \quad u \sqcup\!\sqcup \epsilon = \epsilon \sqcup\!\sqcup u = u$$

$$\forall u, v \in Z^*, \forall x, y \in Z : ux \sqcup\!\sqcup vy = ((ux) \sqcup\!\sqcup v)y + (u \sqcup\!\sqcup (vy))x.$$

This product is associative, commutative, and distributive with respect to the sum. With respect to the sum and to the shuffle product, $A \ll Z \gg$ is a commutative A-algebra.

The Lie bracket of two polynomials P, Q is defined by $[P,Q] = PQ - QP$. It is anticommutative and verifies the Jacobi identity. $Lie < Z >$ is referred to be the smallest sub A-module of $A \ll Z \gg$, which contains the letters of Z, and which is stable by Lie bracket. It is well known that $Lie < Z >$ is the free Lie algebra over Z ([16]). An element of $Lie < Z >$ is called a Lie polynomial. Any Lie polynomial is quasiregular.

A formal power series S is called a Lie series if it can be written as $\sum_{k \geq 1} P_k$, where P_k is a homogenous Lie polynomial of degree $k \geq 1$ ([9]). This writing is unique. The Lie bracket of two Lie series $S = \sum_{k \geq 1} P_k$, $T = \sum_{l \geq 1} Q_l$ is $[S,T] = \sum_{k,l \geq 1} [P_k, Q_l]$. For this product, the set $Lie \ll Z \gg$ of Lie series is a Lie algebra. Any Lie series is quasiregular.

2. Calculus of the formal power series evaluation

2.1. Evaluation of formal power series

Let $Z = \{z_0, z_1, \ldots, z_m\}$ be a finite alphabet.

Definition 2.1.1. :
We call **input related to** Z *the given for each letter* z_i *in* Z, *of a vector* $a = (a^0 \quad a^1 \quad \ldots \quad a^m)$ *of piecewise continous real valued functions defined over* $[0,t]$, $(t \geq 0)$. *Conventionally the 0-component of any input is* $a^0 \equiv 1$. *Following K.T. Chen* ([4], [15]), *we will call* **path associated to the input** $a = (a^0 \quad a^1 \quad \ldots \quad a^m)$, *the time dependent vector* $\xi = {}^t(\xi_0 \quad \xi_1 \quad \ldots \quad \xi_m)$ *defined by :*

$$\forall i \in [0..m], \quad \xi_i(\tau) = \int_0^\tau d\xi_i(\rho) = \int_0^\tau a^i(\rho) d\rho.$$

Thus we have $\xi_0(\tau) = \int_0^\tau d\rho = \tau$, $\xi_i(0) = 0$, $(i \in [0..m])$.

Any analytical dynamic system can be viewed as defining a functional on the inputs , of which value is generally called the output function of the system. More generally, following M. Fliess, we call causal analytical functional any functional of the entry that can be expressed as a convergent Peano-Baker series $\sum_{w \in Z^*} < S|w > \int_0^t \delta_a w$, where occur the iterated integrals of the path ξ, defined as follows (using the symetric order of Fliess notations) :

$$\int_0^t \delta_a \epsilon = 1 \quad \text{and} \quad \int_0^t \delta_a (v z_i) = \int_0^t \left(\int_0^\tau \delta_a v \right) d\xi_i(\tau).$$

In other words, any analytical functional can be encoded by some noncommutative power series, and its output can be obtained by the Evaluation procedure described below. Here we shall call Evaluation of S for the input a at time t, the value of the functional encoded by S for the input a and the time t, also called output function of S. Hence the Evaluation of any formal power series in noncommuting variables can be interpreted as a signal depending of the independent parameters ξ_i, $i \in [0..m]$. In fact, Evaluation function can be viewed as a generalization of the inverse Laplace transform, as already pointed out by M. Fliess and al. ([4], [5]). Namely, the Evaluation function is defined as follows :

Definition 2.1.2. :
The Evaluation of the word w **in** Z^*, *for the input* $a = (a^0 \quad a^1 \quad \ldots \quad a^m)$ *defined over* $[0,t]$, $(t \geq 0)$, *is defined by induction on the length of* w *as follows :*

$$\mathcal{E}_a(w)(t) = \begin{cases} 1 & \text{if} \quad w = \epsilon, \\ \displaystyle\int_0^t \mathcal{E}_a(v)(\tau)\, d\xi_i(\tau) & \text{if} \quad w = v z_i. \end{cases}$$

Definition 2.1.3. :
We will call **Evaluation of the formal power series** S *in* $A \ll Z \gg$, *for the input* $a = (a^0 \quad a^1 \quad \ldots \quad a^m)$ *defined over* $[0,t]$, $(t \geq 0)$, *when it is defined, the function*

$$\mathcal{E}_a(S)(t) = \sum_{w \in Z^*} < S|w > \mathcal{E}_a(w)(t).$$

2.2. Shuffle and product of Evaluations

Let a be an input related to the finite alphabet Z and defined over the compact interval $[0,t] \in \mathbb{R}_+$.

Lemma 2.2.1. :
Let u and v be two words in Z^*. Then the Evaluation of the polynomial $u \sqcup\!\sqcup v$, for the input $a = (a^0 \quad a^1 \ldots \quad a^m)$, is given by $\mathcal{E}_a(u \sqcup\!\sqcup v) = \mathcal{E}_a(u)\mathcal{E}_a(v)$.

The proof can be done easily by induction on the length of uv and results of the integration by parts. We deduce the following result ([7]) :

Theorem 2.2.1. :
Let S, T be two formal power series in $A \ll Z \gg$. Then the Evaluation of the shuffle product $S \sqcup\!\sqcup T$, for the input $a = (a^0 \quad a^1 \quad \ldots \quad a^m)$, is the product of the Evaluations of S and T is $\mathcal{E}_a(S \sqcup\!\sqcup T) = \mathcal{E}_a(S)\mathcal{E}_a(T)$.

2.3. Cauchy product and convolution

Let a be an input related to the finite alphabet Z and defined over the compact interval $[0,t] \in \mathbb{R}_+$.

Lemma 2.3.1. : ([7])
If $\mathcal{E}_a(S)$ is the Evaluation of the formal power series S, for the input $a = (a^0 \quad a^1 \quad \ldots \quad a^m)$, then let z_i be a letter in Z. Let n be an integer greater than or equal to 1. The Evaluation of the formal power series Sz_i^n is :

$$\mathcal{E}_a(Sz_i^n)(t) = \int_0^t \frac{(\xi_i(t) - \xi_i(\tau))^{n-1}}{(n-1)!}\mathcal{E}_a(S)(\tau)\, d\xi_i(\tau).$$

In particular, for $z_i = z_0$, we have :

$$\mathcal{E}_a(Sz_0^n)(t) = \int_0^t \frac{(t-\tau)^{n-1}}{(n-1)!}\mathcal{E}_a(S)(\tau)d\tau.$$

Proof :
The series Sz_i^n can be written as $\sum_{w \in Z^*} <S|w>\, wz_i^n$.

. For $n = 1$, we have :

$$\begin{aligned}
\mathcal{E}_a(Sz_i)(t) &= \sum_{w \in Z^*} <S|w>\, \mathcal{E}_a(wz_i)(t)\\
&= \sum_{w \in Z^*} <S|w> \int_0^t \mathcal{E}_a(w)(\tau)\, d\xi_i(\tau)\\
&= \int_0^t \Big(\sum_{w \in Z^*} <S|w>\, \mathcal{E}_a(w)(\tau)\Big) d\xi_i(\tau)\\
&= \int_0^t \mathcal{E}_a(S)(\tau)\, d\xi_i(\tau).
\end{aligned}$$

. We suppose the result is true for any ν, $1 \leq \nu \leq n$.
. For $\nu = n+1$, by induction hypothesis, we have :

$$\begin{aligned}
\mathcal{E}_a((Sz_i)z_i^n)(t) &= \int_0^t \frac{(\xi_i(t) - \xi_i(\tau_1))^{n-1}}{(n-1)!}\mathcal{E}_a(Sz_i)(\tau_1)\, d\xi_i(\tau_1)\\
&= \int_0^t \mathcal{E}_a(Sz_i)(\tau)\, d\Big[-\frac{(\xi_i(t) - \xi_i(\tau_1))^n}{n!}\Big]\\
&= -\Big[\frac{(\xi_i(t) - \xi_i(\tau))^n}{n!}\mathcal{E}_a(Sz_i)(\tau)\Big]_0^t\\
&\quad + \int_0^t \frac{(\xi_i(t) - \xi_i(\tau))^n}{n!}[d\mathcal{E}_a(Sz_i)(\tau)],
\end{aligned}$$

the first term is vanishing, then :

$$\mathcal{E}_a(Sz_i^{n+1})(t) = \int_0^t \frac{(\xi_i(t) - \xi_i(\tau))^n}{n!}\mathcal{E}_a(S)(\tau)\, d\xi_i(\tau).$$

The particular case $z_i = z_0$ is immediate •

Theorem 2.3.1. : (Convolution Theorem, [7])
Let z_i be a letter in Z. Let H be a formal power series in $A \ll z_i \gg$, so that we have $\mathcal{E}_a(H)(t) = h(\xi_i(t))$. Let G be a quasiregular formal power series in $A \ll Z \gg$. We set

$$\psi(\xi(t)) = \frac{d}{dt}\mathcal{E}_a(G)(t).$$

With these notations, the Evaluation, for the input $a = (a^0 \quad a^1 \quad \ldots \quad a^m)$, of the formal power series GH is

$$\mathcal{E}_a(GH)(t) = \int_0^t \psi(\xi(\tau))h(\xi_i(t) - \xi_i(\tau))d\tau.$$

In particular, if H is a formal power in $A \ll z_0 \gg$, then :

$$\mathcal{E}_a(GH)(t) = \int_0^t \psi(\xi(\tau))h(t-\tau)d\tau.$$

Proof :
(i) First case : $H = z_i^n$, $n \geq 0$:

. For $n = 0$, the result is immediate.
. If $n > 0$ then by induction hypothesis, we have :

$$\begin{aligned}
\mathcal{E}_a(Gz_i^n)(t) &= \int_0^t \mathcal{E}_a(G)(\tau)\frac{(\xi_i(t) - \xi_i(\tau))^{n-1}}{(n-1)!}d\xi_i(\tau)\\
&= \int_0^t \mathcal{E}_a(G)(\tau)d\Big[-\frac{(\xi_i(t) - \xi_i(\tau))^n}{n!}\Big]\\
&= -\Big[\frac{(\xi_i(t) - \xi_i(\tau))^n}{n!}\mathcal{E}_a(G)(\tau)\Big]_0^t\\
&\quad + \int_0^t \frac{(\xi_i(t) - \xi_i(\tau))^n}{n!}d[\mathcal{E}_a(G)(\tau)],
\end{aligned}$$

the first term is vanishing (becaeause $<G|\epsilon> = 0$), then :

$$\mathcal{E}_a(Gz_i^n)(t) = \int_0^t \psi(\xi(\tau))h(\xi_i(t) - \xi_i(\tau))d\tau.$$

(ii) General case : $H = \sum_{n \geq 0} H_n z_i^n$:

In this case we have $h(\xi_i(t)) = \sum_{n \geq 0} H_n \frac{\xi_i^n(t)}{n!}$, and :

$$\begin{aligned}
\mathcal{E}_a(GH)(t) &= \sum_{n \geq 0} H_n \mathcal{E}_a(Gz_i^n)(t)\\
&= \sum_{n \geq 0} H_n \int_0^t \psi(\xi(\tau))\frac{(\xi_i(t) - \xi_i(\tau))^n}{n!}d\tau \text{ (see (i))}\\
&= \int_0^t \psi(\xi(\tau))\sum_{n \geq 0} H_n \frac{(\xi_i(t) - \xi_i(\tau))^n}{n!}d\tau\\
&= \int_0^t \psi(\xi(\tau))h(\xi_i(t) - \xi_i(\tau))d\tau.
\end{aligned}$$

The particular case $z_i = z_0$ is immediate •

2.4. Evaluations of some usual series ([7])

Let z_i be a letter in Z. If the support of the formal power series S is a subset of z_i^*, then we have, for the input $a = (a^0 \quad a^1 \quad \ldots \quad a^m)$ defined over $[0,t]$, $(t \geq 0)$, the following Evaluations :

S	$\mathcal{E}_a(S)$
ϵ	1
z_i^n	$\frac{\xi_i^n(t)}{n!}$
$z_i^{*n}, n \geq 1$	$\exp(\xi_i(t)) \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\xi_i(t))^j}{j!}$
$\sum_{n \geq 0} c_n z_i^n$ then, in particular	$\sum_{n \geq 0} c_n \frac{\xi_i^n(t)}{n!}$
$z_i^* = \sum_{n \geq 0} z_i^n$	$\exp(\xi_i(t))$
$\sum_{n \geq 0} n z_i^n$	$\xi_i(t)\exp(\xi_i(t))$
$\sum_{n \geq 0} (-1)^n z_i^{2n}$	$\cos(\xi_i(t))$
$\sum_{n \geq 0} (-1)^{n+1} z_i^{2n+1}$	$\sin(\xi_i(t))$

So the Evaluation of the generating series $S = \sum_{n\geq 0} c_n z_i^n$ is the "exponential generating series" $\mathcal{E}_a(S) = \sum_{n\geq 0} c_n \frac{\xi_i^n(t)}{n!}$.
Let G be a quasiregular formal power series. One can write $\mathcal{E}_a(G)(t) = \int_0^t \psi(\xi(\tau))d\tau$. Let z_i be a letter in Z.
Then we have :

$$\mathcal{E}_a\Big(G\sum_{n\geq 0} c_n z_i^n\Big)(t) = \int_0^t \psi(\xi(\tau)) \sum_{n\geq 0} c_n \frac{[\xi_i(t)-\xi_i(\tau)]^n}{n!} d\tau,$$

consequently :

S	$\mathcal{E}_a(S)$
Gz_i^n	$\int_0^t \psi(\xi(\tau))\frac{(\xi_i(t)-\xi_i(\tau))^n}{n!}d\tau$
$Gz_i^* = G\sum_{n\geq 0} z_i^n$	$\int_0^t \psi(\xi(\tau))\exp(\xi_i(t)-\xi_i(\tau))d\tau$
$G\sum_{n\geq 0} n z_i^n$	$\int_0^t \psi(\xi(\tau))(\xi_i(t)-\xi_i(\tau))$ $\exp(\xi_i(t)-\xi_i(\tau))d\tau$
$G\sum_{n\geq 0}(-1)^n z_i^{2n}$	$\int_0^t \psi(\xi(\tau))\cos(\xi_i(t)-\xi_i(\tau))d\tau$
$G\sum_{n\geq 0}(-1)^{n+1} z_i^{2n+1}$	$\int_0^t \psi(\xi(\tau))\sin(\xi_i(t)-\xi_i(\tau))d\tau$

3. CHEN SERIES ([4], [15])

The following formal power series will be called **Chen series of the input a defined over $[0,t]$** :

$$\mathcal{C}_a = \sum_{w\in Z^*} \mathcal{E}_a(w)(t)w.$$

The properties of integration by parts show (lemma 2.2.1.) that $\mathcal{C}_a$ verifies the "Friedrich criterion", namely :

$$\forall u,\ v \in Z^*,\quad <\mathcal{C}_a|u\,ш\,v> = <\mathcal{C}_a|u><\mathcal{C}_a|v>.$$

Hence $\mathcal{C}_a$ is an exponential of some Lie series ([9]) :

$$\mathcal{C}_a = \exp\Big(\sum_{k\geq 1} L_k(\mathcal{C}_a)\Big),$$

where $L_k(\mathcal{C}_a)$, $k\geq 1$ is a homogenous Lie polynomial of degree k. This result is a generalization of the Baker-Campbell-Hausdorff formula. The power series $\mathcal{C}_a$ can also be expressed as an infinite exponential product of elements of a Hall basis of the free Lie algebra $Lie \ll Z \gg$ ([15]).

Definition 3.1. :
We call **concatenation of two inputs a, b** *defined over* $[0,t_a]$, $[0,t_b]$, *the new input* $a\sharp b$ *defined over* $[0,t_a+t_b]$ *:*

$$a\sharp b(t) = a(t) \quad \text{for} \quad 0\leq t < t_a$$
$$a\sharp b(t_a+t) = b(t) \quad \text{for} \quad 0\leq t<t_b.$$

Proposition 3.1. : ([14])
With the above notation, we have $\mathcal{C}_{a\sharp b} = \mathcal{C}_a\mathcal{C}_b$.

The proof given in [14] uses the structural differential equation satisfied by $\mathcal{C}_{a\sharp b}$. Here we give an direct proof :

Definition 3.2. :
Let θ be a fixed real. Let e be an input defined over the compact interval $[\theta,\theta+t]$, $t\geq 0$. We defined the **θ_translated iterated integral** *as :*

$$\int_\theta^{\theta+t}\delta_e w = \begin{cases} 1 & \text{if} \quad w=\epsilon, \\ \int_0^t \Big(\int_\theta^{\theta+\tau}\delta_e v\Big)d\xi_i(\theta+\tau) & \text{if} \quad w = vz_i. \end{cases}$$

Remark :
The iterated integral $\int_0^t \delta_e w$ is not an ordinary integral. In particular, the additivity propriety is not at all satisfied $\int_0^{T+t}\delta_e w \neq \int_0^T \delta_e w + \int_T^{T+t}\delta_e w$ (If $w=\epsilon$, it would imply 1=1+1 !).

The proposition 3.1. is equivalent to the following lemma

Lemma 3.1. : (temporal translation lemma)
Let θ be a fixed real. Let e be an input defined over the compact interval $[0,\theta+t]$, $t\geq 0$. Then we have :

$$\mathcal{E}_e(w)(\theta+t) = \sum_{\substack{u,v\in Z^*\\ uv=w}} \mathcal{E}_e(u)(\theta)\int_\theta^{\theta+t}\delta_e v = \sum_{\substack{u,v\in Z^*\\ uv=w}} \int_0^\theta \delta_e u \int_\theta^{\theta+t}\delta_e v.$$

Proof :
Since $\int_0^{\theta+t}\delta_e\epsilon = 1$, then the result is immediate for $w=\epsilon$.
We suppose the result is satisfied for all words w, $|w|\leq n$. If $|w| = n+1$, w can be written as $w = w_1 z_i$, $w_1\in Z^*$, and $z_i\in Z$, then :

$$\mathcal{E}_e(w)(\theta+t) = \int_0^{\theta+t}\mathcal{E}_e(w_1)(\rho)\ d\xi_i(\rho)$$

(additivity of ordinary integral)

$$= \int_0^\theta \mathcal{E}_e(w_1)(\rho)d\xi_i(\rho) + \int_\theta^{\theta+t}\mathcal{E}_e(w_1)(\rho)\ d\xi_i(\rho)$$

($\rho = \theta+\tau$)

$$= \mathcal{E}_e(w_1z_i)(\theta) + \int_0^t \mathcal{E}_e(w_1)(\theta+\tau)\ d\xi_i(\theta+\tau)$$

(induction hypothesis)

$$= \mathcal{E}_e(w_1z_i)(\theta) + \int_0^t \sum_{\substack{u_1,v_1\in Z^*\\ u_1v_1=w_1}}\mathcal{E}_e(u_1)(\theta)\int_\theta^{\theta+\tau}\delta_e v_1\ d\xi_i(\theta+\tau)$$

(distributivity of ordinary integral)

$$= \mathcal{E}_e(w_1z_i)(\theta).1 + \sum_{\substack{u_1,v_1\in Z^*\\ u_1v_1=w_1}}\mathcal{E}_e(u_1)(\theta)\int_0^t\int_\theta^{\theta+\tau}\delta_e v_1\ d\xi_i(\theta+\tau)$$

(definition 3.2.)

$$= \mathcal{E}_e(w_1z_i)(\theta)\int_\theta^{\theta+t}\delta_e\epsilon + \sum_{\substack{u_1,v_1\in Z^*\\ u_1v_1=w_1}}\mathcal{E}_e(u_1)(\theta)\int_\theta^{\theta+t}\delta_e(v_1z_i)$$

$$= \mathcal{E}_e(w_1z_i)(\theta)\int_\theta^{\theta+t}\delta_e\epsilon + \sum_{\substack{u,v\in Z^*,v\neq\epsilon\\ uv=w_1z_i}}\mathcal{E}_e(u)(\theta)\int_\theta^{\theta+t}\delta_e v$$

$$= \sum_{\substack{u,v\in Z^*\\ uv=w}}\mathcal{E}_e(u)(\theta)\int_\theta^{\theta+t}\delta_e v \quad \bullet$$

Proof of the proposition 3.1. :
By the definition 3.1. and the lemma 3.1., we have :

$$\mathcal{C}_{a\sharp b} = \sum_{w\in Z^*}\mathcal{E}_{a\sharp b}(w)(t_a+t_b)w$$

$$= \sum_{w\in Z^*}\Big[\sum_{\substack{u,v\in Z^*\\ uv=w}}\mathcal{E}_{a\sharp b}(u)(t_a)\int_{t_a}^{t_a+t_b}\delta_{a\sharp b}v\Big]w$$

$$= \sum_{w\in Z^*}\Big[\sum_{\substack{u,v\in Z^*\\ uv=w}}\mathcal{E}_a(u)(t_a)\mathcal{E}_b(v)(t_b)\Big]uv = \mathcal{C}_a\mathcal{C}_b.$$

Indeed, integrated integrals $\int_{t_a}^{t_a+t_b}\delta_{a\sharp b}v$ and $\mathcal{E}_b(v)(t_b)$ are equals, having the same inductive definition $\bullet$

Example :
If a, b are the constant inputs over $[0,t_a]$, $[0,t_b]$ then $C_a = \exp(t_a\sum_{i=0}^{m} a^i x_i)$ and $C_b = \exp(t_b\sum_{i=0}^{m} b^i x_i)$. In this case $C_{a\sharp b} = \exp(t_a\sum_{i=0}^{m} a^i x_i)\exp(t_b\sum_{i=0}^{m} b^i x_i)$.

From 1., the Chen series C_a of input a defined over $[0,t]$ can be written as $\sum_{k\geq 0} V_k(C_a)$, where for any positive integers k, we have $V_k(C_a) = \sum_{w\in(z_0^* Z_0)^k z_0^*} \mathcal{E}_a(w)(t)w$. Any word w in $(z_0^* Z_0)^k z_0^*$ can be expressed as :

$$w = z_0^{\alpha_0} z_{i_1} z_0^{\alpha_1}\dots z_{i_k} z_0^{\alpha_k},$$
(for $j\in[0..k]$, $\alpha_{i_j}\geq 0$, $z_{i_j}\in Z_0$).

By the lemma 2.3.1. and the theorem 2.3.1., we obtain easily ([7]) :

$$\mathcal{E}_a(w)(t) = \int_0^t\int_0^{\tau_k}\dots\int_0^{\tau_2}\frac{\tau_1^{\alpha_0}\dots(\tau_k-\tau_{k-1})^{\alpha_{k-1}}(t-\tau_k)^{\alpha_k}}{\alpha_0!\alpha_1!\dots\alpha_{k-1}!\alpha_k!}$$
$$a^{i_1}(\tau_1)\dots a^{i_k}(\tau_k)d\tau_1\dots d\tau_{k-1}d\tau_k.$$

Theorem 3.1. :
Let $a = (a^0\ \ a^1\ \ \dots\ \ a^m)$ be an input defined over $[0,t]$. If C_a is the Chen series of this input, then :

$$\forall k\geq 0,\ V_k(C_a) = \sum_{z_{i_1}\dots z_{i_k}\in Z_0^k}\int_0^t\int_0^{\tau_k}\dots\int_0^{\tau_2} a^{i_1}(\tau_1)\dots a^{i_k}(\tau_k)$$
$$[e^{\tau_1 z_0} z_{i_1} e^{(\tau_2-\tau_1)z_0}.z_{i_k} e^{(t-\tau_k)z_0}]d\tau_1\dots d\tau_k.$$

Proof :
By the above result, we have for any $k\geq 0$:

$$V_k(C_a) = \sum_{z_{i_1}\dots z_{i_k}\in Z_0^k}\ \sum_{\alpha_0,\dots,\alpha_k\geq 0}\int_0^t\int_0^{\tau_k}\dots\int_0^{\tau_2} a^{i_1}(\tau_1)\dots a^{i_k}(\tau_k)$$
$$\frac{\tau_1^{\alpha_0}(\tau_2-\tau_1)^{\alpha_1}!.(t-\tau_k)^{\alpha_k}}{\alpha_0!\alpha_1!\dots!\alpha_k!}d\tau_1\dots d\tau_k z_0^{\alpha_0} z_{i_1} z_0^{\alpha_1}\dots z_{i_k} z_0^{\alpha_k}$$
$$= \sum_{z_{i_1}\dots z_{i_k}\in Z_0^k}\int_0^t\int_0^{\tau_k}\dots\int_0^{\tau_2} a^{i_1}(\tau_1)\dots a^{i_k}(\tau_k)\Big[\sum_{\alpha_0,\dots,\alpha_k\geq 0}$$
$$\frac{[\tau_1 z_0]^{\alpha_0} z_{i_1}[(\tau_2-\tau_1)z_0]^{\alpha_1}!.z_{i_k}[(t-\tau_k)z_0]^{\alpha_k}}{\alpha_0!\alpha_1!\dots!\alpha_k!}\Big]d\tau_1\dots d\tau_k$$
$$= \sum_{z_{i_1}\dots z_{i_k}\in Z_0^k}\int_0^t\int_0^{\tau_k}\dots\int_0^{\tau_2} a^{i_1}(\tau_1)\dots a^{i_k}(\tau_k)$$
$$[e^{\tau_1 z_0} z_{i_1} e^{(\tau_2-\tau_1)z_0}\dots z_{i_k} e^{(t-\tau_k)z_0}]d\tau_1\dots d\tau_k \quad \bullet$$

Since $C_a = \sum_{k\geq 0} V_k(C_a)$, we have the following result :

Corollary 3.1. :
Let $a = (a^0\ \ a^1\ \ \dots\ \ a^m)$ be an input defined over $[0,t]$. The Chen series C_a is described by :

$$C_a = \sum_{k\geq 0}\sum_{z_{i_1}\dots z_{i_k}\in Z_0^k}\int_0^t\int_0^{\tau_k}\dots\int_0^{\tau_2} a^{i_1}(\tau_1)\dots a^{i_k}(\tau_k)$$
$$[e^{\tau_1 z_0} z_{i_1} e^{(\tau_2-\tau_1)z_0}\dots z_{i_k} e^{(t-\tau_k)z_0}]d\tau_1\dots d\tau_k.$$

Example :
If the alphabet Z is reduced to $\{z_0\}$, then :

$$C_a = V_0(C_a) = \sum_{n\geq 0}\mathcal{E}_a(z_0^n)(t)z_0^n = \sum_{n\geq 0}\frac{(tz_0)^n}{n!} = e^{tz_0}.$$

4. REALIZATION

4.1. Generating series

Let Q be a analytical manifold of dimension N. $C^\omega(Q)$ is an algebra of analytical functions over Q. Let Z be a finite alphabet, we call **realization of Z over Q, or polysystem over Q** ([12]), the given for each letter z in Z, of an analytical vector field A_z over Q. Let f be a analytical function in $C^\omega(Q)$. We note $(A_z\circ f)_{|q}$ the evaluation at q of the analytical function $A_z\circ f$.

Notation :
Let w be a word in Z^* :
. if $w=\epsilon$, A_ϵ is the identity,
. if $w=w'z$, A_w is the differential operator $A_{w'}A_z$.

In all the sequel, $\{A_z\}_{z\in Z}$ is supposed to be some fixed polysystem.

Definition 4.1.1. : ([4])
Let f be a analytical function in $C^\omega(Q)$. **The generating series of f related to the $\{A_z\}_{z\in Z}$ with coefficients in $C^\omega(Q)$** *is the formal power series defined by*

$$\sigma f = \sum_{w\in Z^*}(A_w\circ f)w\in C^\omega(Q)\langle\langle Z\rangle\rangle.$$

The generating series of f at the state $q\in Q$ *is the formal power series defined by :*

$$\sigma_q f = \sum_{w\in Z^*}(A_w\circ f)_{|q}w\in \mathbb{R}\langle\langle Z\rangle\rangle.$$

It is well known that the series $\sigma_{q(0)}f$ caracterizes completly the local behaviour (on a neighbourhood of the state $q(0)\in Q$) of the polysystem defined by the analytical vector fields $\{A_z\}_{z\in Z}$, the observation $f\in C^\omega(Q)$, and the initial state $q(0)\in Q$ ([4]).

Theorem 4.1.1. : (M. Fliess [4])
The application σ is a morphism of the algebra $C^\omega(Q)$ to the algebra of formal power series $C^\omega(Q)\langle\langle Z\rangle\rangle$ for the shuffle product.

4.2. Left action of $A\langle Z\rangle$ over $C^\omega(Q)$ ([6])

Let "$*$" be the left action of $A\langle Z\rangle$ over $C^\omega(Q)$, that associates to any $z\in Z$, and to any $f\in C^\omega(Q)$, the Lie derivative of f with respect to the vector field A_z :

$$z*f = A_z\circ f.$$

The action of a word $w\in Z^*$ is defined as follows :

$$\forall f\in C^\omega(Q),\quad w*f = A_w\circ f.$$

The action of a polynomial $P\in A\langle Z\rangle$ is defined by

$$\forall f\in C^\omega(Q),\quad P*f = \sum_{w\in supp(P)}\langle P|w\rangle\, w*f,$$

(this sum is finite since the support of P is finite).
Let S be a formal power series in $A\langle\langle Z\rangle\rangle$. We consider an infinite sum $\sum_{w\in Z^*}\langle S|w\rangle\, w*f$, if this sum is normally convergent ([4], [8]), then we put :

$$S*f = \sum_{w\in Z^*}\langle S|w\rangle\, w*f.$$

Let f be a analytical function in $C^\omega(Q)$. The generating series associated to f related to the analytical vector fields $\{A_z\}_{z\in Z}$ can be written as :

$$\sigma f = \sum_{w\in Z^*}(w*f)w.$$

Let a be an input related to the alphabet Z defined over the compact interval $[0,t]\subset\mathbb{R}_+$ (see 2.1.). By analyticity of the vector fields $\{A_s\}_{s\in Z}$ and the function f, the sum $\sum_{w\in Z^*}(w\bullet f)\mathcal{E}_a(w)$ is normally convergent. So we can express the Evaluation of the generating series σf as the action of the Chen series C_a on f :

$$\mathcal{E}_a(\sigma f)=\sum_{w\in Z^*}(w\bullet f)\mathcal{E}_a(w)=\left(\sum_{w\in Z^*}\mathcal{E}_a(w)w\right)\bullet f=C_a\bullet f.$$

Theorem 4.2.1. :
Let f (resp. g) be a analytical function in $C^\omega(Q)$. Then :

$$C_a\bullet(fg)=(C_a\bullet f)(C_a\bullet g)$$

Proof :
By the theorems 4.1.1. and 2.2.1., we have :

$$C_a\bullet(fg)=\mathcal{E}_a(\sigma(fg))=\mathcal{E}_a(\sigma f\, \text{ш}\, \sigma g)$$
$$=\mathcal{E}_a(\sigma f)\mathcal{E}_a(\sigma g)=(C_a\bullet f)(C_a\bullet g)\quad\bullet$$

4.3. Volterra series ([2], [10])

Let f be an analytical function in $C^\omega(Q)$. The generating series associated to f related to the analytical vector fields $\{A_s\}_{s\in Z}$ is noted by σf. By the corollary 4.1., the Evaluation of σf, for the input a, can be expressed as :

$$\mathcal{E}_a(\sigma f)=C_a\bullet f=\sum_{k\geq 0}V_k(C_a)\bullet f=\sum_{k\geq 0}\sum_{u=z_{i_1}\ldots z_{i_k}\in Z_0^k}$$
$$\int_0^t\int_0^{\tau_1}\ldots\int_0^{\tau_2}a^{i_1}(\tau_1)\ldots a^{i_k}(\tau_k)\kappa_k^u(t,\tau_1,\ldots,\tau_k)d\tau_1\ldots d\tau_k,$$

that is the Volterra series associated to f ([2], [10]), with triangular Volterra kernels, κ_k^u, $k\geq 0$, $u\in Z_0^k$, defined by

$$\kappa_k^u(t,\tau_1,\ldots,\tau_k)=e^{\tau_1 z_0}z_{i_1}e^{(\tau_2-\tau_1)z_0}\ldots z_{i_k}e^{(t-\tau_k)z_0}\bullet f,$$
$$0\leq\ldots\leq\tau_k\leq t.$$

Using the formula $e^\alpha\beta e^{-\alpha}=\sum_{\nu\geq 0}\frac{1}{\nu!}ad_\alpha^\nu\beta=e^{ad_\alpha}\beta$, where for any integer $\nu\geq 1$, $ad_\alpha^\nu\beta$, is the ν- iterated Lie bracket $[\alpha,[\ldots,[\alpha,\beta]\ldots]]$ and $ad_\alpha^0\beta=\beta$, then for any positive integer k, for any word $u=z_{i_1}\ldots z_{i_k}$ in Z_0^k the kernels $\kappa_k^u(t,\tau_1,\ldots,\tau_k)$, $(0\leq\ldots\leq\tau_k\leq t)$, can be written as ([5])

$$\kappa_k^u(t,\tau_1,\ldots,\tau_k)=e^{\tau_1 z_0}z_{i_1}e^{-\tau_1 z_0}e^{\tau_2 z_0}\ldots z_{i_k}e^{-\tau_k z_0}e^{t z_0}\bullet f$$
$$=(e^{ad_{\tau_1 z_0}}z_{i_1})\ldots(e^{ad_{\tau_k z_0}}z_{i_k})e^{t z_0}\bullet f.$$

Hence, generating series, considered as a symbolic encoding of input/output behaviour, allow a fast computation of Volterra kernels. In fact, the Evaluation procedure allows to easy derive the temporal behaviour from the symbolic description. Thus, we get a generalization of the notion of "transfer functions" (generating series on one variable) and "impulse response", encoding signals produced by linear or multilinear systems, and the Heaviside calculus (Fourier and Laplace transforms) as already pointed out by M. Fliess and al. ([4], [5]).

Remarks :

. The second writing of the triangular Volterra kernels depends on t, τ_1, $\tau_2,\ldots,$ τ_k, and do not depends on $\tau_2-\tau_1,\ldots,$ $t-\tau_k$ as in the first writing. That allowed F. Lamnabhi-Lagarrigue to obtain easily the hamitonian expressions of Volterra kernels in the study of singular optimal control ([11]).

. These kernels are obtained for non initialized systems. The importance of these non initialized systems is already pointed out by D. Claude in the study of decoupling by state feedback ([3]), see also ([6]).

5. REFERENCES

[1] **J. Berstel and Ch. Reutenaurer**, *Séries rationelles et leurs langages*, Masson, E.R.I., 1984.

[2] **R.W. Brockett**, *Volterra Series and Geometric Control Theory*, Automatica, 12, pp. 167-176, 1976.

[3] **D. Claude**, *Découplage et Linéarisation des systèmes non linéaires par bouclages statiques*, Thèse d'état, Université de Paris Sud, Centre d'Orsay, 1986.

[4] **M. Fliess**, *Fonctionnelles causales non linéaires et indéterminées non commutative*, Bull. Soc. Math France, N 109, pp. 3-40, 1981.

[5] **M. Fliess, M. Lamnabhi and F. Lamnabhi-Lagarrigue**, *An agebraic approach to nonlinear functional expensions*, IEEE CS-30, 1983, pp. 554-570.

[6] **V. Hoang Ngoc Minh and G. Jacob**, *Découplage des systèmes dynamiques non linéaires*, LIFL technical publication, IT 128, 1988.

[7] **V. Hoang Ngoc Minh**, *Eléments d'un calcul symbolique pour les systèmes dynamiques non linéaires*, Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles non commutatives", Lille, December 1988.

[8] **T. Huillet, A. Monin and G. Salut**, *Réversibilité des systèmes non linéaires à commandes non ponctuelles*, to appear.

[9] **G. Jacob and N. Oussous**, *Sur un résultat de Ree : Séries de Lie et Algèbres de mélange*, LIFL technical publication, IT 103, 1987.

[10] **C. Lesiak and A.J. Krener**, *The existence and Uniqueness of Volterra Series for Nonlinear Systems*, IEEE AC-23, N 6, pp. 1090-1095, 1978.

[11] **F. Lamnabhi-Lagarrigue**, *Séries de Volterra et Commande optimale singulière*, Thèse d'état, Université de Paris Sud, Centre d'Orsay, 1985.

[12] **C. Lobry**, *Contrôlabilité des systèmes non linéaires*, SIAM, J. Cont. Opt., Vol. 8, N 4, pp. 573-605, 1970.

[13] **A. Salomaa and M. Soittola**, *Automata-Theoretic Aspects of formal power series*, Springer-Verlag, 1978.

[14] **H.J. Susmann**, *Lie brackets and local controlability : A sufficient condition for scalar-input systems*, SIAM, J. Cont. Opt., Vol. 21, N 5, pp. 686-713, 1983.

[15] **H.J. Susmann**, *A product expansion for Chen Series*, in "Theory and Applications of Nonlinear Control Systems", C.I. Byrns and Lindquist (eds). 323-335, 1986.

[16] **G. Viennot**, *Algèbres de Lie libres et Monoïdes libres*, Lecture Notes in Mathematics, 691, 1978.

ANNEXE C

EVALUATION TRANSFORM AND SYMBOLIC CALCULUS FOR NON LINEAR CONTROL SYSTEMS

("Ninth International Conference Analysis and Opimization of Systems" Antibes, June 1990)

Evaluation Transform and Symbolic Calculus for Nonlinear Control Systems

V. Hoang Ngoc Minh & G. Jacob, L.I.F.L. - U.A. 369 C.N.R.S.
Université Lille I, 59655 Villeneuve d'Ascq, France.

Abstract : Given a nonlinear control system, one can view its output function as a signal, parametrized by the primitives of the input functions. This signal can be formally described by its Fliess' power series, that is a formal power series on noncommuting variables. The temporal behaviour of the system can be derived from this symbolic description by a transform, that we call "Evaluation transform" and that generalizes the inverse Laplace transform to the nonlinear area. We developpe here the basic tools of that symbolic calculus by introducing a "kernel" for our Evaluation transform. This kernel can be viewed as some "temporal memory" of the system in the Volterra's meaning as well as in the programmation meaning.

1. Introduction and notations

Following M. Fliess ([2]) the output $y(t)$ of any nonlinear control system can be symbolically described by its generating series $S = \sum_{w \in Z^*} < S|w > w$ (also called Fliess' power series) that is a formal power series on noncommuting variables belonging to an alphabet $Z = \{z_0, z_1, \ldots, z_m\}$. This series is nothing else as the adequate generalization of the Heaviside calculus ([2], [3], [8], [9]) in the nonlinear area : the notions of transfer function and of impulsive response, coding signals produced by linear or multilinear control systems, are generalized to generating series and Volterra series, coding signals produced by nonlinear control systems. The inverse Laplace transform allows to recover the signal symbolically described by some transfer function. In the same way, the Evaluation transform ([5], [6]) allows in return to easy derive the temporal behaviour of the system from its symbolic description by computing the Evaluation $E_a(w)$ of each word w, for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$, as being the iterated integral $\int_0^t \delta_a w$. It follows from the Fliess' fundamental formula (also called Peano-Baker formula) which presents the output $y(t)$ as the Evaluation of the generating series $S : y(t) = \sum_{w \in Z^*} < S|w > \mathcal{E}_a(w)(t)$. So the output $y(t)$ can be viewed as a signal depending on the parameters $\xi_z(t) = \int_0^t a^z(\tau)d\tau,\ z \in Z$ ([5], [6]).

We develop here, in continuation of [5] and [6], the basic elements of this symbolic calculus using the notations of Evaluation via kernel functions which plays the rule of temporal memory of systems in the Volterra's meaning as well as in the programmation meaning. So the Evaluation transform becomes a functional depending on the kernel and on the inputs. This systematic treatement has been used in [7] to write MACSYMA programs by use of its recursive definition and of the recursive internal representation by the binary tree of the linear combinations of the noncommutative rational fractions described by $(c_0 z_{j_0})^{*p_0} z_{i_1} (c_1 z_{j_1})^{*p_1} z_{i_2} (c_2 z_{j_2})^{*p_2} \ldots z_{i_{k-1}} (c_{k-1} z_{j_{k-1}})^{*p_{k-1}} z_{i_k} (c_k z_{j_k})^{*p_k}$, where $p_0, \ldots, p_k$ are integers, $c_0, \ldots, c_k$ are complex numbers, $z_{j_0}, \ldots, z_{j_k}$ and $z_{i_0}, \ldots, z_{i_k}$ are letters in Z.

Recall that a **formal power series on the associative variables** $z \in Z$ **(non-commuting** if card $Z \geq 2$) with coefficients in $\mathbb{K}$ ([1]), is any mapping :

$$S : Z^* \longrightarrow \mathbb{K}$$
$$w \longmapsto < S|w >,$$

and the set of all formal power series over Z is denoted by $\mathbb{K} \ll Z \gg$.

A formal power series S will be written as a formal sum $\sum_{w \in Z^*} < S|w > w$, where $< S|w >$ is the **coefficient** of the word w in S. A formal power series $S \in \mathbb{K} \ll Z \gg$ will be said **quasiregular** if and only if its constant term $< S|\epsilon >$ is equal to 0.

The **sum** of two formal power series S and T is the formal power series $S + T$ in $\mathbb{K} \ll Z \gg$ defined by :

$$\forall w \in Z^*, \quad < S + T|w > = < S|w > + < T|w > .$$

The **Cauchy product**, noted by ".", of two formal power series S and T is the formal power series $S.T$ in $\mathbb{K} \ll Z \gg$ defined by :

$$\forall w \in Z^*, \quad < S.T|w > = \sum_{u,v \in Z^*, uv = w} < S|u >< T|v > .$$

The symbol "." will be omitted when there is no ambiguity.

For any quasiregular formal power series S in $\mathbb{K} \ll Z \gg$, S^* represents classically the formal power series $\sum_{n \geq 0} S^n$. In commutative variables, it coincides with the rational fraction $1/(1 - S)$, and in this case we have $S^{*n} = \big(1/(1 - S)\big)^n$.

2. Symbolic calculus

2.1. Evaluation of formal power series

Let $Z = \{z_0, z_1, \ldots, z_m\}$ be a finite alphabet.

Definition 2.1.1. : *We will call* **input relative to** Z *the given of a vector* $a = (\,a^{z_0} \;\; a^{z_1} \;\; \ldots \;\; a^{z_m}\,)$ *of piecewise continous real valued functions defined on a compact interval* $[0, t] \subset \mathbb{R}_+$. *Conventionally the 0_component of any input is* $a^{z_0} \equiv 1$.

Following K.T. Chen ([2]), *we will call* **path associated to the input** $a = (\,a^{z_0} \;\; a^{z_1} \;\; \ldots \;\; a^{z_m}\,)$, *the time dependent vector* $\xi = {}^t(\,\xi_{z_0} \;\; \xi_{z_1} \;\; \ldots \;\; \xi_{z_m}\,)$ *defined as follows :*

$$\forall z \in Z, \quad \xi_z(\tau) = \int_0^\tau d\xi_z(\rho) = \int_0^\tau a^z(\rho) d\rho.$$

Thus we have $\xi_{z_0}(\tau) = \int_0^\tau d\rho = \tau$, *and for any* $z \in Z$, $\xi_z(0) = 0$.

In all the sequel, we suppose that the function f is Stieltjes integrable with respect to $\{\xi_z\}_{z \in Z}$ defined over $[0, t]$, $(t \geq 0)$, and f vanishes at zero. The unit step (vanishing at zero) is noted "un" :

$$\forall \tau \in \,]0, t], \quad un(\tau) = 1, \quad un(0) = 0.$$

Definition 2.1.2. : *The* **Evaluation of the word w in Z^* with respect to the kernel f**, *for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ related to the finite alphabet Z, is defined by induction on the length of w as follows :*

$$\mathcal{E}_a(f;w)(t) = \begin{cases} f(t) & \text{if} \quad w = \epsilon, \\ \displaystyle\int_0^t \mathcal{E}_a(f;v)(\tau)\, d\xi_z(\tau) & \text{if} \quad w = vz. \end{cases}$$

This definition is extended to $I\!K \ll Z \gg$ in the following way :

Definition 2.1.3. : *We will call* **Evaluation of the formal power series S in $I\!K \ll Z \gg$ with respect to the kernel f**, *for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ related to the finite alphabet Z, when it is defined, the functional :*

$$\mathcal{E}_a(f;S) = \sum_{w \in Z^*} < S|w > \mathcal{E}_a(f;w).$$

In particular, for $f = un$, the **Evaluation of the formal power series S in $I\!K \ll Z \gg$**, *for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ related to the finite alphabet Z, is the function* ([5], [6]) *:*

$$\mathcal{E}_a(S) = \mathcal{E}_a(un;S).$$

According to the Fliess' fundamental formula, we see the Evaluation of a formal power series S can be viewed as a transform that associates to S the signal depending on the primitives $\{\xi_z\}_{z \in Z}$ of the inputs functions, and the Evaluation transform is nothing else as a generalization of inverse Laplace and Fourier transforms ([5], [6]).

The two following lemmas can be obtained easily by induction :

Lemma 2.1.1. : *Let u, v be two words in Z^*. Then :*

$$\mathcal{E}_a(f;uv) = \mathcal{E}_a(\mathcal{E}_a(f;u);v).$$

Lemma 2.1.2. : *Let z be a letter in Z. For any $n \geq 1$, we have :*

$$\mathcal{E}_a(f;z^n)(t) = \int_0^t f(\tau)\frac{(\xi_z(t) - \xi_z(\tau))^{n-1}}{(n-1)!} d\xi_z(\tau).$$

In particular, for $z = z_0$, we have :

$$\mathcal{E}_a(f;z_0^n)(t) = \int_0^t f(\tau)\frac{(t-\tau)^{n-1}}{(n-1)!} d\tau.$$

From the lemma 2.1.2., we can see that the Evaluation transform is an "extension on noncommutative variables" of the exponential transform of the "ordinary generating series" $S = \sum_{n \geq 0} c_n z^n$, which associates to S the "exponential generating series" $\mathcal{E}_a(S) = \sum_{n \geq 0} c_n \frac{\xi_z^n}{n!}$ well known in "combinatorics" ([4]) :

Corollary 2.1.1. : *Let z be a letter in Z. Then for any $n \geq 0$, $\mathcal{E}_a(z^n)(t) = \frac{\xi_z^n(t)}{n!}$. In particular, for $z = z_0$, then for any $n \geq 0$, $\mathcal{E}_a(z_0^n)(t) = \frac{t^n}{n!}$.*

Theorem 2.1.1. : *Let S and T be two formal power series in $\mathbb{K} \ll Z \gg$. Let r be a scalar. Then :*

$$\mathcal{E}_a(f; S + r.T) = \mathcal{E}_a(f; S) + r.\mathcal{E}_a(f; T).$$

Proof : The proof is immediate since the linearity of the ordinary integrals •

Theorem 2.1.2. : *Let S and T be two formal power series in $\mathbb{K} \ll Z \gg$. Then :*

$$\mathcal{E}_a(f; S.T) = \mathcal{E}_a(\mathcal{E}_a(f; S); T).$$

Proof : Actually, by the definition 2.1.3., we have :

$$\begin{aligned}
\mathcal{E}_a(\mathcal{E}_a(f; S); T) &= \sum_{u \in Z^*} \sum_{v \in Z^*} < S|u >< T|v > \mathcal{E}_a(\mathcal{E}_a(f; u); v) \\
&= \sum_{u \in Z^*} \sum_{v \in Z^*} < S|u >< T|v > \mathcal{E}_a(f; uv) \quad \text{(lemma 2.1.1.)} \\
&= \mathcal{E}_a\Big(\sum_{u \in Z^*} \sum_{v \in Z^*} < S|u >< T|v > uv\Big) \\
&= \mathcal{E}_a(f; S.T) \quad \bullet
\end{aligned}$$

The introduction of the kernel f for the Evaluation function $\mathcal{E}_a$ ([5], [6]) allows to give a notion of memory for the system. This kernel can be viewed as the temporal memory of the system in the Volterra's meaning as well as in the programmation meaning ([7]), that justifies our approach. So the Evaluation transform becomes a functional depending on the kernel f and on the inputs $\{a^z\}_{z \in Z}$. For implement these functionals, we used the λ-notation of MACSYMA and the recursive internal representation by binary trees of the linear combinations of the noncommutative rational fractions described by $(c_0 z_{j_0})^{*p_0} z_{i_1} (c_1 z_{j_1})^{*p_1} z_{i_2} (c_2 z_{j_2})^{*p_2} .. z_{i_{k-1}} (c_{k-1} z_{j_{k-1}})^{*p_{k-1}} z_{i_k} (c_k z_{j_k})^{*p_k}$, where $p_0, \ldots, p_k$ are integers, $c_0, \ldots, c_k$ are complex numbers, $z_{j_0}, \ldots, z_{j_k}$ and $z_{i_0}, \ldots, z_{i_k}$ are letters.

Theorem 2.1.3. : (Convolution theorem) *Let z be a letter in Z. Let H be the formal power series on the only letter in z. The Evaluation with respect to the kernel f, for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ related to the finite alphabet Z, of the formal power series H is :*

$$\mathcal{E}_a(f; H)(t) = \int_0^t h(\xi_z(t) - \xi_z(\tau)) df(\tau),$$

where $h(\xi_z(t))$ the Evaluation of the formal power series H. In particular, if H is a formal power series in $\mathbb{K} \ll z_0 \gg$, then :

$$\mathcal{E}_a(f;H)(t) = \int_0^t h(t-\tau)df(\tau).$$

Proof : (i) First case : $H = z^n$, $n \geq 0$: For $n = 0$, the result is immediate. If $n > 0$ then we have :

$$\begin{aligned}
\mathcal{E}_a(f;z^n)(t) &= \int_0^t f(\tau)\frac{(\xi_z(t)-\xi_z(\tau))^{n-1}}{(n-1)!}d\xi_z(\tau) \quad \text{(lemma 2.1.2.)}\\
&= \int_0^t f(\tau)d\Big[-\frac{(\xi_z(t)-\xi_z(\tau))^n}{n!}\Big]\\
&= -\Big[\frac{(\xi_z(t)-\xi_z(\tau))^n}{n!}f(\tau)\Big]_0^t + \int_0^t \frac{(\xi_z(t)-\xi_z(\tau))^n}{n!}df(\tau)\\
&\quad \text{(integration by parts)}\\
&= \int_0^t h(\xi_z(t)-\xi_z(\tau))df(\tau) \quad (f(0)=0)).
\end{aligned}$$

(ii) General case : $H = \sum_{n\geq 0} H_n z^n$, we have :

$$\begin{aligned}
\mathcal{E}_a(f;H)(t) &= \sum_{n\geq 0} H_n \mathcal{E}_a(f;z^n)(t)\\
&= \sum_{n\geq 0} H_n \int_0^t \frac{(\xi_z(t)-\xi_z(\tau))^n}{n!}df(\tau) \quad \text{(case } (i))\\
&= \int_0^t h(\xi_z(t)-\xi_z(\tau))df(\tau).
\end{aligned}$$

In particular, for $z = z_0$, since $\xi_{z_0}(t) = t$, then we have result •

Corollary 2.1.2. : *Let $G \in \mathbb{K} \ll Z \gg$ be a formal power series, and let H be a formal power series on the only letter in z. The Evaluation with respect to the kernel f, for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ related to the finite alphabet Z, of the formal power series GH is :*

$$\mathcal{E}_a(GH)(t) = < G|\epsilon > h(\xi_z(t)) + \int_0^t h(\xi_z(t)-\xi_z(\tau))d\mathcal{E}_a(G)(\tau),$$

where $h(\xi_z(t))$ the Evaluation of the formal power series H. In particular, if H is a formal power series in $\mathbb{K} \ll z_0 \gg$, then :

$$\mathcal{E}_a(GH)(t) = < G|\epsilon > h(t) + \int_0^t h(t-\tau)d\mathcal{E}_a(G)(\tau).$$

Proof : Actually, since $GH = < G|\epsilon > H + G_1 H$, where G_1 is the quasiregular formal power series $I\!\!K \ll Z \gg$. Hence by the theorem 2.1.1, the Evaluation of the formal power series GH, for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ related to the finite alphabet Z, is $< G|\epsilon > h(\xi_z(t)) + \mathcal{E}_a(G_1 H)(t)$. Using the Convolution theorem in the particular case where the kernel f is the Evaluation of the quasiregular formal power series G_1, we have the expected result •

3. Calculus examples

Example 3.1.

In this example, we compute the Evaluation of the series z^{*n} for any letter z and for any integer $n \geq 0$.

Lemma 3.1.1. : *For any integer $n \geq 1$, we have*

$$\mathcal{E}_a((\alpha z)^{*n})(t) = \exp(\alpha\xi_z(t)) g_n(\alpha\xi_z(t)),$$

where the g_n are polynomials and verifie the following inductive equations :

$$g_n(\alpha\xi_z(t)) = \begin{cases} 1 & \text{if} \quad n = 1 \\ g_{n-1}(\alpha\xi_z(t)) + \alpha \int_0^t g_{n-1}(\alpha(\xi_z(t) - \xi_z(\tau))) d\xi_z(\tau) & \text{if} \quad n > 1. \end{cases}$$

Proof : Since $\mathcal{E}_a((\alpha z)^*)(t) = \exp(\alpha\xi_z(t))$, we can write $g_1(\alpha\xi_z(t)) = 1$. We suppose that the result is true for any ν, $0 \leq \nu \leq n-1$. For $\nu = n$, we have $(\alpha z)^{*n} = (\alpha z)^*(\alpha z)^{*n-1}$. By the induction hypothesis and by the corollary 2.1.2., we obtain :

$$\begin{aligned} \mathcal{E}_a((\alpha z)^{*n})(t) = \exp(\alpha\xi_z(t)) & g_{n-1}(\alpha\xi_z(t)) \\ & + \int_0^t \exp(\alpha(\xi_z(t) - \xi_z(\tau))) g_{n-1}(\alpha(\xi_z(t) - \xi_z(\tau))) d\exp(\alpha\xi_z(\tau)) \\ & = \exp(\alpha\xi_z(t)) \left[g_{n-1}(\alpha\xi_z(t)) + \alpha \int_0^t g_{n-1}(\alpha(\xi_z(t) - \xi_z(\tau))) d\xi_z(\tau) \right], \end{aligned}$$

hence we have the expected result •

Lemma 3.1.2. : *The family $g_n(\alpha\xi_z(t)) = \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\alpha\xi_z)^j(t)}{j!}$, for $n \geq 1$, is the unique solution of the inductive equations :*

$$g_n(\alpha\xi_z(t)) = \begin{cases} 1 & \text{if} \quad n = 1 \\ g_{n-1}(\alpha\xi_z(t)) + \alpha \int_0^t g_{n-1}(\alpha(\xi_z(t) - \xi_z(\tau))) d\xi_z(\tau) & \text{if} \quad n > 1. \end{cases}$$

Proof : Given $G_1 = 1 \in I\!K \ll Z \gg$, we have clearly $g_1(\alpha\xi_z(t)) = \mathcal{E}_a(G_1)(t) = 1$. Suppose that for any $n \geq 1$, we have $g_n(\alpha\xi_z(t)) = \mathcal{E}_a(G_n)(t)$, with $G_n \in I\!K \ll Z \gg$. Thus we have (see the corollary 2.1.2.) $\mathcal{E}_a(G_n) = \mathcal{E}_a(G_{n-1}) + \mathcal{E}_a(\alpha z G_{n-1})$. This equation is true if $G_n = (1+\alpha z)G_{n-1}$. In other words :

$$G_n = (1+\alpha z)^{n-1} = \sum_{j=0}^{n-1} \binom{n-1}{j} (\alpha z)^j.$$

Since $\mathcal{E}_a((\alpha z)^j)(t) = \frac{(\alpha\xi_z(t))^j}{j!}$, $j \geq 0$, then we have for any integer $n \geq 1$:

$$g_n(\alpha\xi_z(t)) = \mathcal{E}_a(G_n)(t) = \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\alpha\xi_z)^j(t)}{j!} \quad \bullet$$

By the lemma 3.1.1. and the lemma 3.1.2., we have the following proposition :

Propositon 3.1.1. : *For any positive integer n, for any complex number α, we have :*

$$\mathcal{E}_a[(\alpha z)^{*n}](t) = \begin{cases} 1 & \text{if} \quad n = 0 \\ \exp[\alpha\xi_z(t)] \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{[\alpha\xi_z(t)]^j}{j!} & \text{if} \quad n > 0. \end{cases}$$

In particular, for $z = z_0$, we have :

$$\mathcal{E}_a[(\alpha z_0)^{*n}](t) = \begin{cases} 1 & \text{if} \quad n = 0 \\ \exp(\alpha t) \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\alpha t)^j}{j!} & \text{if} \quad n > 0. \end{cases}$$

By the theorem 2.1.2. and the proposition 3.1.1., we deduce the following theorem

Theorem 3.1.1. : *For any positive integer n, for any complex number α, we have*

$$\mathcal{E}_a[f;(\alpha z)^{*n}](t) = \begin{cases} f(t) & \text{if} \quad n = 0 \\ \sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp[\alpha(\xi_z(t) - \xi_z(\tau))] \\ \qquad \frac{[\alpha(\xi_z(t) - \xi_z(\tau))]^j}{j!} df\,(\tau) & \text{if} \quad n > 0. \end{cases}$$

In particular, for $z = z_0$, we have :

$$\mathcal{E}_a[f;(\alpha z_0)^{*n}](t) = \begin{cases} f(t) & \text{if} \quad n = 0 \\ \sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp[\alpha(t-\tau)] \frac{[\alpha(t-\tau)]^j}{j!} df(\tau) & \text{if} \quad n > 0. \end{cases}$$

Example 3.2.

In this example, we compute the Evaluation of the formal power series S that verifies the following polynomial equation :

$$S + \beta_1 Sz + \beta_2 Sz^2 + \ldots \beta_n Sz^n = G,$$

where z is a letter of Z, and G a formal power series of $\mathbb{K} \ll Z \gg$. We have $SK = G$, where K is the formal power series of $\mathbb{K} \ll Z \gg$ defined by $\sum_{k=0}^{n} \beta_k z^k$ with $\beta_0 = 1$.

Since the constant term $< K|\epsilon > = \beta_0 = 1$ does not vanish, then the formal power series K^{-1} exists and is a formal power series in the single commutative variable z. Suppose that K admits r complex distinguished roots $\mu_1, \ldots, \mu_r$ of respective multiplicity order $m_1, \ldots, m_r$. One can express unically K^{-1} under partial fraction decomposition form $\sum_{l=1}^{r} \sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} H_{l,k}$, where for any $l \in [1..r]$ and for any $k \in [1..m_l]$, $\lambda_{l,k} \in \mathbb{C}$, and each $H_{l,k}$ is $\left(\frac{z}{\mu_l}\right)^{*k}$. Let $h_{l,k}(\xi_z(t))$ be the Evaluation of $H_{l,k}$:

$$h_{l,k}(\xi_z(t)) = \exp\left(\frac{\xi_z(t)}{\mu_l}\right) \sum_{j=0}^{k-1} \binom{k-1}{j} \frac{1}{j!} \left(\frac{\xi_z(t)}{\mu_l}\right)^j \qquad \text{(proposition 3.1.1.).}$$

So $S = GK^{-1}$ can be expressed as $\sum_{l=1}^{r} \sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} GH_{l,k}$. By the theorem 2.1.1. and by the corollary 2.1.2., we obtain the Evaluation of S :

$$\mathcal{E}_a(S)(t) = \sum_{l=1}^{r} \sum_{k=1}^{m_l} \frac{\lambda_{l,k}}{(-\mu_l)^k} \left[< G|\epsilon > h_{l,k}(\xi_z(t)) + \int_0^t h_{l,k}(\xi_z(t) - \xi_z(\tau)) d\mathcal{E}_a(G)(\tau) \right].$$

Let us indicate that in the particular case $z = z_0$, the above calculation corresponds mutadis mutandis to the inverse Laplace transform in the study of linear control systems.

As conclusion, we can say that if the Fliess' series is considered as a symbolic encoding of input/output behaviour of the nonlinear control systems, then the Evaluation transform allows in return to easy derive the temporal behaviour from this symbolic description, (see [6] for a simple obtaintion of the Taylor expansion of the Volterra kernel). Thus, we get a generalization of the notion of transfer function (generating series on one variable) and impulse response, encoding signals produced by linear or multilinear systems, and the Heaviside calculus (Fourier and Laplace transforms) as already pointed out by M. Fliess and al. ([2], [3], [8], [9]). In the symbolic calculus for linear control systems area, the integration operator is noted by "$\frac{1}{p}$". Here, it coincides with the letter z_0. And the letters z of the Fliess' encoding alphabet Z plays an analogous part : they encode the "Stieltjes integration operators". And we have the following Evaluations of some usual formal power series :

S	$\mathcal{E}_a(f;S)$
ϵ	$f(t)$
z^n	$\int_0^t \frac{(\xi_z(t)-\xi_z(\tau))^n}{n!} df(\tau)$
$z^{*^n}, n \geq 1$	$\sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp(\xi_z(t)-\xi_z(\tau)) \frac{(\xi_z(t)-\xi_z(\tau))^j}{j!} df(\tau)$
$\sum_{n\geq 0} c_n z^n$	$\sum_{n\geq 0} c_n \int_0^t \frac{(\xi_z(t)-\xi_z(\tau))^n}{n!} df(\tau)$
then, in particular	
$z^* = \sum_{n\geq 0} z^n$	$\int_0^t \exp(\xi_z(t)-\xi_z(\tau)) df(\tau)$
$\sum_{n\geq 0} n z^n$	$\int_0^t (\xi_z(t)-\xi_z(\tau)) \exp(\xi_z(t)-\xi_z(\tau)) df(\tau)$
$\sum_{n\geq 0} (-1)^n z^{2n}$	$\int_0^t \cos(\xi_z(t)-\xi_z(\tau)) df(\tau)$
$\sum_{n\geq 0} (-1)^n z^{2n+1}$	$\int_0^t \sin(\xi_z(t)-\xi_z(\tau)) df(\tau)$

In particular, for $f = un$, we have the following Evaluations ([5], [6], [7]) :

S	$\mathcal{E}_a(S)$
ϵ	1
z^n	$\frac{\xi_z^n(t)}{n!}$
$z^{*^n}, n \geq 1$	$\exp(\xi_z(t)) \sum_{j=0}^{n-1} \binom{n-1}{j} \frac{(\xi_z(t))^j}{j!}$
$\sum_{n\geq 0} c_n z^n$	$\sum_{n\geq 0} c_n \frac{\xi_z^n(t)}{n!}$
then, in particular	
$z^* = \sum_{n\geq 0} z^n$	$\exp(\xi_z(t))$
$\sum_{n\geq 0} n z^n$	$\xi_z(t) \exp(\xi_z(t))$
$\sum_{n\geq 0} (-1)^n z^{2n}$	$\cos(\xi_z(t))$
$\sum_{n\geq 0} (-1)^n z^{2n+1}$	$\sin(\xi_z(t))$

4. References

[1] **J. Berstel and C. Reutenauer**, *Séries rationelles et leurs langages*, Masson, Collection E.R.I., 1984.

[2] **M. Fliess**, *Fonctionnelles causales non linéaires et indéterminées non commutative*, Bull. Soc. Math France, 109, 1981, pp. 3-40.

[3] **M. Fliess, M. Lamnabhi and F. Lamnabhi-Lagarrigue**, *An algebraic approach to nonlinear functional expansions*, IEEE Trans. Circ. Syst., CAS-30, 1983, pp. 554-570.

[4] **D. Foata**, *La série génératrice exponentielle dans les problèmes d'énumération*, Séminaire de Math. Sup., Presses de L'Université de Montréal, Montréal, 1974.

[5] **V. Hoang Ngoc Minh**, *Eléments d'un calcul symbolique pour les systèmes dynamiques non linéaires*, Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles Non Commutatives", Lille, December 1988.

[6] **V. Hoang Ngoc Minh and G. Jacob**, *Symbolic calculus and Volterra series*, IFAC Symposium "Non Linear Control Systems Design", Capri, Juin 1989.

[7] **V. Hoang Ngoc Minh and G. Jacob**, *Transformation d'Evaluation et Calcul Symbolique pour les Systèmes Non Linéaires*, LIFL technical report, IT 168, 1989.

[8] **M. Lamnabhi**, *A new symbolic calculus for the reponse of nonlinear systems*, Systems & Control Letters, 1982, pp. 154-162.

[9] **M. Lamnabhi**, *Functional analysis of nonlinear circuits : a generating power series approach*, IEE proceeding, Vol. 133, Pt H, N° 5, pp. 375-384.

ANNEXE D

EVALUATION TRANSFORM AND ITS IMPLEMENTATION IN MACSYMA

("New Trends in Systems Theory", Genoa-Italy, July 1990)

Evaluation Transform and its impementation in MACSYMA

V. Hoang Ngoc Minh & G. Jacob, L.I.F.L. - U.A. 369 C.N.R.S.
Université Lille I, 59655 Villeneuve d'Ascq, France.

Abstract : Given a nonlinear control system, one can view its output function as a signal, parametrized by the primitives of the input functions. This signal can be formally described by its M. Fliess' power series, that is a formal power series on noncommutative variables. Then the temporal behaviour of the system can be derived from this generating series via the "Evaluation transform". We emphasize notion of Evaluation transform, as generalizing inverse Laplace transform to the nonlinear area. Here, we present the main properties of this Evaluation transform, and we conclude by presenting a consise implementation in the algebraic system MACSYMA, allowing a particularly quick computation.

1. Introduction and notations

The output $y(t)$ of any control system can be symbolically described by its **generating series** $S = \sum_{w \in Z^*} < S|w > w$ ([4]), that is a formal power series on non-commutative variables belonging to a finite alphabet $Z = \{z_0, z_1, \ldots, z_m\}$. By the fundamental formula of M. Fliess ([4]), (also called "Peano-Baker formula"), the output $y(t)$ is $\sum_{w \in Z^*} < S|w > \int_0^t \delta_a w$, where $\int_0^t \delta_a w$ is the "iterated integral" associated to the word w. This formula presents the output $y(t)$ as the "Evaluation" for the input $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ of the generating series S, namely : $y(t) = \mathcal{E}_a(S)(t)$ ([6], [7]). This Evaluation function $\mathcal{E}_a$ associates to any formal power series S (submitted to some convergence conditions) the "output" $\mathcal{E}_a(S)(t)$, obtained by replacing each word w in S by its Evaluation $\mathcal{E}_a(w)(t)$. Then the Evaluation of S can be viewed as a signal depending on the $m+1$ parameters $\xi_z(t) = \int_0^t a^z(\tau)d\tau,\ z \in Z$ ([6], [7]). This point of view leads naturally to develop a noncommutative "symbolic calculus", that generalizes the Heaviside calculus ([4], [5]) in the nonlinear area. So, the notions of "transfer function" (generating series on one variable) and "impulsive response", coding signals produced by linear or multilinear systems, can be generalized to generating series on $m+1$ variables and Volterra series, coding signals produced by nonlinear control systems ([4], [5], [9], [10]). And The Evaluation function $\mathcal{E}_a$ corresponds to the "inverse Laplace transform" ([6], [7]).

In continuation of [6], [7], we have introduced a "kernel" for the Evaluation function ([8]). This kernel can be viewed as the "temporal memory" of the system in the Volterra's meaning as well as in the programmation meaning. So the Evaluation transform becomes a functional depending on the kernel and on the inputs $\{a^z\}_{z \in Z}$. Here, this functional is implemented in the algebraic system MACSYMA by use of its recursive definition and of the recursive internal representation by the "binary trees" of the linear combinations of the "noncommutative rational fractions" described by $(c_0 z_{j_0})^{*p_0} z_{i_1} (c_1 z_{j_1})^{*p_1} z_{i_2} (c_2 z_{j_2})^{*p_2} \ldots z_{i_{k-1}} (c_{k-1} z_{j_{k-1}})^{*p_{k-1}} z_{i_k} (c_k z_{j_k})^{*p_k}$, where $p_0, \ldots, p_k$ are integers, $c_0, \ldots, c_k$ are complex numbers, $z_{j_0}, \ldots, z_{j_k}$ and $z_{i_0}, \ldots, z_{i_k}$ are letters.

Recall that a **formal power series on the associative variables** $z \in Z$ (non-commutative if card $Z \geq 2$) with coefficients in A ([1]), is any mapping :

$$S : Z^* \longrightarrow A, \qquad w \longmapsto < S|w >,$$

and the set of all formal power series over Z is denoted by $A \ll Z \gg$. A formal power series S will be written as a formal sum $\sum_{w \in Z^*} < S|w > w$, where $< S|w >$ is the coefficient of the word w in S. A formal power series S will be said **quasiregular** if and only if its constant term $< S|\epsilon >$ vanishes. The sum of two formal power series S, T is the formal power series $S + T$ defined by :

$$\forall w \in Z^*, \quad < S + T|w > = < S|w > + < T|w > .$$

The **Cauchy product**, noted by ".", of two formal power series S, T is the series formal power $S.T$ defined by :

$$\forall w \in Z^*, \quad < S.T|w > = \sum_{u,v \in Z^*, uv = w} < S|u >< T|v >$$

("." will be omitted when there is no ambiguity). For any quasiregular formal power series S, S^* represents classically the formal power series $\sum_{n \geq 0} S^n$. In commutative variables, it coincides with the rational fraction $1/(1 - S)$ and $S^{*n} = \big(1/(1 - S)\big)^n$.

2. Symbolic calculus

Definition 2.1. : *We will call* **input related to a finite alphabet** Z *the given of a vector* $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m} \,)$ *of piecewise continous real valued functions defined on* $[0, t] \subset I\!R_+$. *Conventionally the 0-component of any input is* $a^{z_0} \equiv 1$.

We will call **path associated to the input** $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m} \,)$, *the time dependent vector* $\xi = {}^t(\, \xi_{z_0} \quad \xi_{z_1} \quad \ldots \quad \xi_{z_m} \,)$ *(K.T. Chen, [2]), defined as follows :*

$$\forall z \in Z, \quad \xi_z(\tau) = \int_0^\tau d\xi_z(\rho) = \int_0^\tau a^z(\rho) d\rho.$$

Thus we have $\xi_{z_0}(\tau) = \displaystyle\int_0^\tau d\rho = \tau$, *and for any* $i \in [0..m]$, $\xi_z(0) = 0$.

In all the sequel, we suppose that the function f is Stieltjes integrable with respect to $\{\xi_z\}_{z \in Z}$ defined over $[0, t], (t \geq 0)$ and f vanishes at zero. The unit step (vanishing at zero) is noted "un" : $\forall \tau \in]0, t], \quad un(\tau) = 1, \quad un(0) = 0$.

Definition 2.2. : *The* **Evaluation of the word** w **with respect to the kernel** f, *for the input* $a = (\, a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m} \,)$ *related to a finite alphabet* Z, *is defined by induction on the length of* w *as follows :*

$$\mathcal{E}_a(f; w)(t) = \begin{cases} f(t) & \text{if} \quad w = \epsilon, \\ \displaystyle\int_0^t \mathcal{E}_a(f; v)(\tau) d\xi_z(\tau) & \text{if} \quad w = vz. \end{cases}$$

Definition 2.3. : *We will call* **Evaluation of the formal power series** S **with respect to the kernel** f, *for the input* $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ *related to a finite alphabet* Z, *when it is defined, the functional :*

$$\mathcal{E}_a(f;S) = \sum_{w \in Z^*} < S|w > \mathcal{E}_a(f;w).$$

In particular, for $f = un$, *the* **Evaluation of the formal power series** S, *for the input* $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ *related to a finite alphabet* Z, *is the function* ([6], [7]) :

$$\mathcal{E}_a(S) = \mathcal{E}_a(un;S).$$

We have the following fundamental results (the proofs can be found in [8]) :

(P1) *Let* S, T *be two formal power series. Then* $\mathcal{E}_a(f;S.T) = \mathcal{E}_a(\mathcal{E}_a(f;S);T)$.

(P2) *Let* S, T *be two formal power series. Let* r *be a scalar. Then :*

$$\mathcal{E}_a(f;S + r.T) = \mathcal{E}_a(f;S) + r.\mathcal{E}_a(f;T).$$

(P3) *Let* z *be a letter in* Z. *Let* H *be the formal power series on the only letter* z *and* $h(\xi_z(t))$ *its Evaluation. The Evaluation with respect to the kernel* f, *for the input* $a = (a^{z_0} \quad a^{z_1} \quad \ldots \quad a^{z_m})$ *related to a finite alphabet* Z, *of* H *is :*

$$\mathcal{E}_a(f;H)(t) = \int_0^t h(\xi_z(t) - \xi_z(\tau))df(\tau).$$

(P4) *For any positive integer* n, *for any complex number* α, *we have :*

$$\mathcal{E}_a[f;(\alpha z)^{*n}](t) = \begin{cases} f(t) & \text{if} \quad n = 0 \\ \displaystyle\sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp[\alpha(\xi_z(t) - \xi_z(\tau))] \frac{[\alpha(\xi_z(t) - \xi_z(\tau))]^j}{j!} df(\tau) & \\ & \text{if} \quad n > 0. \end{cases}$$

According to the M. Fliess' fundamental formula, the Evaluation of a formal power series S can be viewed as a transform that associates to S the signal depending on the primitives $\{\xi_z\}_{z \in Z}$ of the inputs functions, and the Evaluation transform is nothing else as a generalization of inverse Laplace and Fourier transforms ([6], [7]). Hence, the M. Fliess' series, considered as a symbolic encoding of input/output behaviour of the nonlinear control systems, the Evaluation transform allows in return to easily derive the temporal behaviour from this symbolic description (see the Annex A for the Evaluation of some usual formal power series and see [7] for a simple computation of the Taylor expansion of the Volterra kernel). Thus, we get a generalization of the notion of "transfer function" (generating series on one variable) and "impulse response", encoding signals produced by linear or multilinear systems, and the Heaviside calculus (Fourier and Laplace transforms) as already pointed out by M. Fliess and al. ([4], [5], [9],

[10]). In the symbolic calculus for linear control system area, the integration operator is noted by "$\frac{1}{p}$". Here, it coincides with the letter z_0. And the letters z of M. Fliess' encoding alphabet Z plays an analogous part : they encode the "Stieltjes integration operators". By (P3), we deduce a correspondence between certain convolutions of signals and Cauchy products of a formal power series $G \in A \ll Z \gg$ and a formal power series H on the only one letter z :

$$\mathcal{E}_a(GH)(t) = < G|\epsilon > h(\xi_z(t)) + \int_0^t h(\xi_z(t) - \xi_z(\tau)) d\mathcal{E}_a(G)(\tau).$$

Note that there is some dissymetry this convolution formula and not at all in the linear area, it is a consequence of the noncommutative nature of the Cauchy product of G and H. The introduction of the kernel f for the Evaluation function $\mathcal{E}_a$ ([6], [7]) allows to give a notion of "memory" for the system. This kernel can be viewed as the "temporal memory" of the system in the Volterra's meaning as well as in the programmation meaning, that justifies our approach. So the Evaluation transform becomes a functional depending on the kernel f and on the inputs $\{a^z\}_{z\in Z}$. In 3., we give an implementation of these functionals in MACSYMA, by using the "λ−notations". The λ−notation is used for unnamed functions to indicate the correspondence between the variables of the functions and the arguments that are to be substituted for them. This is elegant and useful for passing functional arguments to the other functions.

3. One implementation in MACSYMA

3.1. A Choice of the internal representation

In all the sequence, we will consider the formal power series which is linear combinations of the "noncommutative rational fractions" described by :

$$(c_0 z_{j_0})^{*p_0} z_{i_1} (c_1 z_{j_1})^{*p_1} z_{i_2} (c_2 z_{j_2})^{*p_2} .. z_{i_{k-1}} (c_{k-1} z_{j_{k-1}})^{*p_{k-1}} z_{i_k} (c_k z_{j_k})^{*p_k},$$

where $p_0, \ldots, p_k$ are integers, $c_0, \ldots, c_k$ are complex numbers, $z_{j_0}, \ldots, z_{j_k}$ and $z_{i_0}, \ldots, z_{i_k}$ are letters in Z.

For the internal representation, such a linear combination of noncommutative rational fractions can be an element of A, or a letter of Z, or an element of the form $S_1 \ \phi \ S_2$, where ϕ is one of the "binary operators" in $\{*, +, ., \wedge\wedge, \wedge*\}$:

. For $\phi =$ "$*$" then S_1 is an element in A, S_2 is a formal power series (multiplication by a scalar case).

. For $\phi =$ "$+$" then S_1, S_2 are the formal power series (addition of two formal power series case).

. For $\phi =$ "." then S_1, S_2 are formal power series (Cauchy product case).

. For $\phi =$ "$\wedge\wedge$" then S_2 is an integer, S_1 is a formal power series (noncommutative exponentiation case).

. For $\phi =$ "$\wedge *$" then S_2 is an integer, S_1 is a formal power series of the form αz, where α is an element in A and z is a letter (noncommutative rational fractions case).

Such a linear combination of noncommutative rational fractions can be represented by a "binary tree". The analysis of this tree, as of its sub-trees, can be described by the functions **Root** (For the "root"), **Left** and **Right** (For the two "operants"). The implementation in MACSYMA of these functions is given in the Annex B.

3.2. Presentation of the Evaluation algorithm

The implementation of the Evaluation fonctional $\mathcal{E}_a(f;.)$ is given as a recursive definition, following the recursive structure of our non commutative rational fractions, and the recursive properties of the Evaluations obtained by (P1) to (P4) :

if S is a real **then** multiplication the Evaluation of the empty word by a scalar,
see the definition 2.1.2. and P2.
else if S is a letter **then** Stieltjes integral of f with respect to the primitive
indexed by S, see the definition 2.1.2.
else if Root(S)="*"**then** multiplication by a scalar case, see P2
else if Root(S)="+"**then** addition case, see P2.
else if Root(S)="."**then** Cauchy product case, see P1
else if Root(S)="^^"**then** noncommutative exponential case,
if Left(S) is a letter **then** letter repetition
case, see P3
else series repetition case, see P1
else if Root(S)="^*"**then** rational fraction case, see P4
else no more case able to treat.

In the Annex B, we give an implementation of the following MACSYMA functions

. "**Lamb-exp(expr)**" takes as argument an explicit expression **expr** of some function on t, and returns as value the $\lambda-$notation of the function defined by **expr**.

. "**Somme(f1,f2)**" takes as arguments two $\lambda-$notations **f1** and **f2**, and returns as value the $\lambda-$notation of the function that associates to t the sum "f1(t)+f2(t)".

. "**Produit(r,f)**" takes as arguments an real **r** and a $\lambda-$notation **f**, and returns as value the $\lambda-$notation of the function that associates to t the product "r$*$f(t)".

. "**Stieltjes(f,z)**" takes as arguments a $\lambda-$notation **f** and a letter **z**, and returns as value the $\lambda-$notation of the function that associates to t the Stieltjes integral $\int_0^t f(\tau)d\xi_z(\tau)$.

. "**Conv(f,z,n)**" takes as arguments a $\lambda-$notation **f**, a letter **z** and an integer **n**, and returns as value the $\lambda-$notation of the function that associates to t the Evaluation $\mathcal{E}_a(\mathbf{f};\mathbf{z}^{\mathbf{n}})(t)$ of the word $\mathbf{z}^{\mathbf{n}}$. Its implementation follows (P3)

. "**Star-Jacob(f,S,n)**" takes as arguments a $\lambda-$notation **f**, the formal power series **S** (on a single letter) and an integer **n**, and returns as value the $\lambda-$notation of the function associates to t the Evaluation $\mathcal{E}_a(\mathbf{f};\mathbf{S}^{*\mathbf{n}})(t)$ of the noncommutative rational fraction $\mathbf{S}^{*\mathbf{n}}$. Its implementation follows (P4).

. "**Jacob(f,S)**" takes as arguments a $\lambda-$notation **f** and a linear combination of the noncommutative rational fractions **S**, and returns as value the $\lambda-$notation of the function that associates to t the Evaluation $\mathcal{E}_a(\mathbf{f};\mathbf{S})(t)$.

3.3. Example

We will compute the Volterra approximation of the "output function" associated to the formal power series $S=(z_0+z)^*$. One can express S as $S=z_0^*(z.z_0^*)^* = \sum_{k=0}^{K} z_0^*.(z.z_0^*)^k + \sum_{k\geq K+1} z_0^*.(z.z_0^*)^k$ ([7]). Hence the Volterra approximation, up to the order K, of the "output" associated to S is the Evaluation of the $K+1$ first terms of the aboved expression : $\mathcal{E}_s((z_0+z)^*) = \sum_{k=0}^{K} \mathcal{E}_s(z_0^*.(z.z_0^*)^k) + \ldots$

Let z_1, z_2, z_3 encode the following "input functions" t, $\exp(t)$, $\sin(t)$ respectively. Using the function "Volterra(K,z)" given in the Annex B, we have for $K = 5$:

```
(c40) Volterra(5,z1);
           10        8        6         4           2              t
       (t    + 10 t  + 80 t  + 480 t  + 1920 t  + 3840) %e
(d40) ---------------------------------------------------------
                                   3840
(c45) Volterra(5,z2);
              t     5 t        3 t         2 t         t
            %e  (%e    + 10 %e    + 20 %e    + 45 %e  + 44)
(d45)       -------------------------------------------------
                                     120
(c50) Volterra(5,z3);
                   t
(d50)          - %e  (cos(5 t) - 20 cos(4 t) + 205 cos(3 t)
                 - 1360 cos(2 t) + 5810 cos(t) - 6556)/1920
```

4. References

[1] **J. Berstel and C. Reutenauer**, *Rational Series and their languages*, Springer-Verlag, 1988.

[2] **K.T. Chen**, *Iterated path integrals*, Bull. Amer. Math. Soc., vol 83, 1977, pp. 831-879.

[3] **J. Davenport, Y. Siret and E. Tournier**, *Calcul Formel. Systèmes et algorithmes de manipulations algébriques*, Masson, Collection E.R.I, 1987.

[4] **M. Fliess**, *Fonctionnelles causales non linéaires et indéterminées noncommutatives*, Bull. Soc. Math France, 109, 1981, pp. 3-40.

[5] **M. Fliess, M. Lamnabhi and F. Lamnabhi-Lagarrigue**, *An algebraic approach to nonlinear functional expansions*, IEEE Trans. Circ. Syst., CAS-30, 1983, pp. 554-570.

[6] **V. Hoang Ngoc Minh**, *Eléments d'un calcul symbolique pour les systèmes dynamiques non linéaires*, Journées-Séminaire "Traitements Algébriques et Informatiques des Séries Formelles Non Commutatives", Lille, December 1988.

[7] **V. Hoang Ngoc Minh and G. Jacob**, *Symbolic calculus and Volterra series*, IFAC Symposium "Non Linear Control Systems Design", Capri, Juin 1989.

[8] **V. Hoang Ngoc Minh and G. Jacob**, *Transformation d'Evaluation et Calcul Symbolique pour les Systèmes Non Linéaires*, LIFL technical report, IT 168, 1989.

[9] **M. Lamnabhi**, *A new symbolic calculus for the response of nonlinear systems*, Systems & Control Letters, 1982, pp. 154-162.

[10] **M. Lamnabhi**, *Functional analysis of nonlinear circuits : a generating power series approach*, IEE proceeding, Vol. 133, Pt H, N° 5, 1986, pp. 375-384.

Annex A

We have the following Evaluations of some usual formal power series :

S	$\mathcal{E}_a(f;S)$
ϵ	$f(t)$
z^n	$\int_0^t \frac{(\xi_z(t)-\xi_z(\tau))^n}{n!}df(\tau)$
$z^{*^n}, n \geq 1$	$\sum_{j=0}^{n-1}\binom{n-1}{j}\int_0^t \exp(\xi_z(t)-\xi_z(\tau))\frac{(\xi_z(t)-\xi_z(\tau))^j}{j!}df(\tau)$
$\sum_{n\geq 0} c_n z^n$	$\sum_{n\geq 0} c_n \int_0^t \frac{(\xi_z(t)-\xi_z(\tau))^n}{n!}df(\tau)$
then, in particular	
$z^* = \sum_{n\geq 0} z^n$	$\int_0^t \exp(\xi_z(t)-\xi_z(\tau))df(\tau)$
$\sum_{n\geq 0} nz^n$	$\int_0^t (\xi_z(t)-\xi_z(\tau))\exp(\xi_z(t)-\xi_z(\tau))df(\tau)$
$\sum_{n\geq 0}(-1)^n z^{2n}$	$\int_0^t \cos(\xi_z(t)-\xi_z(\tau))df(\tau)$
$\sum_{n\geq 0}(-1)^n z^{2n+1}$	$\int_0^t \sin(\xi_z(t)-\xi_z(\tau))df(\tau)$

In particular, for $f = un$, we have the following Evaluations ([6], [7]) :

S	$\mathcal{E}_a(S)$
ϵ	1
z^n	$\frac{\xi_z^n(t)}{n!}$
$z^{*^n}, n \geq 1$	$\exp(\xi_z(t))\sum_{j=0}^{n-1}\binom{n-1}{j}\frac{(\xi_z(t))^j}{j!}$
$\sum_{n\geq 0} c_n z^n$	$\sum_{n\geq 0} c_n \frac{\xi_z^n(t)}{n!}$
then, in particular	
$z^* = \sum_{n\geq 0} z^n$	$\exp(\xi_z(t))$
$\sum_{n\geq 0} nz^n$	$\xi_z(t)\exp(\xi_z(t))$
$\sum_{n\geq 0}(-1)^n z^{2n}$	$\cos(\xi_z(t))$
$\sum_{n\geq 0}(-1)^n z^{2n+1}$	$\sin(\xi_z(t))$

Annex B

```
Root(S):=inpart(S,0)$

Left(S):=inpart(S,1)$

Right(S):=if inpart(S,0)="." or inpart(S,0)="+" or inpart(S,0)="*"
          then inpart(S,allbut(1)) else inpart(S,2)$

Lamb_exp(f):=block(i:i+1,l[i](t):=factor(ratsimp(f)),l[i])$

Somme:lambda([f1,f2],Lamb_exp(f1(t)+f2(t)))$

Produit:lambda([r,f],Lamb_exp(r*f(t)))$

Stieltjes:lambda([f,z],Lamb_exp(integrate(f(tau)*a[z](tau),tau,0,t)))$

Conv:lambda([f,z,n],Lamb_exp(
if (f(t)=1) then Xi[z](t)^n/n!
else integrate(f(tau)*((Xi[z](t)-Xi[z](tau))^(n-1)/(n-1)!)*a[z](tau),tau,0,t)))$

Star_Jacob:lambda([f,S,n],Lamb_exp(
if (f(t)=1) then %e^(Jacob(un,S)(t))*sum(binomial(n-1,j)
                                  *(Jacob(un,S)(t))^j/j!,j,0,n-1)
else sum(binomial(n-1,j)*integrate(diff(f(tau),tau)
                  *%e^(Jacob(un,S)(t)-Jacob(un,S)(tau))
                  *(Jacob(un,S)(t)-Jacob(un,S)(tau))^j/j!,tau,0,t),j,0,n-1)))$

Jacob:lambda([f,S],
if numberp(S) then Produit(S,f)
else if atom(S) then Stieltjes(f,S)
   else if Root(S)="*" then Produit(Left(S),Jacob(f,Right(S)))
      else if Root(S)="+" then Somme(Jacob(f,Left(S),t),Jacob(f,Right(S)))
         else if Root(S)="." then Jacob(Jacob(f,Left(S)),Right(S))
            else if Root(S)="^^" then if atom(Left(S)) then
                                          Conv(f,Left(S),Right(S))
                                 else Jacob(Jacob(f,Left(S)),
                                          Left(S)^^(Right(S)-1))
              else if Root(S)="^*" then Star_Jacob(f,Left(S),Right(S))
                 else print("Error in S :", S))$

Volterra(K,z):=block(
v[K,z]:=Jacob(v[K-1,z],z.z0^*1),
v[0,z]:lambda([t],%e^t),
y[K,z]:=Somme(y[K-1,z],v[K,z]),
y[0,z]:v[0,z],
y[K,z](t))$
```

ANNEXE E

COMPORTEMENT ENTREE/SORTIE DES SYSTEMES ANALYTIQUES NON LINEAIRES : APPROXIMATIONS RATIONNELLES, APPROXIMATIONS STRUCTURELLES NILPOTENTES

("Analysis of Contolled Dynamical Systems",
Lyon, July 1990)

Comportement Entrée/Sortie des systèmes analytiques non linéaires : approximations rationnelles, approximations structurelles nilpotentes

Hoang Ngoc Minh, G. Jacob et N.E. Oussous

Laboratoire d'Informatique Fondamentale de Lille
U.A. 369 du C.N.R.S., Université de Lille I,
59655 Villeneuve d'Ascq Cedex. FRANCE.

1 Mars 1990

Résumé : Nous présentons deux techniques de calcul du comportement Entrée/Sortie d'un système dynamique non linéaire. La première exploite les expressions rationnelles non commutatives, la seconde repose sur un théorème de factorisation relatif à l'algèbre de Lie libre et au produit de mélange. On donne les grandes lignes d'une implantation en Macsyma.

Abstract : We present here two computing methods of the Input/Output behaviour of analytical nonlinear systems. The first uses noncommutative rational expressions, the second is based on the factorization theorem related to free Lie algebra and shuffle product. We give the sketch of an implementation in Macsyma.

1 Introduction

Les séries formelles non commutatives sur un alphabet fini $X = \{x_0, x_1, \ldots, x_m\}$ ont été introduites par M.Fliess [3] pour coder le comportement entrée/sortie des systèmes analytiques non linéaires affines en la commande, de la forme :

$$(\Sigma) \quad \begin{cases} \dot{q}(t) &= \sum_{x \in X} Y_x(q) a^x(t), \\ y(t) &= h(q(t)). \end{cases}$$

Ce codage s'est révélé un outil fécond pour aborder des problèmes aussi divers que la réalisation analytique locale minimale ([4], [16,21], [22]), le découplage et la commande interactive ([1], [10]), l'immersion dans un système linéaire ([1]).

Nous proposons ici de présenter deux techniques de calcul, exact ou approché, du comportement entrée/sortie d'un tel système (Σ). La première est basée sur le calcul des *expressions rationnelles non commutatives* sur le monoïde libre, la seconde repose sur un *théorème de factorisation*, relatif à l'algèbre de Lie libre et au produit de mélange. Cette deuxième technique a d'ailleurs déjà été proposée par Huillet, Monin et Salut dans [23,24,25]. Nous la développons ici, *pour l'implémentation*, en utilisant la factorisation de Lyndon, (une version plus générale concernant d'autres factorisations est en préparation).

Nos techniques sont à rapprocher, pour les distinguer, des *algorithmes de calcul symbolique* présentés par Fliess, Lamnabhi et Lamnabhi-Lagarrigue [5]. Ceux-ci en effet présentent comme données de leurs algorithmes les développements en série entière des entrées $\{a^x\}_{x\in X}$, ou mieux leurs transformées de Laplace-Borel, et donnent en résultat la transformée de Laplace-Borel de la sortie. Elles demandent une première phase de calcul algébrique non commutatif (en LISP), un transfert de fichier de données, et une seconde phase, pour le calcul des transformées Laplace-Borel inverse.

Dans ce travail, en continuité avec nos travaux sur la *transformation d'Evaluation* ([9,11, 12,13]), nous cherchons à préserver le plus longtemps possible le paramétrage par les entrées. Notre algorithme permet alors tout aussi bien de fournir *l'expression paramétrée de la sortie*, que de calculer explicitement la sortie pour une entrée donnée, en utilisant toute la capacité d'intégration formelle du système de calcul algébrique utilisé (actuellement MACSYMA).

2 Transformation d'Evaluation

Soit donc $X = \{x_0, x_1, \ldots, x_m\}$ un alphabet "de codage" en bijection avec les entrées $\{a^x\}_{x\in X}$ (l'entrée a^{x_0} est constante et égale 1).

Toute série formelle S (*vérifiant une certaine "condition de convergence"*, voir [3], [29]) est le codage d'une fonctionnelle entrée/sortie, appelée aussi *"opérateur de Fliess"*([29]). L'image d'une entrée a par cette fonctionnelle n'est autre que l'image de S par la transformation d'Evaluation ([9,11]) :

$$\mathcal{E}_a(S)(t) = \sum_{w\in X^*} < S|w > \mathcal{E}_a(w)(t) = \sum_{w\in X^*} < S|w > \int_0^t \delta_a w \tag{1}$$

On l'obtient donc en remplaçant, dans l'expression de S, chaque mot w par son intégrale itérée $\int_0^t \delta_a w$.

Ainsi la sortie y du système (Σ), donnée par la formule fondamentale de Fliess :

$$y(t) = \sum_{w\in X^*} \mathcal{Y}(w) \circ h_{|_{q(0)}} \mathcal{E}_a(w), \tag{2}$$

où pour tout mot w de X^*, $\mathcal{Y}(w)$ représente l'opérateur différentiel :

$$\mathcal{Y}(w) = \begin{cases} \textit{l'identité} & si \quad w = \varepsilon, \\ Y_x \circ \mathcal{Y}(v) & si \quad w = xv \end{cases} \tag{3}$$

est donc exactement l'Evaluation de la série génératrice $\sum_{w\in X^*} \mathcal{Y}(w) \circ h_{|_{q(0)}} w$ du système (Σ).

L'Evaluation d'une série S doit être considérée comme un *signal paramétré par les primitives des entrées*, $\xi_x(t) = \int_0^t a^x(\tau)d\tau$, où $x \in X$. En outre, nous définissons l'Evaluation d'une série S par rapport à un *noyau* f (qui est une fonction Stieltjes intégrable par rapport aux $\{\xi_x\}_{x\in X}$ et s'annulant en 0). Ces fonctions noyaux jouent le rôle d'une "mémoire temporelle" au cours de l'exécution récursive du programme ([13]), et correspondent aussi à une Evaluation d'une série génératrice avec un "passé". Plus précisément, la transformation d'Evaluation avec noyau est définie comme suit :

Definition 2.1 *Nous appellons Evaluation de la série formelle S par rapport au noyau f, pour l'entrée a, quand elle est définie, la fonction (de t et des $\xi_x(t)$) :*

$$\mathcal{E}_a(f; S) = \sum_{w\in X^*} < S|w > \mathcal{E}_a(f; w), \tag{4}$$

où $\mathcal{E}_a(f;w)$ est l'Evaluation du mot w par rapport au noyau f, pour l'entrée a, définie par récurrence sur la longueur de w comme suit :

$$\mathcal{E}_a(f;w)(t) = \begin{cases} f(\xi(t)) & si \quad w = \varepsilon, \\ \int_0^t \mathcal{E}_a(f;v)(\tau)d\xi_x(\tau) & si \quad w = vx. \end{cases} \tag{5}$$

L'Evaluation de la série formelle S *(sans noyau)* pour l'entrée a est donc aussi l'Evaluation de S par rapport au noyau échelon unité :

$$\mathcal{E}_a(S) = \mathcal{E}_a(un;S). \tag{6}$$

On démontre alors les propriétés fondamentales suivantes ([12]) :

(P1) *Soient S,T deux séries formelles. Alors $\mathcal{E}_a(f;S.T) = \mathcal{E}_a(\mathcal{E}_a(f;S);T)$.*

(P2) *Soient S,T deux séries formelles. Soit r un scalaire. Alors :*

$$\mathcal{E}_a(f;S+rT) = \mathcal{E}_a(f;S) + r\mathcal{E}_a(f;T).$$

(P3) *Soient S,T deux séries formelles. Alors $\mathcal{E}_a(S \sqcup\!\sqcup T) = \mathcal{E}_a(S)\mathcal{E}_a(T)$.*

(P4) *Soit x une lettre de X. Soit H une série formelle en une seule lettre x et $h(\xi_x(t))$ son Evaluation. L'Evaluation de H par rapport au noyau f, pour l'entrée a relative à l'alphabet fini X, est :*

$$\mathcal{E}_a(f;H)(t) = \int_0^t h(\xi_x(t) - \xi_x(\tau))df(\xi(\tau)).$$

(P5) *Pour tout entier $n > 0$ et pour tout nombre complexe α, on a :*

$$\mathcal{E}_a[f;(\alpha x)^{*n}](t) = \sum_{j=0}^{n-1} \binom{n-1}{j} \int_0^t \exp[\alpha(\xi_x(t) - \xi_x(\tau))]\frac{\alpha^j[\xi_x(t) - \xi_x(\tau)]^j}{j!}df(\xi(\tau)),$$

où l'on a posé $(\alpha x)^{*n} = (1 - \alpha x)^{-n}$.

3 Approximations et expressions rationnelles

Soit G_i, pour $i = 0,..,k$, une série sur une seule lettre x_{j_i}, et soit $g_i[\xi_{x_{j_i}}(t)]$ son Evaluation, pour l'entrée a relative à l'alphabet fini X. D'après les propriétés **(P1)** à **(P5)** l'Evaluation de la série $S = G_0 x_{i_1} G_1 \ldots x_{i_k} G_k$ est ([9]) :

$$\int_0^t \int_0^{\tau_k} \cdots \int_0^{\tau_2} g_0[\xi_{x_{j_0}}(\tau_1)] g_1[\xi_{x_{j_1}}(\tau_2) - \xi_{x_{j_1}}(\tau_1)] \ldots g_k[\xi_{x_{j_k}}(t) - \xi_{x_{j_k}}(\tau_k)] d\xi_{x_{i_1}}(\tau_1) \ldots d\xi_{x_{i_k}}(\tau_k). \tag{7}$$

Cette formule peut se généraliser à des séries G_i échangeables sur plusieurs lettres. En effet, si G_i est échangeable, g_i est une fonction analytique des $\{\xi_x\}_{x \in X}$, dont G_i est la transformée de Laplace-Borel multidimensionnelle ([5]), ce qui s'écrit encore :

$$g_i(\xi) = g_i(t, \xi_{x_1}, \ldots, \xi_{x_m}) = \mathcal{E}_a(G_i). \tag{8}$$

L'Evaluation de $S = G_0 x_{i_1} G_1 \ldots x_{i_k} G_k$ est alors donnée par :

$$\int_0^t \int_0^{\tau_k} \cdots \int_0^{\tau_2} g_0[\xi(\tau_1)] g_1[\xi(\tau_2) - \xi(\tau_1)] \ldots g_k[\xi(t) - \xi(\tau_k)] d\xi_{x_{i_1}}(\tau_1) \ldots d\xi_{x_{i_k}}(\tau_k).$$

Ceci s'applique en particulier aux séries échangeables de la forme $G_i = \left(\sum_{x\in X} \alpha_x x\right)^*$ (les α_x sont des nombres complexes), dont l'Evaluation est $\exp\left(\sum_{x\in X} \alpha_x \xi_x(t)\right)$. Tout ceci s'intègre très simplement au logiciel d'Evaluation déjà présenté en [13].

Cependant, nous ne savons pas Evaluer, *en gardant le paramétrage des entrées*, des expressions rationnelles aussi simples que $(x_0x_1)^*$, ni en général les approximants de Padé non commutatifs introduits en ([6,7,8], [28]). Il semble qu'il y ait là une limitation intrinsèque : les "factorisations rationnelles" (basée sur le produit de Cauchy) sont mal adaptées à l'Evaluation. Il convient donc de chercher un autre type de factorisation, basée sur l'algèbre de Lie et le produit de mélange.

4 Approximations structurelles nilpotentes

La formule fondamentale de M.Fliess (2), met en évidence l'*opérateur différentiel* :

$$H = \sum_{w\in X^*} \mathcal{E}_a(w)\mathcal{Y}(w), \tag{9}$$

appelé aussi *opérateur de transport* dans [23,24,25] (voir également [26]), qui représente en un certain sens la structure du système. On peut aussi décrire cet opérateur comme l'image par la transformation $\mathcal{E}_a \otimes \mathcal{Y}$ de la série formelle "double" :

$$\sum_{w\in X^*} w \otimes w. \tag{10}$$

Rappelons que $\mathcal{Y}$ est un morphisme de monoïde pour le produit de Cauchy (ou de "concaténation"), et que $\mathcal{E}_a$ est un morphisme pour le produit de mélange.

Or on peut obtenir une factorisation de cette série double grâce au théorème de PBW :

4.1 Base de Poincaré-Birkhoff-Witt

Théorème 4.1 PBW.-*Soit $\mathcal{L}$ une algèbre de Lie et soit $\mathcal{B} = (P_i)_{i\geq 1}$ une base totalement ordonnée de $\mathcal{L}$. Alors $\{P_{j_1}^{i_1} P_{j_2}^{i_2} \cdots P_{j_n}^{i_n} \mid i_1, i_2, \ldots, i_n \geq 0,\ n \geq 0 \text{ et } P_{j_1} > P_{j_2} > \cdots > P_{j_n}\}$ est une base de l'algèbre enveloppante de $\mathcal{L}$ appelée base de Poincaré-Birkhoff-Witt, et notée* PBW.$\mathcal{B}$.

Notons $\mathcal{L}ie\langle X\rangle$ l'algèbre de Lie libre sur X. Son algèbre enveloppante est l'algèbre $K\langle X\rangle$ des polynômes non commutatifs sur X. Soit donc $\mathcal{B}$ une base de $\mathcal{L}ie\langle X\rangle$, et PBW.$\mathcal{B}$ la base de $K\langle X\rangle$ qui lui est associée par le théorème précédent.

4.2 Base duale de la base PBW.$\mathcal{B}$

L'espace vectoriel des séries formelles est naturellement isomorphe au dual de l'espace vectoriel des polynômes non commutatifs. Cette dualité s'exprime par le produit scalaire suivant :

$$\begin{array}{ccc} K\langle\langle X\rangle\rangle \times K\langle X\rangle & \longrightarrow & K \\ (S,R) & \longmapsto & \langle S|R\rangle = \sum_{w\in X^*} \langle S|w\rangle\langle R|w\rangle \end{array}$$

A la base PBW.B de $K\langle X\rangle$, on peut associer la "base duale" $\{S_Q\}_{Q\in PBW.B}$. Les séries S_Q sont définies par $\langle S_Q|Q'\rangle = \delta_Q^{Q'}$, pour $Q,Q' \in$ PBW.B Elles vérifient :

$$\text{pour tout } R \in K\langle X\rangle, \qquad R = \sum_{Q\in PBW.B} \langle S_Q|R\rangle\, Q. \tag{11}$$

En particularisant aux mots $w \in X^*$, on peut alors écrire :

$$\begin{aligned}\sum_{w\in X^*} w\otimes w &= \sum_{w\in X^*} w\otimes\Big(\sum_{Q\in PBW.B} \langle S_Q|w\rangle\, Q\Big)\\ &= \sum_{Q\in PBW.B}\Big(\sum_{w\in X^*} \langle S_Q|w\rangle\, w\Big)\otimes Q = \sum_{Q\in PBW.B} S_Q\otimes Q\end{aligned} \tag{12}$$

4.3 Factorisation

On définit un produit sur $K\langle\langle X\rangle\rangle \otimes K\langle X\rangle$, en posant ([15,18]) :

$$(S_1\otimes P_1)(S_2\otimes P_2) = S_1 ш S_2\otimes P_1P_2.$$

La factorisation annoncée repose sur le lemme suivant ([18]) :

Lemme 4.1 *Soit $Q = P_{j_1}^{i_1}P_{j_2}^{i_2}\cdots P_{j_k}^{i_k}$ un polynôme de la base* PBW.B. *On a alors :*

$$S_Q = \frac{1}{i_1!i_2!\cdots i_k!} S_{P_{j_1}}^{ш i_1} ш S_{P_{j_2}}^{ш i_2} ш \cdots ш S_{P_{j_k}}^{ш i_k} \tag{13}$$

On en déduit alors le théorème suivant ([18]) :

Théorème 4.2 (de factorisation)

$$\sum_{w\in X^*} w\otimes w = \prod_{P\in\mathcal{B}} \exp(S_P\otimes P) \qquad \textit{(produit ordonné décroissant)}. \tag{14}$$

On a en effet,

$$\begin{aligned}\sum_{w\in X^*} w\otimes w &= \sum_{Q\in PBW.B} S_Q\otimes Q\\ &= \sum_{\substack{P_{j_1}>P_{j_2}>\cdots>P_{j_k}\\ i_1,i_2,\ldots,i_k\geq 1}} \frac{1}{i_1!i_2!\cdots i_k!} S_{P_{j_1}}^{ш i_1} ш S_{P_{j_2}}^{ш i_2} ш \cdots ш S_{P_{j_k}}^{ш i_k}\otimes P_{j_1}^{i_1}P_{j_2}^{i_2}\cdots P_{j_k}^{i_k}\\ &= \prod_{P\in\mathcal{B}}\Big(\sum_{i\geq 0}\frac{1}{i!} S_P^{ш i}\otimes P^i\Big) = \prod_{P\in\mathcal{B}} \exp(S_P\otimes P).\end{aligned} \tag{15}$$

Nous verrons que le calcul de cette factorisation est grandement simplifié si $\mathcal{B}$ est la base de Lyndon, munie de l'ordre lexicographique.

5 Factorisation de Lyndon

5.1 Mots de Lyndon et bases associées

5.1.1 Ordre lexicographique et classes de conjugaison.

Soit $(X = \{x_0, x_1, ..., x_m\}, <)$ un alphabet totalement ordonné. Un mot de longueur n est une suite $w = x_{i_1}x_{i_2}...x_{i_n}$ de n lettres de X. On notera $|w|$ la longueur du mot w. On appelle ordre lexicographique l'ordre total sur X^* défini comme suit : $\forall\, u,v \in X^*$, $u < v$ si et seulement si

$$\begin{cases} & (i) \quad \exists w \neq \varepsilon \text{ tel que } uw = v, \\ \text{ou} & \\ & (ii) \quad \exists x,y,z \in X^* \text{ et } a,b \in X \text{ tels que } u = xay, v = xbz \text{ et } a < b. \end{cases}$$

Pour cet ordre, on a ([17]) les deux propriétés suivantes :

$$\begin{cases} (1) \quad \forall w \in X^*, & u < v \iff wu < wv. \\ (2) \quad \text{Si } v \notin uX^*, \forall w,z \in X^*, & u < v \implies uw < vz. \end{cases}$$

On dit qu'un mot u est un *facteur* du mot v s'il existe $x,y \in X^*$ tels que $v = xuy$. Si $x = \varepsilon$ (resp. $y = \varepsilon$), on dira que u est le *facteur gauche* (resp. *droit*) de v, *propre* si $y \neq \varepsilon$ (resp. $x \neq \varepsilon$). Deux mots u et v de X^* sont dits *conjugués* si il existe $x,y \in X^*$ tels que $u = xy$ et $v = yx$.

5.1.2 Mots de Lyndon.

Pour plus de détails sur les mots de Lyndon, on peut consulter les références [2], [17] ou [18].

Definition 5.1 *Un mot $w \in X^*$ est un mot de Lyndon si et seulement si, il vérifie l'une des deux propriétés, équivalentes, suivantes :*
(i) il est strictement plus petit que tous ses conjugués,
(ii) il est strictement plus petit que ses facteurs droits propres.
On notera L l'ensemble des mots de Lyndon sur X.

Propriétés.

1. Soit $b \in L\backslash X$ et soit m son plus long facteur droit propre dans L. Si $b = lm$, le couple $\sigma(b) = (l,m)$ est appelé *la factorisation standard* de b. Elle vérifie : $l \in L$ et $l < lm < m$.

2. $b \in L$ est un mot de Lyndon si et seulement si

$$\begin{cases} & b \in X, \\ \text{ou} & \\ & b = lm \text{ avec } l,m \in L \text{ et } l < m. \end{cases}$$

Ceci donne un moyen *algorithmique* pour la construction des mots de Lyndon.

Lemme 5.1 [17],[19].- *Soient $b \in L\backslash X$ et $\sigma(b) = (l,m)$ sa factorisation standard, et soit c un mot de Lyndon vérifiant $b < c$. Alors le couple (b,c) est la factorisation standard de bc si, et seulement si, $c \leq m$.*

Théorème 5.1 Lyndon [17],[18].- *Tout mot $w \in X^*$ peut être écrit de manière unique comme suit:*

$$w = l_1^{i_1} l_2^{i_2} \cdots l_n^{i_n} \tag{16}$$

où chaque $l_i \in L$ et $l_1 > l_2 > \cdots > l_n$.

5.1.3 Base de Lyndon de $\mathcal{L}ie\langle X\rangle$.

Associons à tout mot de Lyndon b le polynôme de Lie P_b, défini récursivement comme suit :

$$\begin{cases} P_x = x \quad pour \quad x \in X, \\ P_b = [P_l, P_m], \quad pour\ b \in L\backslash X,\ tel\ que\ \sigma(b) = (l,m). \end{cases} \tag{17}$$

Les polynômes P_b forment une base de $\mathcal{L}ie\langle X\rangle$ appelée base de Lyndon ([17], [27]). On peut donc construire les mots de Lyndon, et les polynômes de Lie associés, en les énumérant par longueur, comme l'indique l'exemple 5.1.

Exemple 5.1 $X = \{x_0, x_1\}$,

Longueur	*Mots de Lyndon*	*Base de Lyndon*
1	x_0	x_0
	x_1	x_1
2	x_0x_1	$[x_0, x_1]$
3	$x_0^2x_1$	$[x_0, [x_0, x_1]]$
	$x_0x_1^2$	$[[x_0, x_1], x_1]$
...	...	...

Mots et base de Lyndon : énumération lexicographique par longueur.

Soit alors $w \in X^*$ un mot quelconque. Par la formule (16), on peut l'écrire :

$$w = l_1^{i_1} l_2^{i_2} \cdots l_n^{i_n}.$$

On peut alors lui associer le polynôme :

$$P_w = P_{l_1}^{i_1} P_{l_2}^{i_2} \cdots P_{l_n}^{i_n}$$

La famille $\{ P_w \mid w \in X^* \}$ est exactement la base PBW.L de $K\langle X\rangle$ associée à la base de Lyndon de $\mathcal{L}ie\langle X\rangle$ par le théorème PBW.

La construction de la base duale $\{ S_w \mid w \in X^* \}$ découle du théorème suivant ([18]) :

Théorème 5.2 *Le polynôme S_w défini par (11) peut être défini récursivement comme suit :*

$$S_w = \begin{cases} \varepsilon & si \quad w = \varepsilon, \\ x_i S_v & si \quad w = x_i v \in L, \\ \dfrac{1}{i_1! i_2! \cdots i_k!} S_{l_1}^{\sqcup\!\sqcup i_1} \sqcup\!\sqcup S_{l_2}^{\sqcup\!\sqcup i_2} \sqcup\!\sqcup \cdots \sqcup\!\sqcup S_{l_k}^{\sqcup\!\sqcup i_k} & si \quad w = l_1^{i_1} l_2^{i_2} \cdots l_k^{i_k},\ où\ l_j \in L. \end{cases} \tag{18}$$

Cette définition récursive des S_w donne un algorithme simple pour les calculer.

6 Algorithmes et mise en oeuvre

6.1 Listes standard

Definition 6.1 *Soit $lst = [u_1, u_2, \ldots, u_n]$ une liste d'éléments de L. On dira que lst est standard si et seulement si elle vérifie la propriété suivante :*

$$(S) \quad \begin{cases} u_i \text{ est une lettre,} \\ ou \\ si\ u_i = xy \text{ tel que } \sigma(u_i) = (x,y) \text{ alors } y \geq u_j,\ j \geq i. \end{cases}$$

1. Si tous les facteurs u_i sont des lettres, alors lst vérifie (S),
2. Si lst est décroissante (ie $u_1 \geq u_2 \ldots \geq u_n$), alors lst vérifie encore (S).

Une *inversion* est un couple (u_i, u_{i+1}) extrait de lst tel que $u_i < u_{i+1}$.

```
Factoriser(w)
  lst ← [x_1, x_2, ···, x_n],   lst_1 ← lst,
  k ← inversion(lst),
  Tant Que k ≠ 0 Faire Debut
    lst_1 ← [x_1, x_2, ..., x_k.x_{k+1}, ..., x_n],
    lst_2 ← [x_1, x_2, ..., [x_k, x_{k+1}], ..., x_n],
    k ← inversion(lst),
  FinTQ
  long ← longueur(lst_1),
  w ← ∏_{i=1}^{long} lst_1[i],   P_w ← ∏_{i=1}^{long} lst_2[i],
  Si long = 1 Alors Debut
    lst_1[1] est un mot de Lyndon,
    lst_2[1] est le polynome de Lie correspondant,
  FinSI
```

Algorithme.1: Factorisation de Mots

6.2 Algorithme de factorisation

Les algorithmes présentés ici sont une extension de ceux présentés dans [19]. L'algorithme 1 utilise la notion de liste standard. Il permet simultanément une enumération des mots de Lyndon, de la base de Lyndon de $\mathcal{L}ie\langle X\rangle$ et de la base PBW.L jusqu'à un ordre donné.

Dans cet algorithme, inversion(lst) est une fonction qui renvoie la position, dans la liste lst, de la première inversion depuis la droite. Le "crochet" : $[x_k, x_{k+1}]$ représente le crochet de Lie et le point "." dans $x_k.x_{k+1}$ représente le produit non commutatif (concaténation).

L'algorithme 2 permet de construire la base duale de PBW.L. Il utilise l'algorithme précédent pour factoriser un mot $w \in X^*$.

La fonction Gauche(w) extrait la première lettre du mot w. La fonction Transformer(lst_1) prend en argument la liste lst_1 construite par la fonction Factoriser et la transforme en une liste de listes à deux éléments [mot de Lyndon,sa puissance].

La fonction Shufflen(lst_1) prend en argument une liste de la forme :

$$lst_1 = [[l_{j_1}, i_1], [l_{j_2}, i_2], \ldots, [l_{j_k}, i_k]].$$

Elle utilise la fonction Duale, la fonction Shufflep(P, n) qui calcule la puissance shuffle d'un polynôme et la fonction Shuffle pour calculer

$$\frac{1}{i_1!i_2!\ldots i_k!}\mathtt{Duale}(l_{j_1})^{ш i_1} ш \mathtt{Duale}(l_{j_2})^{ш i_2} ш \ldots ш \mathtt{Duale}(l_{j_k})^{ш i_k}.$$

6.3 Calcul de l'approximation structurelle nilpotente d'ordre k

Soit $\mathcal{L} = (l_1, l_2, \ldots)$ une énumération des mots de Lyndon (on utilise par exemple l'ordre lexicographique par longueur). Pour tout $k \in I\!N$, on peut calculer l'approximation :

$$\mathcal{A}_k = \prod_{j\leq k, lex} \exp(S_{l_j} \otimes P_{l_j}) \qquad \text{(ordre lexicographique)}.$$

<u>Duale</u>(w)
  <u>Si</u> $w = 1$ <u>Alors</u> 1
  <u>Sinon</u> <u>Si</u> $w \in L$ <u>Alors</u>
    Gauche(w).Duale(reste(w))
  <u>Sinon</u> <u>Debut</u>
    Factoriser(w),
    $lst_1 \longleftarrow$ Transformer(lst_1),
    Shufflen(lst_1)
  <u>FinSI</u>

Algorithme.2: Construction de la base duale

En Evaluant, on obtient un approximant de l'"opérateur de transport" H du système :

$$A_k(H) = \prod_{j \leq k, lex} \exp\Big(\mathcal{E}_a(S_{l_j})\mathcal{Y}(P_{l_j})\Big) \quad \text{(ordre lexicographique)} \tag{19}$$

On appelle $A_k(H)$ l'*approximant structurel nilpotent* d'ordre k de H. On aura $A_k(H) = H$ pour k assez grand si et seulement si l'algèbre de Lie $\mathcal{Y}(\mathcal{L}ie\!<\!X\!>)$ (algèbre de commande de (Σ)) est nilpotente.

Le calcul de $A_k(H)$ nécessite le calcul des Evaluations $\mathcal{E}_a(S_{l_j})$, qui découle immédiatement du théorème 5.2 et de la propriété (P3), et dont la programmation s'intègre naturellement à l'Algorithme 2.

References

[1] D.Claude.– Découplage et linéarisation des systèmes non linéaires par bouclages statiques, *Thèse d'Etat, Université de Paris Sud, Centre d'Orsay,* 1986.

[2] J.P.Duval.– Génération d'une section des classes de conjugaison et arbre des mots de Lyndon de longueur bornée, *Theoret. Comput. Sci.,* 60, p.255–283, 1988.

[3] M.Fliess.– Fonctionnelles causales non linéaires et indéterminées non commutatives, *Bull. Soc. Math. France,* 109, p.3-40, 1981.

[4] M.Fliess.– Réalisation locale des systèmes non linéaires, algèbres de Lie filtrées transitives et séries génératrices, *Invent. Math.*71, p.521-537, 1983.

[5] M.Fliess, M.Lamnabhi et F.Lamnabhi-Lagarrigue, An algebraic approach to nonlinear functional expansions, *IEEE Trans. Circ. Syst.,* CAS-30, p. 554–570, 1983

[6] C.Hespel & G.Jacob.– Approximation of nonlinear systems by bilinear ones, *Algebraic and Geometric Methods in Nonlinear Control Theory,* p. 511-520, 1986.

[7] C.Hespel & G.Jacob.– Calcul des approximations locales bilinéaires de systèmes analytiques, *APII,* 23, p. 331–349, 1989.

[8] C.Hespel.– Approximation of nonlinear dynamic systems by rational series, *to appear in "Algebraic and Computing Traitment of Noncommutative power Series" (ed. G.Jacob and C.Reutenauer), Theoret. Comput. Sci.,* 1990.

[9] Hoang Ngoc Minh.– Evaluation transform, *to appear in "Algebraic and Computing Traitment of Noncommutative power Series" (ed. G.Jacob and C.Reutenauer), Theoret. Comput. Sci.,* 1990.

[10] Hoang Ngoc Minh & G.Jacob.– Decoupling for nonlinear control systems, *Publication du LIFL Lille,* IT–162, 1989.

[11] Hoang Ngoc Minh & G.Jacob.– Symbolic Calculus and Volterra Series, *IFAC Symposium "Nonlinear Control Systems Design", Capri–Italy,* June 1989.

[12] Hoang Ngoc Minh & G.Jacob.– Evaluation transform and symbolic calculus for nonlinear control systems, *"Ninth international conference analysis and optimization of systems", Antibes–France,* June 1990.

[13] Hoang Ngoc Minh & G.Jacob.– Evaluation transform and its implementation in Macsyma, *"New Trends in Systems Theory", Genoa–Italy,* July 1990.

[14] J.E.Humphreys.– *Introduction to Lie algebras and representation theory,* Springer–Verlag. 1980.

[15] G.Jacob and N.Oussous.– Sur un résultat de REE: séries de Lie et algèbres de mélange, Publication du LIFL Lille, IT–103, 1987.

[16] G.Jacob and N.Oussous.– Local and minimal realization of nonlinear dynamical systems and Lyndon words, *IFAC Symposium "Nonlinear Control Systems Design", Capri–Italy,* June 1989.

[17] M.Lothaire.– Combinatorics on words, *Reading, Massachusetts,* 1983.

[18] G.Melançon and C.Reutenauer.– Lyndon words, free algebras and shuffles, Publication du LITP Paris et UQAM Montreal, Quebec, 87–63,1987.

[19] N.E.Oussous.– Mots de Lyndon et Macsyma, *CALSYF "revue du GRECO de Calcul Formel",* 1988.

[20] N.E.Oussous.– Macsyma Computation of Local Minimal Realization of Dynamical Systems of which Generating Power Series are Finite, *To appear in J. Symbolic Computation,* 1990.

[21] N.E.Oussous.– Computation, on Macsyma, of the minimal differential representation of noncommutative polynomials, *to appear in "Algebraic and Computing Traitment of Noncommutative power Series" (ed. G.Jacob and C.Reutenauer), Theoret. Comput. Sci.,* 1990.

[22] C.Reutenauer.– The local realisation of generating series of finite Lie rank, in: M.Fliess and M.Hazewinkel, eds., *Algebraic and Geometric Methods In Nonlinear Control Theory,* p. 33–43, 1986.

[23] T.Huillet, A.Monin & G.Salut.– Lie algebraic canonical representations in nonlinear control systems, *Math. systems theory,* 20, p. 193–213, 1987.

[24] T.Huillet, A.Monin & G.Salut.– Représentations exponentielles en commande non linéaire, *APII,* 21, p. 419–448, 1987.

[25] T.Huillet, A.Monin & G.Salut.– Représentations exponentielles des systèmes analytiques généraux, *APII,* 24, p. 37–56, 1990.

[26] H.J.Sussmann.– A product expansion for Chen Series, *in "Theory and Applications of Nonlinear Control Systems",* C.I. Byrns and Lindquist, eds., p. 323–335, 1986.

[27] G.Viennot.– *Algèbres de Lie libres et monoïdes libres,* (Lecture Notes In Mathematics, Springer–Verlag. 691,1978).

[28] G.Viennot et P.Leroux.– A combinatorial approach to nonlinear functional expensions: an introduction with example, *to appear in "Algebraic and Computing Traitment of Noncommutative power Series" (ed. G.Jacob and C.Reutenauer), Theoret. Comput. Sci.,* 1990.

[29] Y.Wang et E.D.Sontag.– Realization and Input/Output relations: the analytic case, *Proceding of the 28th Conference on Decision and Control IEEE, Tampa–Florida,* December 1989.

BIBLIOTHEQUE INTERUNIVERSITAIRE SECTION DE SCIENCES LILLE

Abstract

We develop some computing tools in order to obtain some technical results in control theory, and their implementation via algebraic computation :

. In the first chapter, we recall some results concerning formal power series on commutative and noncommutative variables. We recall also some elements of free Lie algebra.

. In the second chapter, we develop *the basic tools of a symbolic calculus* for nonlinear control systems by introducing *the Evaluation transform.* This transform generalizes the inverse Laplace transform in the nonlinear area. It is also an extension on noncommutative variables of the exponential transform of the ordinary generating series to the associated exponential generating series.

. In the third chapter, we study *a duality between Chen series and Fliess series.* We give *a sum expansion for Chen series.* This sum expansion allows us to obtain simply *the Taylor expansion of Volterra kernels.*

. In the fourth chapter, we study *the nonlinear differential equations with forcing terms.* We propose a concise and efficient recursive method to compute the approximate solution (in MACSYMA) by transforming these equations (via Fliess series and Evaluation transform) in *convolution equations.*

. In the fifth chapter, we use Fliess series and free Lie algebra to study *the decoupling problems for nonlinear control systems* (disturbance decoupling problem, diagonal or triangular decoupling problem and noninteracting control systems).

. In the sixth chapter, we point out our choice of the internal representation, in particular the binary trees for *representing formal power series.* We present our recursive algorithms, in particular the use of $\lambda-$notations in MACSYMA to compute *the Evaluation of the noncommutative rational fractions.*

Key words

Algebraic computation, decoupling, formal power series, generating series, Lie algebra, nonlinear control systems, nonlinear differential equations, symbolic calculus.

PPN 036114189

Résumé

Nous développons certains outils informatiques, et les mettons en œuvre pour obtenir puis implanter grâce au calcul formel des résultats fins en théorie de la commande :

. Dans le premier chapitre, nous rappelons des résultats concernant les séries formelles en variables commutatives et non commutatives. Nous rappelons également des éléments de l'algèbre de Lie libre.

. Dans le deuxième chapitre, nous développons *les bases d'un calcul symbolique* pour les systèmes dynamiques non linéaires en introduisant *la transformation d'Evaluation*. Cette transformation généralise la transformation de Laplace inverse pour les systèmes dynamiques non linéaires. Elle est aussi une extension en plusieurs variables non commutatives du passage de la série génératrice ordinaire à la série génératrice exponentielle.

. Dans le troisième chapitre, nous étudions *une dualité entre les séries de Chen et les séries de Fliess*. Nous donnons une *graduation des séries de Chen*. Cette graduation nous permet d'obtenir simplement *les développements de Taylor des noyaux de Volterra*.

. Dans le quatrième chapitre, nous étudions *les équations différentielles non linéaires en régime forcé*. Nous proposons une méthode itérative simple et efficace pour calculer une solution approchée (en MACSYMA) en transformant ces équations (via les séries de Fliess et la transformation d'Evaluation) en *équations de convolution*.

. Dans le cinquième chapitre, nous utilisons les séries de Fliess et l'algèbre de Lie libre pour étudier *les problèmes de découplage des systèmes dynamiques non linéaires* (rejet de perturbations, découplage diagonal ou triangulaire, systèmes non interactifs).

. Dans le sixième chapitre, nous indiquons nos choix de structures de données, en particulier les arbres binaires pour *représenter les séries formelles non commutatives*. Nous présentons nos algorithmes récursifs, en particulier l'utilisation les $\lambda-$notations de MACSYMA pour calculer *l'Evaluation des fractions rationnelles non commutatives*.

Mots clés

Algèbre de Lie libre, calcul formel, calcul symbolique, découplage, équations différentielles non linéaires, séries formelles, séries génératrices, systèmes dynamiques non linéaires.