

Remarques algorithmiques liées au rang d'un opérateur différentiel linéaire

Moulay Barkatou, Frédéric Chyzak et Michèle Loday-Richaud

*Université de Limoges, Faculté des Sciences
Laboratoire d'Arithmétique, de Calcul formel et d'Optimisation
123, avenue Albert Thomas, 87060 LIMOGES cedex, France
email : moulay.barkatou@unilim.fr*

*INRIA – Projet Algorithmes
Domaine de Voluceau,
Rocquencourt BP 105, 78153 Le CHESNAY cedex, France
email : frederic.chyzak@inria.fr*

*Université d'Angers – U.F.R. Sciences
U.M.R. 6093 du C.N.R.S. (Algèbre et géométrie)
2 boulevard Lavoisier, 49045 ANGERS cedex 01, France
email : michele.loday@univ-angers.fr*

*Dédié à Jean Thomann
à l'occasion de son 60^e anniversaire*

Résumé. Dans cet article nous décrivons et nous comparons différentes méthodes algorithmiques pour résoudre le problème suivant qui apparaît naturellement lors de l'implantation en calcul formel des solutions formelles d'équations différentielles et de leurs (multi-)sommés : étant donné une série $\hat{f}(x) = \sum_{m \geq 0} u_m x^m$ solution d'une équation différentielle linéaire ordinaire et un nombre entier $r > 0$ construire une équation différentielle vérifiée par les sous-séries $\hat{f}^j(x) = \sum_{m \geq 0} u_{j+mr} x^{j+mr}$ de ses termes de r en r . Les méthodes proposées sont implantées en MAPLE et nous donnons l'adresse des programmes correspondants.

En appendice nous indiquons une démarche constructive réaliste pour le calcul effectif des invariants formels d'une équation ou d'un système.

Mots clés. Équations différentielles linéaires ordinaires, équations aux différences linéaires, extraction de sous-séries, transformation de Mellin, méthodes algorithmiques.

Abstract. This paper aims at introducing and comparing various methods for solving the following problem which arises naturally in computer algebra for differential equations : given a

series $\hat{f}(x) = \sum_{m \geq 0} u_m x^m$ solution of an ordinary linear differential equation and given an integer $r \geq 2$ find a differential equation satisfied by the sub-series $\hat{f}^j(x) = \sum_{m \geq 0} u_{j+mr} x^{j+mr}$. These methods have been implemented in MAPLE. References are given to the corresponding programs.

Possible realistic algorithms to effectively calculate the formal invariants of an ordinary linear differential system of order one and arbitrary rank are sketched in an Appendix.

Keywords. Ordinary linear differential equations, linear difference equations, extraction of sub-series, Mellin transform, algorithmic methods.

AMS classification. Primary 34M25, 12H05, 12H10, Secondary 68W30.

Table des matières

1	Introduction	89
2	Approche directe dans l'anneau des opérateurs différentiels	92
2.1	Calcul dans le cadre des équations	93
2.1.1	Par saturation galoisienne.	93
2.1.2	Par élimination.	94
2.2	Calcul par l'intermédiaire des systèmes	96
2.2.1	Par saturation galoisienne directe.	96
2.2.2	Par réduction du rang.	98
3	Approche duale dans l'espace des opérateurs aux différences	102
3.1	Saturation galoisienne de l'équation aux différences	103
3.2	Élimination polynomiale	103
3.3	Saturation galoisienne du système compagnon	104
3.4	Réduction du rang	105
4	Comparaison des diverses méthodes et exemple	107
4.1	Relation entre L et Λ	107
4.2	Exemple de l'équation de Ramis–Sibuya	108
4.3	Remarque sur l'utilisation d'équations différentielles linéaires affines	113
5	Appendice : invariants formels d'une équation ou d'un système	114
5.1	Invariants formels des équations différentielles	114
5.2	Invariants formels des systèmes différentiels	118

5.2.1	Détermination de la régularité, rang de Moser et calcul des solutions régulières.	119
5.2.2	Cas des singularités irrégulières.	121
5.3	Vecteur cyclique	125

1 Introduction

Étant donné un système différentiel linéaire

$$\frac{dY}{dx} = \frac{1}{x^{r+1}} A(x)Y$$

de dimension n à coefficients méromorphes d'ordre $r + 1$ à l'origine ($r \in \mathbb{Z}$, $A \in \text{gl}(n, K)$ où $K = \mathbb{C}\{x\}$ est l'anneau des séries convergentes à l'origine et $A(0) \neq 0$) il est d'usage d'appeler *rang de Poincaré* du système le nombre r . Ce nombre n'est pas invariant par transformation de jauge méromorphe $Y = T(x)Z$ où $T \in K[1/x]$ mais il admet une valeur minimale appelée *vrai rang de Poincaré*. Les systèmes de rang de Poincaré $r \leq -1$ n'ont pas de singularité à l'origine, ceux de rang $r = 0$ ont une singularité régulière et, lorsque le vrai rang de Poincaré est strictement positif, l'origine est un point singulier irrégulier.

On dit, de même, qu'une équation différentielle

$$a_n(x)y^{(n)} + a_{n-1}(x)y^{(n-1)} + \cdots + a_0(x)y = 0$$

à coefficients méromorphes en $x = 0$ est de rang r si les coefficients $\frac{a_{n-1}}{a_n}, \dots, \frac{a_0}{a_n}$ de l'équation normalisée ont un pôle d'ordre $r + 1$ en 0.

Lorsque le système est de vrai rang de Poincaré égal à r et qu'il porte le seul niveau r (les exponentielles d'une solution fondamentale formelle sont toutes d'ordre r), les séries solutions formelles divergentes sont r -sommables et, en particulier, de type Gevrey d'ordre r . Lorsqu'il porte plusieurs niveaux $r_1 < r_2 < \cdots < r_s \leq r$ ses solutions formelles divergentes sont au pire multisommables des niveaux (r_1, r_2, \dots, r_s) et de type Gevrey d'ordre r_1 ([M95]).

C'est au début des années 80 que, grâce à de nouvelles avancées de la théorie des équations différentielles ordinaires, s'est exprimé un intérêt croissant pour les méthodes effectives et le développement d'algorithmes liés aux singularités d'équations différentielles : calcul de solutions formelles, d'invariants méromorphes, de sommes des solutions séries divergentes, . . .

Ces dernières années, Jean Thomann en particulier s'est beaucoup investi dans le développement et l'implantation des algorithmes de sommation. Les différentes méthodes théoriques connues reposent sur des transformations intégrales de type opérateurs de Borel et de Laplace associés aux différents niveaux de l'équation. Or, dans cette famille de transformations, les opérateurs de Borel et de Laplace classiques, i.e.,

de niveau 1, jouent un rôle particulier qui les rend incontournables pour mener à bien les différentes étapes de calcul. Pour s'en convaincre mentionnons simplement le fait suivant : la transformée de Borel d'un opérateur différentiel à coefficients polynomiaux est un opérateur différentiel à coefficients polynomiaux auquel on peut appliquer toute la théorie des équations différentielles et, en particulier, les algorithmes numériques ; au contraire, la transformée de Borel d'ordre $r > 1$ d'un tel opérateur est une équation de r -convolution mal connue. Mais, dans le cas d'un niveau unique r , cas de la r -sommabilité, il se trouve que l'on peut se limiter à l'implantation du cas $r = 1$, cas classique de la sommabilité au sens de Borel-Laplace ; en effet, si une série $\hat{f}(x) = \sum_{m \geq 0} u_m x^m$ est r -sommable alors les sous-séries extraites $\hat{f}^j(x) = x^j \sum_{m \geq 0} u_{mr} x^{mr}$ des termes de r en r sont 1-sommables en la variable $t = x^r$ et il y a compatibilité des procédés de sommation. Dans le cas de plusieurs niveaux, J. Thomann a examiné à la fois la méthode des accélératrices (due à J. Écalle) et celle de Laplace-itérée (due à W. Balsler). Ces méthodes consistent à appliquer successivement des transformations de type transformations de Borel ou de Laplace. Là encore, la tactique la plus efficace à ce jour consiste à se ramener systématiquement au niveau 1, i.e., au niveau 1 à chaque étape. Pour plus de détails on renvoie à [JNT96] et à [NT96]. On est ainsi conduit à se poser la question suivante :

Comment obtenir le plus simplement possible des équations différentielles auxquelles satisfont les $\hat{f}^j(x)$ à partir d'une équation différentielle à laquelle satisfait $\hat{f}(x)$?

Nous explicitons au paragraphe 2 un certain nombre de méthodes conduisant à une équation différentielle qui admet simultanément toutes les \hat{f}_j pour solutions. Puis nous indiquons au sous-paragraphe 4.3 comment obtenir des équations différentielles (chacune d'ordre inférieur mais plusieurs équations) auxquelles satisfont séparément chacune de séries \hat{f}_j .

La même question apparaît de façon sous-jacente dans les algorithmes de calcul direct, c'est-à-dire, sans sommation, des invariants méromorphes comme, par exemple, dans [L-R90] en dimension 2. Ce sont là les relations de récurrence sur les coefficients des séries $\hat{f}^j(x)$ qui sont directement utiles et on est amené à considérer le problème dual :

Comment obtenir le plus simplement possible des équations aux différences auxquelles satisfont les suites de coefficients des séries $\hat{f}^j(x)$ à partir d'une équation aux différences à laquelle satisfait la suite de coefficients de la série $\hat{f}(x)$?

Ces deux problèmes se correspondent par transformation de Mellin. Nous avons cherché, dans la mesure du possible, à présenter en parallèle des méthodes traitant directement l'équation différentielle et les méthodes analogues duales pour l'équation aux différences qui s'en déduit par transformation de Mellin. Nous montrons que les

approches directes conduisent toutes à la même équation différentielle $Ly = 0$ admettant les \hat{f}^j pour solutions. Les approches duales conduisent à une même équation aux différences $\Lambda \tilde{y} = 0$. Il eût été raisonnable d'espérer que L et Λ se correspondent par transformation de Mellin. Il n'en est rien comme le confirme l'exemple de l'équation de Ramis–Sibuya que nous explicitons. Nous expliquerons pourquoi il en est ainsi au sous-paragraphe 4.1.

Notons enfin que le développement d'algorithmes adaptés aux systèmes incite à se poser les mêmes questions en termes de systèmes. Les performances de l'algorithme 5*(1) ou même 5(1) sur l'exemple de l'équation de Ramis–Sibuya montrent d'ailleurs qu'on peut avoir intérêt à travailler avec un système plutôt qu'une équation même si la donnée initiale est une équation (*cf.* paragraphe 4.2).

Il existe des solutions connues voire programmées à ces problèmes, en particulier, sous leur forme duale ([SZ94], [N95]). À notre connaissance toutes les méthodes proposées en termes de systèmes sont originales.

Les méthodes développées ou au moins certains de leurs aspects sont valables de façon générale pour des équations ou des systèmes à coefficients méromorphes. Pour qu'elles deviennent totalement algorithmiques, nous nous restreindrons néanmoins à des équations ou à des systèmes à coefficients polynomiaux ou rationnels.

La distinction que nous faisons entre des méthodes dites de saturation galoisienne et des méthodes dites d'élimination est à frontière floue dans la mesure où ces éliminations ont pour objet de réaliser la saturation galoisienne.

Les arguments galoisiens ont été à la base des travaux initiateurs de Turrittin sur la question plus générale de la réduction du rang des systèmes ([T63], [L-R01]). L'argument de Turrittin est simple et analogue à celui évoqué plus haut : si un système est de rang r en la variable x il est de rang 1 en la «variable» $t = x^r$. La simple substitution de t à x^r fait apparaître des puissances fractionnaires de t . Pour rester dans le cadre des systèmes à coefficients méromorphes il s'agit alors de saturer l'effet de ce changement de variable par l'action du groupe de Galois \mathfrak{S}_r de l'équation $x^r = t$; ce faisant, on multiplie la dimension par r (du moins génériquement).

Les arguments d'élimination s'appuient sur des algorithmes de Gauss ou d'Euclide dans des algèbres de polynômes non commutatives.

L'exécution de ces algorithmes en MAPLE sur l'exemple de l'équation de Ramis–Sibuya nous amène à faire les commentaires suivants. Les temps de calcul requis sont raisonnablement comparables lorsque $r = 2$. En revanche, lorsque $r > 2$, ils deviennent significativement différents. Les algorithmes les plus coûteux sont ceux qui font un usage effectif de nombres algébriques sur le corps des coefficients (essentiellement de $\omega = e^{2\pi i/r}$) et ceux qui utilisent de l'arithmétique non commutative. Les plus performants sont ceux qui s'appuient sur la théorie de la réduction du rang des systèmes ou le produit de Hadamard en restant dans le cadre de l'élimination linéaire.

Le calcul de la complexité de ces algorithmes reste à faire. Les procédures que nous avons écrites n'ont pas, pour l'instant, été optimisées. Les observations faites sur l'exemple de l'équation de Ramis–Sibuya nous incitent d'ailleurs à ne pas porter un même intérêt à tous ces algorithmes.

Pour répondre à une demande explicite de certains utilisateurs, nous complétons cet article par un appendice sur le calcul des invariants formels des opérateurs différentiels (équations ou systèmes) et, en particulier, de leurs niveaux qui sont des valeurs privilégiées pour la réduction du rang. Il ne s'agit pas ici de présenter des méthodes nouvelles mais de rassembler et d'ordonner des résultats algorithmiques existants en un mode opératoire précis dont chacun pourra utiliser tout ou partie. Les programmes correspondants sont implantés en MAPLE et disponibles à l'adresse électronique citée en référence [BCL].

2 Approche directe dans l'anneau des opérateurs différentiels

Dans ce paragraphe nous considérons une équation différentielle linéaire

$$Dy \equiv a_n(x)y^{(n)} + a_{n-1}(x)y^{(n-1)} + \dots + a_0(x)y = 0 \quad (1)$$

dont les coefficients a_0, a_1, \dots, a_n sont polynomiaux. Nous supposons que cette équation admet une solution série formelle $\hat{f}(x) = \sum_{m \geq 0} u_m x^m$ et nous nous proposons de déterminer une équation différentielle qui admet pour solutions les séries

$$\hat{f}^j(x) = \sum_{m \geq 0} u_{j+mr} x^{j+mr}.$$

Nous notons $\partial = \frac{d}{dx}$ l'opérateur de dérivation usuelle et $\mathbb{W}_1 = \mathbb{C}[x][[\partial]]$ l'algèbre de Weyl des opérateurs différentiels à coefficients polynomiaux à une indéterminée x dans laquelle les relations de commutation sont engendrées par l'unique relation

$$x\partial - \partial x = 1.$$

L'algèbre de Weyl \mathbb{W}_1 se plonge de façon canonique dans l'algèbre $\mathbb{C}(x)[\partial]$ des opérateurs différentiels à coefficients rationnels munie de la même relation de commutation. Celle-ci a l'avantage d'être euclidienne à gauche (et à droite), donc aussi principale à gauche (et à droite) ([Ore33]).

Quitte à multiplier D par une puissance convenable de x , nous supposons désormais que le 0-poids de D (cf. 4.1) est positif ou nul et que, par conséquent, D est polynomial en x et en l'opérateur d'Euler $\delta = x\partial$. Les opérateurs D considérés appartiennent alors à l'algèbre $\mathbb{A}_1 = \mathbb{C}[x][[\delta]]$ si on utilise la dérivation δ ou à l'image \mathbb{A}'_1 de celle-ci dans \mathbb{W}_1 si on utilise la dérivation ∂ . Les relations de commutation dans \mathbb{A}_1 sont engendrées par l'unique relation

$$x\delta - \delta x = x.$$

Ceux des algorithmes ci-dessous qui sont écrits avec ∂ restent valables, à des modifications mineures de notation près, avec δ .

2. Dans l'algèbre $\mathbb{C}(x)[\partial]$ (ou dans $\mathbb{C}(x)[\delta]$) déterminer un plus petit commun multiple à gauche de D_0, D_1, \dots, D_{r-1} .
3. Si on le souhaite, déterminer l'unique représentant unitaire et primitif de ce ppcm dans \mathbb{A}'_1 , d'où L .

La procédure **LCLM** du paquetage DEtools de MAPLE fournit un ppcm d'une liste d'opérateurs ; le résultat est donné avec un coefficient de tête égal à 1. On obtient un représentant de ce ppcm dans l'anneau \mathbb{A}'_1 en chassant les dénominateurs.

La procédure **skew_gcdex** du paquetage Ore_algebra de MAPLE fournit un pgcdd de deux polynômes p et q et la relation de Bézout qu'ils vérifient avec des coefficients dans \mathbb{A}'_1 , i.e., sans dénominateur. Elle permet aussi de calculer leur ppcm bien qu'elle ne le donne pas directement. La commande **skew_gcdex**(p, q, Dx, A) où $A := \mathbf{skew_algebra}(\text{diff} = [Dx, x])$ restitue $[g, a, b, u, v]$ où $g = ap + bq$ est le pgcdd de p et de q et où $up = -vq$ est le ppcm de p et de q . On obtient le ppcm cherché par la commande **skew_product**(u, p, A).

diff → *diff*

2.1.2 Par élimination. On peut conduire l'élimination de diverses manières. Nous en présentons deux variantes.

• **Première variante :**

On introduit une nouvelle indéterminée Ω . Si l'on remplace x par Ωx et ∂ par $\Omega^{-1}\partial$ (ou δ par lui-même), quitte à multiplier au préalable D par une puissance convenable de x comme ci-dessus, on obtient un polynôme $P(\Omega)$ à coefficients dans \mathbb{A}'_1 . L'existence d'un ppcm de deux éléments quelconques de l'anneau $\mathbb{C}(x)[\partial]$ permet de définir un corps des fractions à gauche non commutatif F_g formé des quotients formels $A(x, \partial)^{-1}B(x, \partial)$ pour $A(x, \partial) \neq 0$ et $B(x, \partial) \in \mathbb{C}(x)[\partial]$ muni de la relation d'équivalence évidente ([Ore33] I.3, [Co] 0.5). On peut ainsi plonger de façon canonique \mathbb{A}'_1 et $\mathbb{C}(x)[\partial]$ dans le corps F_g . Dans l'anneau euclidien $F_g[\Omega]$ (ici, c'est F_g qui est non-commutatif alors que Ω commute avec les éléments de F_g) les polynômes $P(\Omega)$ et $\Omega^r - 1$ sont premiers entre eux puisque, pour $j = 0, 1, \dots, r-1$, $P(\omega^j) = D_j \neq 0$. Ils vérifient donc une relation de Bézout

$$S(\Omega)P(\Omega) + T(\Omega)(\Omega^r - 1) = G$$

où, quitte à chasser les dénominateurs, on peut supposer que G est dans \mathbb{A}'_1 alors que S et T sont des polynômes de $\mathbb{A}'_1[\Omega]$ de degré respectivement inférieur à r et au degré de P .

Notons que, quitte à chasser les dénominateurs à chaque étape, l'algorithme d'Euclide peut être conduit dans l'algèbre $\mathbb{A}'_1[\Omega]$ en effectuant des pseudo-divisions.

En spécifiant Ω en les diverses racines $r^{\text{èmes}}$ de l'unité ω^j on en déduit que G est un multiple à gauche ML du ppcm L obtenu précédemment. Par ailleurs, $S(\Omega)$ est de degré $\leq r-1$ en Ω ; le même facteur M divise à gauche dans $\mathbb{C}(x)[\partial]$ tous les $S(\omega^j)$ donc aussi toutes leurs combinaisons linéaires et, en particulier, les coefficients de $S(\Omega)$. Il en résulte que M divise à gauche $T(\Omega)(\Omega^r - 1)$, donc également

le coefficient constant $T(0)$ de $T(\Omega)$ puis, par récurrence, tous les coefficients de T . Après simplification par M , factorisation du contenu commun des coefficients et normalisation, on obtient donc la relation $G = uL$ où $u \in \mathbb{C}[x]$.

Algorithme : version 2.

1. Déterminer le polynôme $P \in \mathbb{A}'_1[\Omega]$ en substituant Ωx à x et $\Omega^{-1}\partial$ à ∂ (ou δ à lui-même) dans l'opérateur D .
2. Grâce à un algorithme d'Euclide dans l'algèbre $F_g[\Omega]$ où F_g désigne le corps des fractions à gauche de $\mathbb{C}(x)[\partial]$ déterminer la relation de Bézout

$$S(\Omega)P(\Omega) + T(\Omega)(\Omega^r - 1) = uL$$

dans laquelle S et T sont dans $\mathbb{A}'_1[\Omega]$, leurs coefficients sans contenu commun dans \mathbb{A}'_1 , $u \in \mathbb{C}[x]$ et $L \in \mathbb{A}'_1$ est primitif et normalisé.

• **Deuxième variante :**

Exprimons D en la dérivation δ :

$$D = b_n(x)\delta^n + b_{n-1}(x)\delta^{n-1} + \dots + b_0(x)$$

où, conformément à l'hypothèse que le 0-poids de D est positif ou nul, les coefficients $b_j(x)$ appartiennent à $\mathbb{C}[x]$.

Il s'agit de déterminer le plus petit multiple à gauche de D dont les coefficients ne contiennent que des puissances de x^r , i.e., en posant $t = x^r$, un multiple à gauche de D d'ordre minimal et appartenant à l'algèbre $\mathbb{C}[t][\delta]$. Pour cela, on procède par élimination de Gauß. On plonge à nouveau $\mathbb{C}[x]$ dans son corps des fractions $\mathbb{C}(x)$ puis on identifie $\mathbb{C}(x)$ au $\mathbb{C}(t)$ -espace vectoriel $\mathbb{C}(t)_{r-1}[x] \simeq \mathbb{C}(t)[x]/(x^r - t)$ de base $1, x, \dots, x^{r-1}$ (voir la remarque 2.3 ci-dessous). Le quotient de $\mathbb{C}(x)[\delta]$ par l'idéal engendré par D à droite est alors un $\mathbb{C}(t)$ -espace vectoriel de dimension finie nr engendré par les monômes $x^\ell \delta^k$ pour $0 \leq \ell \leq r-1$ et $0 \leq k \leq n-1$. Notons U_j le représentant de δ^j dans cette base. Une élimination de Gauss entre les U_j successifs, et ce jusqu'à $j = nr$ au plus, conduit à une relation de dépendance $\mathbb{C}(t)$ -linéaire $\sum L_j(t)U_j = 0$. L'opérateur $L = \sum L_j(x^r)\delta^j$ est un multiple de D invariant par \mathfrak{S}_r et d'ordre minimal. C'est donc, après suppression des dénominateurs et normalisation, l'opérateur L .

Algorithme : version 3.

1. L'opérateur D étant supposé de 0-poids positif ou nul, l'exprimer dans l'algèbre $\mathbb{A}_1 = \mathbb{C}[x][\delta]$:

$$D = b_n(x)\delta^n + b_{n-1}(x)\delta^{n-1} + \dots + b_0(x) \text{ et poser } t = x^r.$$

2. Remplacer D par l'opérateur unitaire $R = \delta^n + \frac{b_{n-1}(x)}{b_n(x)}\delta^{n-1} + \dots + \frac{b_0(x)}{b_n(x)}$ de $\mathbb{C}(x)[\delta]$ et exprimer les coefficients de ce dernier dans la base $(x^\ell)_{0 \leq \ell \leq r-1}$ du $\mathbb{C}(t)$ -espace vectoriel $\mathbb{C}(x)$.

3. Déterminer les représentants U_j des puissances successives δ^j de δ dans la base $(x^\ell \delta^k)_{\substack{0 \leq \ell \leq r-1 \\ 0 \leq k \leq n-1}}$ du $\mathbb{C}(t)$ -espace vectoriel $\mathbb{C}(x)[\delta]/(D)$.
4. Par un algorithme de Gauss dans ce $\mathbb{C}(t)$ -espace vectoriel et en chassant les dénominateurs déterminer la première relation $\sum L_j(t)U_j = 0$ de dépendance $\mathbb{C}[t]$ -linéaire entre les U_j .
5. $L = \sum L_j(x^r)\delta^j$ à normalisation près.

Remarque 2.1. Le point 2 de l'algorithme ci-dessus s'obtient par division euclidienne dans l'anneau $\mathbb{C}(t)[x]$. Nous détaillons cet algorithme dans la remarque (2.3), variante 1, ci-dessous. Ce calcul revient, de fait, à extraire les termes de r en r dans le développement en série d'une fraction rationnelle.

2.2 Calcul par l'intermédiaire des systèmes

Dans ce paragraphe nous conservons pour donnée initiale l'équation différentielle $Dy = 0$ et nous proposons des algorithmes qui opèrent sur son système compagnon. Ces mêmes algorithmes peuvent s'appliquer directement à un système sans réduction à une forme compagnon si telle était la donnée initiale.

2.2.1 Par saturation galoisienne directe. Considérons un système méromorphe quelconque (pas nécessairement sous forme compagnon) à l'origine

$$\Delta Y \equiv \frac{dY}{dx} - \frac{1}{x^{r+1}}A(x)Y = 0.$$

(A étant quelconque, r n'est pas en général le rang de Poincaré du système mais cette notation est plus simple pour la suite). L'action itérée de $\tilde{\omega} : x \mapsto \omega x$ donne naissance à r systèmes $\Delta_j Y \equiv \frac{dY}{dx} - \frac{1}{x^{r+1}}A(\omega^j x)Y$, $j = 0, 1, \dots, r-1$, où, si $Y(x)$ est une solution de $\Delta Y = 0$, on voit que $Y(\omega^j x)$ est solution de $\Delta_j Y = 0$ et le vecteur

inconnu $\tilde{Y} = \begin{bmatrix} Y(x) \\ Y(\omega x) \\ \vdots \\ Y(\omega^{r-1}x) \end{bmatrix}$ satisfait à l'équation matricielle de dimension nr

$$\tilde{\Delta} \tilde{Y} \equiv \frac{d\tilde{Y}}{dx} - \frac{1}{x^{r+1}}\tilde{A}(x)\tilde{Y} = 0 \quad \text{où} \quad \tilde{A}(x) = \bigoplus_{j=0}^{r-1} A(\omega^j x).$$

Supposons désormais que $\Delta Y = 0$ soit le système compagnon de l'équation (1) : $Dy = 0$.

Alors, à toute solution $\hat{f}(x)$ de (1) correspond la solution

$$\widehat{F}(x) = \begin{bmatrix} \hat{f}(x) \\ \hat{f}'(x) \\ \vdots \\ \hat{f}^{(r-1)}(x) \end{bmatrix}$$

de $\Delta Y = 0$ et les solutions

$$\widetilde{F}(x) = \begin{bmatrix} \lambda_0 \widehat{F}(x) \\ \lambda_1 \widehat{F}(\omega x) \\ \vdots \\ \lambda_{r-1} \widehat{F}(\omega^{r-1}x) \end{bmatrix}$$

de $\widetilde{\Delta Y} = 0$ pour des constantes complexes $\lambda_0, \lambda_1, \dots, \lambda_{r-1}$ arbitraires.

Considérons les vecteurs-lignes $E_1 = [1 \ 0 \ \dots \ 0]$ de dimension n et $\widetilde{E} = [E_1 \ E_1 \ \dots \ E_1]$ de dimension nr . La méthode du vecteur cyclique (cf. 5.3) appliquée au système $\widetilde{\Delta Y} = 0$ à partir du vecteur \widetilde{E} fournit une équation différentielle qui admet pour solution toutes les combinaisons linéaires possibles de $\hat{f}(x), \hat{f}(\omega x), \dots, \hat{f}(\omega^{r-1}x)$. Cet espace vectoriel est aussi engendré par les séries $\hat{f}^0(x), \hat{f}^1(x), \dots, \hat{f}^{r-1}(x)$ auxquelles nous nous intéressons. Par construction, la méthode fournit l'équation différentielle d'ordre minimal admettant pour solutions toutes celles de l'équation (1) et leurs orbites sous l'action du groupe de Galois \mathfrak{S}_r de $x^r = t$. On obtient donc ici encore la même équation $Ly = 0$ que celle obtenue par la méthode précédente où L est le ppcm des opérateurs D_0, D_1, \dots, D_{r-1} .

En particulier, et comme on le verra sur l'exemple de l'équation de Ramis–Sibuya, l'équation $Ly = 0$ n'est pas toujours d'ordre maximal nr ; autrement dit, le vecteur \widetilde{E} n'est pas nécessairement cyclique.

Ces observations nous conduisent à l'algorithme suivant :

Algorithme : version 4.

1. Écrire le système compagnon de l'équation (1), à savoir,

$$\Delta Y \equiv \frac{dY}{dx} - \frac{1}{x^{r+1}} A(x)Y$$

où

$$\frac{1}{x^{r+1}}A(x) = \begin{bmatrix} 0 & 1 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \vdots & & & 0 & 1 \\ \frac{-a_0}{a_n} & \cdots & \frac{-a_{n-2}}{a_n} & \frac{-a_{n-1}}{a_n} \end{bmatrix}.$$

2. Écrire le système somme directe

$$\tilde{\Delta}\tilde{Y} \equiv x^{r+1}\frac{d\tilde{Y}}{dx} - \tilde{A}(x)\tilde{Y} = 0 \quad \text{où} \quad \tilde{A}(x) = \bigoplus_{j=0}^{r-1} A(\omega^j x).$$

3. Appliquer l'algorithme du vecteur cyclique au système $\tilde{\Delta}\tilde{Y} = 0$ et au vecteur \tilde{E} , d'où l'obtention de l'opérateur L après normalisation éventuelle.

2.2.2 Par réduction du rang. Il s'agit, ici encore, d'une méthode par saturation galoisienne mais celle-ci est partie intégrante de la réduction du rang dont nous utilisons maintenant les propriétés.

Notons $\mathcal{Y}(x)$ une matrice fondamentale de solutions d'un système quelconque $\Delta Y = 0$. Le système r -réduit associé à $\Delta Y = 0$ est l'unique système de la variable

$$t = x^r, \text{ méromorphe en } 0 \text{ et admettant les solutions } \begin{bmatrix} \mathcal{Y}(x) \\ x^{-1}\mathcal{Y}(x) \\ \vdots \\ x^{-(r-1)}\mathcal{Y}(x) \end{bmatrix} \text{ pour un choix}$$

donné $x = t^{1/r}$ d'une racine $r^{\text{ème}}$ de t . Ce système admet pour matrice fondamentale de solutions la matrice

$$\mathbf{V}(t) = \begin{bmatrix} \mathcal{Y}(t^{1/r}) & \mathcal{Y}(\omega t^{1/r}) & \cdots & \mathcal{Y}(\omega^{r-1}t^{1/r}) \\ (t^{1/r})^{-1}\mathcal{Y}(t^{1/r}) & (\omega t^{1/r})^{-1}\mathcal{Y}(\omega t^{1/r}) & \cdots & (\omega^{r-1}t^{1/r})^{-1}\mathcal{Y}(\omega^{r-1}t^{1/r}) \\ \vdots & \vdots & & \vdots \\ (t^{1/r})^{-(r-1)}\mathcal{Y}(t^{1/r}) & (\omega t^{1/r})^{-(r-1)}\mathcal{Y}(\omega t^{1/r}) & \cdots & (\omega^{r-1}t^{1/r})^{-(r-1)}\mathcal{Y}(\omega^{r-1}t^{1/r}) \end{bmatrix}$$

pour un choix $x = t^{1/r}$ quelconque d'une racine $r^{\text{ème}}$ de t .

Pour $\ell = 0, 1, \dots, r-1$, notons $A^\ell(t) = \sum_m A_{\ell+rm} t^{-m}$ de sorte que

$$A(x) = A^0(x^r) + xA^1(x^r) + \cdots + x^{r-1}A^{r-1}(x^r).$$

Le système r -réduit associé à $\Delta Y = 0$ s'écrit ([L-R01], Theorem 1.1)

$$\tilde{\Delta}^r Y \equiv r t^2 \frac{dY}{dt} - \left(A(t) - \bigoplus_{j=0}^{r-1} j t I \right) Y = 0$$

où I désigne la matrice identité de dimension n et où

$$A(t) = \begin{bmatrix} A^0(t) & tA^{r-1}(t) & tA^{r-2}(t) & \cdots & \cdots & \cdots & tA^1(t) \\ A^1(t) & A^0(t) & tA^{r-1}(t) & \ddots & & & \vdots \\ A^2(t) & A^1(t) & A^0(t) & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & A^0(t) & tA^{r-1}(t) & tA^{r-2}(t) \\ \vdots & & & \ddots & A^1(t) & A^0(t) & tA^{r-1}(t) \\ A^{r-1}(t) & \cdots & \cdots & \cdots & A^2(t) & A^1(t) & A^0(t) \end{bmatrix}$$

Supposons maintenant, à nouveau, que $\Delta Y = 0$ soit le système compagnon de l'équation (1) : $Dy = 0$ admettant une solution formelle $\hat{f}(x) = \sum_{m \geq 0} u_m x^m$. Alors, compte tenu de la forme de la solution fondamentale V , le système $\tilde{\Delta}^r Y = 0$ admet, pour tout j , une solution dont la première composante est égale à $\sum_{m \geq 0} u_{j+mr} t^m$. Dans un souci d'homogénéité avec les sections précédentes, revenons à la variable $x = t^{1/r}$ et aux séries $\hat{f}^j(x) = \sum_{m \geq 0} u_{j+mr} x^{j+mr}$ en posant $X(x) = \left(\bigoplus_{j=0}^{r-1} x^j I \right) Y(x^r)$. On obtient le système

$$x^{r+1} \frac{dX}{dx} - B(x) X = 0 \quad (2)$$

où la matrice B s'écrit

$$B(x) = \begin{bmatrix} A^0(x^r) & x^{r-1} A^{r-1}(x^r) & \cdots & \cdots & x^2 A^2(x^r) & x A^1(x^r) \\ x A^1(x^r) & A^0(x^r) & \ddots & & & x^2 A^2(x^r) \\ x^2 A^2(x^r) & x A^1(x^r) & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & A^0(x^r) & x^{r-1} A^{r-1}(x^r) \\ x^{r-1} A^{r-1}(x^r) & \cdots & \cdots & \cdots & x A^1(x^r) & A^0(x^r) \end{bmatrix}$$

Ce système admet, pour $j = 0, 1, \dots, r-1$, une solution dont la première composante est égale à $\hat{f}^j(x)$. Le résultat est d'ailleurs vrai pour toute composante d'indice $1 \bmod n$. On peut alors proposer l'algorithme suivant :

Algorithme : version 5.

1. Écrire le système compagnon de l'équation (1) à savoir

$$x^{r+1} \frac{dY}{dx} = A(x)Y \quad \text{où} \quad A(x) = x^{r+1} \begin{bmatrix} 0 & 1 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \vdots & & & 0 & 1 \\ \frac{-a_0}{a_n} & \cdots & \frac{-a_{n-2}}{a_n} & \frac{-a_{n-1}}{a_n} \end{bmatrix}.$$

2. Écrire le système r -réduit en x associé (2) dont la matrice $\mathbf{B}(x)$ est donnée ci-dessus.
3. Écrire l'équation différentielle obtenue par l'algorithme du vecteur cyclique à partir du vecteur initial $\tilde{E} = (1, 0, \dots, 0)$ (ou $\tilde{E} = (\underbrace{0, \dots, 0}_{kn \text{ termes}}, 1, 0, \dots)$ pour n'importe quel $k < r$) d'où l'opérateur L après normalisation éventuelle.

Remarque 2.2. Cette approche montre que l'algorithme s'applique également aux séries formelles-logarithmiques solutions mais non, en général, à celles qui contiennent des exponentielles ([L-R01] Prop 3.3).

En effet, considérons un système général $\Delta Y = 0$ et supposons que n' colonnes d'une de ses solutions fondamentales formelles s'écrivent $\hat{H}(x)x^M$ où la matrice \hat{H} de taille $n \times n'$ est à coefficients séries formelles en x et où M est une matrice constante de taille $n' \times n'$.

La restriction de V à ses n premières lignes contient les colonnes

$$\hat{H}(\omega^j x)(\omega^j x)^M = \hat{H}(\omega^j x)x^M \omega^{jM} \quad \text{pour } j = 0, 1, \dots, r-1.$$

Ainsi l'espace vectoriel qu'elles engendrent contient toutes les combinaisons linéaires des colonnes $\hat{H}(\omega^j x)x^M$ pour tout j et en particulier, si l'on note comme précédemment \hat{H}^j les sous-séries des termes de r en r extraites de \hat{H} , les colonnes $\hat{H}^j(x)x^M$ pour $j = 0, 1, \dots, r-1$. En revanche, si le système $\Delta Y = 0$ admet des solutions de la forme $\hat{H}(x)x^M e^{Q(1/x)}$ alors $Q(1/(\omega^j x)) = Q(1/x)$ seulement si $Q(1/x)$ est un polynôme en $1/x^r$ ce qui n'est pas le cas en général. Ainsi, on ne peut pas en général réaliser les combinaisons linéaires souhaitées sur \hat{H} .

Remarque 2.3. Nous précisons ici le point 2 de l'algorithme précédent (version 5). Lorsque l'équation de départ (1) : $Dy = 0$ est à coefficients polynomiaux son système

compagnon est, en général, à coefficients fractions rationnelles en x . Pour construire la matrice $\mathbf{B}(x)$ il s'agit d'extraire des coefficients de la matrice compagnon A les sous-séries des termes de r en r . Deux méthodes naturelles se présentent pour extraire les sous-séries des termes de r en r d'une série définie comme fraction rationnelle.

Variante 1 : L'une des méthodes consiste en un algorithme de division euclidienne dans l'anneau $\mathbb{C}(T)[X]$ après avoir remarqué que la décomposition cherchée d'un élément $a(x) \in \mathbb{C}(x)$ sous la forme

$$a(x) = a^0(t) + a^1(t)x + \cdots + a^{r-1}(t)x^{r-1} \quad \text{où } t = x^r$$

traduit l'isomorphisme canonique entre $\mathbb{C}(x)$ et $\mathbb{C}(T)[X]/(X^r - T)$.

Notons $a = \frac{p}{q}$ une décomposition irréductible de a . On peut voir p et q comme des éléments de $\mathbb{C}[X]$ et donc aussi de $\mathbb{C}(T)[X]$.

Si $q \equiv 1$ la division euclidienne de $p(X)$ par $X^r - T$ dans $\mathbb{C}(T)[X]$ fournit une identité

$$p(X) = Q(T, X)(X^r - T) + R(T, X) \quad \text{avec } \deg_X R \leq r - 1.$$

Si $q \not\equiv 1$ on remarque que $q(X)$ et $X^r - T$ sont premiers entre eux. On applique l'algorithme d'Euclide pour établir l'identité de Bézout

$$Uq + V(X^r - T) = 1 \quad \text{avec } U \text{ et } V \in \mathbb{C}(T)[X]$$

et on effectue la division euclidienne de pU par $X^r - T$ d'où une identité

$$p(X)U(T, X) = Q(T, X)(X^r - T) + R(T, X) \quad \text{avec } \deg_X R \leq r - 1.$$

Dans les deux cas $R(t, x)$ fournit la décomposition de $a(x)$ cherchée.

Variante 2 :

Une autre méthode s'appuie sur la remarque suivante.

Le système $\frac{d\mathbf{X}}{dx} = \mathbf{B}(x)\mathbf{X}$ admet la solution fondamentale $U_n(\omega) \left(\bigoplus_{j=0}^{r-1} \mathcal{Y}(\omega^j x) \right)$ où

$$U_n(\omega) = \begin{bmatrix} I_n & I_n & \cdots & I_n \\ I_n & \omega I_n & \cdots & \omega^{r-1} I_n \\ \vdots & \vdots & & \vdots \\ I_n & \omega^{r-1} I_n & \cdots & \omega^{(r-1)(r-1)} I_n \end{bmatrix} \quad \text{est une matrice de Van der Monde par}$$

blocs construite sur $I_n, \omega I_n, \omega^2 I_n, \dots, \omega^{r-1} I_n$. On passe donc du système $\tilde{\Delta}\tilde{Y} \equiv \frac{d\tilde{Y}}{dx} - \frac{1}{x^{r+1}}\tilde{A}(x)\tilde{Y} = 0$ où $\tilde{A}(x) = \bigoplus_{j=0}^{r-1} A(\omega^j x)$ au système $\frac{d\mathbf{X}}{dx} = \mathbf{B}(x)\mathbf{X}$ par la transformation de jauge constante $\tilde{Y} = U_n(\omega)\mathbf{X}$. D'où, un possible calcul de $\mathbf{B}(x)$

par la formule

$$\begin{aligned} \mathbf{B}(x) &= U_n(\omega)^{-1} \tilde{A}(x) U_n(\omega) \\ &= \frac{1}{r} U_n\left(\frac{1}{\omega}\right) \tilde{A}(x) U_n(\omega) \end{aligned}$$

Notons que cette formule très simple nécessite d'introduire le nombre algébrique ω que l'on avait jusque là évité dans cette version 5 de l'algorithme.

3 Approche duale dans l'espace des opérateurs aux différences

Nous nous posons ici la même question qu'au paragraphe précédent mais sous sa forme duale : *Comment obtenir des équations aux différences auxquelles satisfont les suites des coefficients des séries $\hat{f}^j(x)$ à partir d'une équation aux différences à laquelle satisfait la suite des coefficients de la série $\hat{f}(x)$?*

Nous notons $\mathbb{A}_1^* = \mathbb{C}[m][\tau]$ l'algèbre des opérateurs aux différences à coefficients polynomiaux où m est la variable ($m \in \mathbb{Z}$ est l'indice des suites si l'on fait opérer \mathbb{A}_1^* dans l'espace $\mathbb{C}^{\mathbb{Z}}$ des suites complexes et $m \in \mathbb{R}$ ou \mathbb{C} si l'on fait opérer \mathbb{A}_1^* dans l'espace des fonctions complexes de variable réelle ou complexe) et où τ est l'opérateur de décalage arrière ($\tau(u_m) = u_{m-1}$) ou l'opérateur de translation de 1 à droite ($\tau f(m) = f(m-1)$). Dans cette algèbre les relations de commutation sont engendrées par l'unique relation

$$m\tau - \tau m = \tau.$$

La transformation de Mellin $\mathcal{M} : \mathbb{A}_1 \rightarrow \mathbb{A}_1^*$ définie par

$$\mathcal{M}(x) = \tau \quad \text{et} \quad \mathcal{M}(\delta) = m.$$

établit un isomorphisme d'algèbres entre \mathbb{A}_1 et \mathbb{A}_1^* . On notera qu'en particulier, elle fait correspondre de part et d'autre les relations de commutation

$$\delta x - x\delta = x \quad \text{et} \quad m\tau - \tau m = \tau.$$

À un opérateur différentiel $D = \sum_{\substack{0 \leq i \leq p \\ 0 \leq k \leq n}} b_{i,k} x^i \delta^k$ d'ordre n et de degré p elle fait correspondre l'opérateur aux différences $D^* = \sum_{\substack{0 \leq i \leq p \\ 0 \leq k \leq n}} b_{i,k} \tau^i m^k$ d'ordre p et de degré n que l'on peut réécrire sous la forme $D^* = \sum_{0 \leq i \leq p} c_i(m) \tau^i$ grâce à la relation de commutation. À l'équation différentielle (1) correspond ainsi, par transformation de Mellin, l'équation aux différences

$$D^* y = 0. \tag{1^*}$$

Si une série $f(x) = \sum_{m \geq 0} u_m x^m$ est solution de l'équation (1) alors la suite de ses coefficients $(u_m)_{m \geq 0}$ est solution de l'équation (1*) avec conditions initiales adéquates et réciproquement.

Les procédures de passage de D à D^* et vice-versa sont élémentaires. On en trouve des implantations dans [BCL] et dans *Gfun* ([SZ94]).

Les méthodes du paragraphe précédent se transposent dans ce contexte. Nous les évoquons rapidement.

3.1 Saturation galoisienne de l'équation aux différences

Comme \mathbb{A}_1 , l'algèbre duale \mathbb{A}_1^* se plonge de façon canonique dans l'algèbre $\mathbb{C}(m)[\tau]$ et celle-ci étant euclidienne à droite et à gauche on peut y calculer le ppcm des opérateurs déduits de D^* par l'action itérée du générateur $\check{\omega}^* : \tau \rightarrow \omega\tau$ du groupe de Galois \mathfrak{S}_r^* de l'équation $\tau^r = \sigma$.

Algorithme : version 1*.

1. Pour $j = 0, 1, \dots, r - 1$, déterminer l'opérateur D_j^* en substituant $\omega^j \tau$ à τ dans l'opérateur aux différences D^* .
2. Déterminer un plus petit commun multiple à gauche dans l'algèbre $\mathbb{C}(m)[\tau]$ des opérateurs $D_0^*, D_1^*, \dots, D_{r-1}^*$.
3. En déduire son unique représentant primitif normalisé $\Lambda = \sum_{0 \leq j \leq r} \Lambda_j(m) \tau^{rj}$ à coefficients dans \mathbb{A}_1^* .

On peut ici encore utiliser la procédure `skew_gcdex` du paquetage `Ore_algebra` de MAPLE pour déterminer le ppcm des opérateurs D_j^* (point 2.).

3.2 Élimination polynomiale

• **Première variante :**

On peut comme dans l'algorithme, version 2, traduire l'action du groupe de Galois par l'introduction d'une indéterminée Ω et réaliser la saturation galoisienne en éliminant Ω par un calcul de pgcd à droite dans l'algèbre $F_g^*[\Omega]$ euclidienne à droite et à gauche.

Algorithme : version 2*.

1. Déterminer le polynôme $P^* \in \mathbb{A}_1^*[\Omega]$ en substituant $\Omega\tau$ à τ dans l'opérateur D^* .
2. Par un algorithme d'Euclide dans l'algèbre $F_g^*[\Omega]$ déterminer la relation de Bézout

$$S^*(\Omega)P^*(\Omega) + T^*(\Omega)(\Omega^r - 1) = v\Lambda$$

dans laquelle S^* et de T^* appartiennent à $\mathbb{A}_1^*[\Omega]$, leurs coefficients sont des polynômes de \mathbb{A}_1^* dont les contenus sont premiers entre eux, $v \in \mathbb{C}[m]$ et Λ est normalisé.

• **Deuxième variante : produit de Hadamard.**

Rappelons que le produit de Hadamard de deux séries est le produit terme à terme de leurs coefficients. Les sous-séries $\hat{f}^j(x) = \sum_{m \geq 0} u_{j+mr} x^{j+mr}$ des termes de r en r d'une série $\hat{f}(x) = \sum_{m \geq 0} u_m x^m$ sont donc les produits de Hadamard de \hat{f} par les séries $\frac{x^j}{1-x^r} = \sum_{m \geq 0} x^{j+mr}$.

Des algorithmes généraux de calcul de produits de Hadamard par élimination existent. On trouvera par exemple, dans le paquetage *Gfun* ([SZ94] de MAPLE, une procédure **hadamardproduct** qui, étant donnés deux relations de récurrence vérifiées par des séries \hat{f} et \hat{g} respectivement, restitue une relation de récurrence vérifiée par le produit de Hadamard $\hat{f} \odot \hat{g}$ de ces séries. Dans le cas particulier qui nous occupe on peut en donner une version spécifique ; il s'agit, dans ce cas, de construire un multiple à gauche de l'opérateur aux différences $D^* \in \mathbb{A}^*$ qui s'exprime en fonction des puissances de τ^r . Pour cela, on peut observer que le quotient de $\mathbb{C}(m)[\delta]$ par D^* à droite est un $\mathbb{C}(m)$ -espace vectoriel de dimension finie p , de base $(\tau^j)_{0 \leq j \leq p-1}$, et chercher une relation de dépendance $\mathbb{C}(m)$ -linéaire entre les représentants des puissances successives de τ^r . Dans tous les cas, il suffit d'aller jusqu'à τ^{pr} .

Algorithme : version 3*.

1. Remplacer D^* par l'opérateur unitaire $R^* = \sum_{0 \leq i \leq p} \frac{b_i(m)}{b_p(m)} \tau^i \in \mathbb{C}(m)[\tau]$.
2. Pour $j \in \mathbb{N}$ variant de 0 à p (au plus) déterminer l'unique représentant $V_j \in \mathbb{C}(m)[\tau]$ de degré $\leq p-1$ de τ^{jr} modulo R^* à droite.
3. Par un algorithme de Gauss dans le $\mathbb{C}(m)$ -espace vectoriel $\mathbb{C}(m)[\delta]/(R^*)$ déterminer, en supprimant les dénominateurs, la première relation

$$\sum_{0 \leq j \leq p} \Lambda_j(m) V_j = 0$$

de dépendance linéaire entre les V_j à coefficients $\Lambda_j \in \mathbb{C}[m]$.

4. $\Lambda = \sum_{0 \leq j \leq p} \Lambda_j(m) \tau^{rj}$ est, après normalisation éventuelle, l'opérateur aux différences cherché.

3.3 Saturation galoisienne du système compagnon

Un opérateur aux différences admet une forme compagnon analogue à celle d'une équation différentielle. Sous l'action du générateur $\check{\omega}^* : \tau \rightarrow \omega\tau$ du groupe de galois \mathfrak{S}_r^* un système $(\tau - C^*(m))Y = 0$ devient $(\tau - \omega^{-1}C^*(m))Y = 0$ et une solution $(u_m)_m$ devient $(\omega^{-m}u_m)_m$. La version qui suit est un analogue de la version 4. On note

$E_1 = [1 \ 0 \ \cdots \ 0]$ le vecteur-ligne représentant la première forme coordonnée dans \mathbb{C}^p et \tilde{E} le vecteur ligne $\tilde{E} = [E_1 \ E_1 \ \cdots \ E_1]$ de dimension rp .

Algorithme : version 4*.

1. Écrire le système compagnon de l'équation (1*), à savoir,

$$\Delta^* Y \equiv \tau Y - C^*(m)Y = 0$$

$$\text{où } C^*(m) = \begin{bmatrix} 0 & 1 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \vdots & & & 0 & 1 \\ \frac{-c_0}{c_p} & \cdots & & \frac{-c_{p-2}}{c_p} & \frac{-c_{p-1}}{c_p} \end{bmatrix}.$$

2. Écrire le système somme directe

$$\tilde{\Delta}^* \tilde{Y} \equiv \tau \tilde{Y} - \tilde{C}^*(m) \tilde{Y} = 0 \quad \text{où} \quad \tilde{C}^*(m) = \bigoplus_{j=0}^{r-1} \omega^j C^*(m).$$

3. L'algorithme du vecteur cyclique appliqué au système $\tilde{\Delta}^* \tilde{Y} = 0$ et au vecteur \tilde{E} conduit à l'opérateur aux différences Λ .

3.4 Réduction du rang

Étant donné un système aux différences d'ordre un et de dimension p

$$\Delta^* Y \equiv (\tau - C^*(m))Y = 0 \tag{1}$$

le système d'ordre r et de même dimension p

$$(\tau^r - B^*(m))Y = 0 \tag{2}$$

où

$$B^*(m) = (\tau^{r-1} C^*(m))(\tau^{r-2} C^*(m)) \cdots (\tau C^*(m)) C^*(m)$$

admet pour solutions toutes celles de $\Delta^* Y = 0$ et il est invariant sous l'action de \mathfrak{S}_r^* . C'est donc un analogue du système de rang réduit pour les systèmes différentiels.

Algorithme : version 5*.

1. Écrire le système compagnon de l'équation (1*), à savoir,

$$\Delta^* Y \equiv \tau Y - C^*(m)Y = 0$$

où

$$C^*(m) = \begin{bmatrix} 0 & 1 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \vdots & & & 0 & 1 \\ \frac{-c_0}{c_p} & \cdots & & \frac{-c_{p-2}}{c_p} & \frac{-c_{p-1}}{c_p} \end{bmatrix}.$$

2. Écrire le système r -réduit $\widetilde{\Delta}^* Y \equiv \tau^r Y - \mathbf{B}^*(m)Y = 0$ associé dont la matrice $\mathbf{B}^*(m)$ est donnée par

$$\mathbf{B}^*(m) = C^*(m-r+1)C^*(m-r+2) \cdots C^*(m-1)C^*(m).$$

3. L'algorithme du vecteur cyclique appliqué au système $\widetilde{\Delta}^* Y = 0$ à partir du vecteur initial $\widetilde{E} = (1, 0, \dots, 0)$ conduit à l'opérateur

$$\Lambda = \sum_{0 \leq j \leq p} \Lambda_j(m) \tau^{rj}.$$

Remarque 3.1. On a utilisé ici un système r -réduit de même dimension p que le système initial. Comme dans le cas différentiel, on aurait pu introduire un système r -réduit de dimension pr mais il faut noter qu'on ne peut ici introduire une puissance $r^{\text{ème}}$ du décalage $\sigma = \tau^{1/r}$ comme on avait introduit la racine $t = x^{1/r}$ de la variable x . Au lieu du système (2) on utilise quelquefois le système de dimension pr

$$\tau \mathbf{Z} = \mathbf{C}^* \mathbf{Z} \quad \text{où} \quad \mathbf{C}^* = \begin{bmatrix} 0 & 0 & \cdots & 0 & A \\ A & 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & A & 0 \end{bmatrix} \quad (3)$$

Si la suite $(u_m)_m$ est solution de (1) la suite $\begin{bmatrix} (u_{mr})_m \\ (u_{mr-1})_m \\ \vdots \\ (u_{mr-r+1})_m \end{bmatrix}$ est solution de (3) et,

si on partitionne le vecteur inconnu $\mathbf{Z} = \begin{bmatrix} Z_0 \\ \vdots \\ Z_{r-1} \end{bmatrix}$ en r vecteurs de dimension p , on

obtient les équations
$$\left\{ \begin{array}{l} \tau Z_0 = AZ_{r-1} \\ \tau Z_1 = AZ_0 \\ \vdots \\ \tau Z_{r-1} = AZ_{r-2} \end{array} \right. .$$
 En particulier, chaque vecteur Z_j est solution du système r -réduit (2).

4 Comparaison des diverses méthodes et exemple

4.1 Relation entre L et Λ

De façon globale, l'approche directe et l'approche duale se correspondent par l'isomorphisme de Mellin \mathcal{M} qui échange les coordonnées. Cependant, les opérateurs L et Λ auxquels conduit chacune de ces deux approches ne se correspondent pas par cet isomorphisme car une permutation de l'ordre et du rôle des coordonnées n'est pas compatible avec les opérations effectuées. Ceci est lié au fait que les divisions, `pgcdd`, `ppcmg`,... sont définis dans l'algèbre $\mathbb{C}(x)[\delta]$ à coefficients rationnels alors qu'on fait agir l'isomorphisme de Mellin par permutation des coordonnées sur l'algèbre $\mathbb{A}_1 = \mathbb{C}[x][\delta]$ à coefficients polynomiaux. Les relations arithmétiques établies dans $\mathbb{C}(x)[\delta]$ induisent des relations analogues dans \mathbb{A}_1 en supprimant les dénominateurs. Ce faisant on remplace divisions par pseudo-divisions au sens des algèbres de Ore et on introduit des facteurs supplémentaires dont il faut tenir compte. Expliquons plus en détail :

L'opérateur différentiel L est un `ppcmg` dans l'algèbre $\mathbb{C}(x)[\delta]$ mais appartenant à $\mathbb{A}_1 = \mathbb{C}[x][\delta]$, des opérateurs différentiels D_j (*cf.* Algorithme : version 1). Il vérifie donc des relations de la forme

$$S_j D_j = u(x^r) L \quad \text{pour } j = 0, 1, \dots, r-1 \quad (1)$$

où $S_j \in \mathbb{A}_1$ et où $u(x^r) \in \mathbb{C}[x^r]$.

L'opérateur aux différences Λ est un `ppcmg`, appartenant à \mathbb{A}_1^* , des opérateurs D_j^* transformés de Mellin des opérateurs D_j . Il vérifie donc des relations de la forme

$$G_j D_j^* = v(m) \Lambda \quad \text{pour } j = 0, 1, \dots, r-1 \quad (2)$$

où $G_j \in \mathbb{A}_1^*$ et où $v(m) \in \mathbb{C}[m]$.

Par transformation de Mellin inverse, les relations (2) deviennent

$$\tilde{G}_j D_j = v(\delta) \tilde{\Lambda} \quad \text{pour } j = 0, 1, \dots, r-1$$

et, en les comparant à (1), on voit que c'est $v(\delta) \tilde{\Lambda}$ et non pas $\tilde{\Lambda}$ lui-même en général qui est un multiple à gauche dans $\mathbb{C}(x)[\delta]$ de l'opérateur L . A fortiori, comme on le constate sur l'exemple de l'équation de Ramis–Sibuya ci-dessous, $\tilde{\Lambda}$ n'est pas égal à

L en général. En supprimant les dénominateurs et en remarquant que $\tilde{\Lambda}$ et L sont des polynômes en x^r et δ on en déduit une relation de la forme

$$u_1(x^r)v(\delta)\tilde{\Lambda} = Q(x^r, \delta)L \quad (3)$$

où $u_1(x^r) \in \mathbb{C}[x^r]$, $v(\delta) \in \mathbb{C}[\delta]$ et où $Q(x, \delta) \in \mathbb{A}_1$ et la relation duale

$$u_1(\tau^r)v(m)\Lambda = Q(\tau^r, m)L^*. \quad (4)$$

À partir de la transformée de Mellin de la relation (1), on obtient, de même, des relations du type $v_1(m)u(\tau^r)L^* = P(m, \tau^r)\Lambda$ et $v_1(\delta)u(x^r)L = P(\delta, x^r)\tilde{\Lambda}$.

4.2 Exemple de l'équation de Ramis–Sibuya

L'équation de Ramis–Sibuya est l'équation $Dy = 0$ où l'opérateur D s'écrit

$$D = a_3(x)\partial^3 + a_2(x)\partial^2 + a_1(x)\partial + a_0(x) \quad (5)$$

avec

$$\begin{cases} a_3(x) = x^6(3x^3 - 10x^2 - 2x - 4) \\ a_2(x) = x^3(12x^5 - 47x^4 - 16x^3 - 50x^2 - 8x - 8) \\ a_1(x) = 2x(3x^6 - 14x^5 - 12x^4 - 5x^3 - 14x^2 - 6x - 4) \\ a_0(x) = 12x^4 - 14x^3 + 60x^2 + 12x + 8 \end{cases}$$

Cette équation admet pour solutions les deux exponentielles $e^{1/x}$ et e^{1/x^2} et la série de Ramis–Sibuya $\hat{f}(x) = \hat{g}(x) + \hat{g}(x^2)$ où \hat{g} désigne la série d'Euler $\hat{g}(x) = \sum_{n \geq 0} (-1)^n n! x^{n+1}$.

Nous nous proposons d'appliquer les algorithmes précédents (directs et duaux) à cette équation.

Nous nous plaçons, pour l'instant, dans le cas $r = 2$.

L'opérateur L calculé par les algorithmes du paragraphe 2 est l'opérateur

$$\begin{aligned} L_5 = & x^9(3x^4 + 86x^2 + 168)\partial^5 + 2x^6(21x^6 + 691x^4 + 1598x^2 + 168)\partial^4 \\ & + x^5(162x^6 + 6061x^4 + 16226x^2 + 1848)\partial^3 \\ & + 2x^2(90x^8 + 3816x^6 + 12079x^4 + 250x^2 - 168)\partial^2 \\ & + 12x(3x^8 + 144x^6 + 642x^4 - 12x^2 + 112)\partial \\ & - 24(15x^4 + 50x^2 + 56) \end{aligned}$$

ou, si on l'exprime en fonction de l'opérateur d'Euler δ ,

$$\begin{aligned} L_5 = & x^4 (3x^4 + 86x^2 + 168) \delta^5 + 2x^2 (6x^6 + 261x^4 + 758x^2 + 168) \delta^4 \\ & + x^2 (15x^6 + 779x^4 + 2930x^2 - 168) \delta^3 \\ & + (6x^8 + 351x^6 + 2236x^4 - 1348x^2 - 336) \delta^2 \\ & + (-10x^6 + 854x^4 + 1036x^2 + 1680) \delta \\ & - 360x^4 - 1200x^2 - 1344. \end{aligned}$$

C'est un opérateur d'ordre 5 et non pas d'ordre maximal $nr = 6$ ce qui fournit un exemple de vecteur \tilde{E} non cyclique, comme annoncé au paragraphe 2.2.1. L'opérateur est de degré 13 si on l'exprime en fonction de ∂ mais de degré 8 seulement si on le développe suivant les puissances de δ .

L'opérateur aux différences Λ calculé au paragraphe 3 est l'opérateur

$$\begin{aligned} \Lambda_{10} = & 9(m-8)(m-10)^2(m-9)^2(16m^5 - 296m^4 \\ & + 2058m^3 - 7601m^2 + 20271m - 33468) \tau^{10} \\ & - (1792m^9 - 87200m^8 + 1836352m^7 - 22004044m^6 \\ & + 166720060m^5 - 847108525m^4 + 3019206360m^3 \\ & - 7738806953m^2 + 13424517510m - 11698417512)(m-8) \tau^8 \\ & + (-1216m^{10} + 55392m^9 - 1079288m^8 + 11719164m^7 \\ & - 77174132m^6 + 309642768m^5 - 659015878m^4 - 98207604m^3 \\ & + 5491902246m^2 - 16879573428m + 19197313776) \tau^6 \\ & + (-256m^{10} + 7936m^9 - 77344m^8 - 103744m^7 + 8452288m^6 \\ & - 81663212m^5 + 421470752m^4 - 1353600540m^3 \\ & + 2746004600m^2 - 3157587760m + 1367558400) \tau^4 \\ & + (-512m^9 + 18944m^8 - 295744m^7 + 2553728m^6 \\ & - 13375840m^5 + 43086584m^4 - 78262880m^3 \\ & + 47433816m^2 + 81784848m - 130369344) \tau^2 \\ & + 32(m-1)(m-4)(16m^5 - 456m^4 \\ & + 5066m^3 - 28333m^2 + 86123m - 126126). \end{aligned}$$

C'est un opérateur d'ordre 10 en τ (5 en τ^2) et de degré 10.

Son transformé de Mellin inverse est l'opérateur d'ordre 10 (et non pas 5 comme L_5)

$$\begin{aligned}
\tilde{\Lambda}_{10} = & 16x^4(3x^3 + 10x^2 - 2x + 4)(3x^3 - 10x^2 - 2x - 4)\delta^{10} \\
& + 8x^2(639x^8 - 5228x^6 - 2196x^4 - 288x^2 - 64)\delta^9 \\
& + 2x^2(37413x^8 - 192160x^6 - 29020x^4 + 12016x^2 + 4864)\delta^8 \\
& + (575667x^{10} - 1639604x^8 + 182652x^6 + 26304x^4 - 66368x^2 + 512)\delta^7 \\
& + (2519145x^{10} - 3228668x^8 + 1194340x^6 \\
& \quad - 201280x^4 + 190976x^2 - 17152)\delta^6 \\
& + (6379893x^{10} - 2007731x^8 - 1528848x^6 \\
& \quad - 943852x^4 - 118880x^2 + 237120)\delta^5 \\
& + (9089109x^{10} + 2101448x^8 - 2215606x^6 \\
& \quad + 2098096x^4 - 1103560x^2 - 1775584)\delta^4 \\
& + (6651108x^{10} + 3375593x^8 + 2148924x^6 \\
& \quad + 5161060x^4 + 5571168x^2 + 7937664)\delta^3 \\
& + (1928556x^{10} + 970506x^8 + 436398x^6 \\
& \quad - 19150552x^4 - 12389544x^2 - 21442336)\delta^2 \\
& + (-234120x^8 - 545820x^6 + 33655632x^4 + 17110512x^2 + 31203904)\delta \\
& + 162000x^6 - 19817280x^4 - 11638080x^2 - 16144128.
\end{aligned}$$

La relation (3) est vérifiée par $\tilde{\Lambda}_{10}$ et L_5 en prenant

- $u_1(x^2) = 243x^{20} + 34830x^{18} + 2064960x^{16} + 65046960x^{14}$
 $+ 1163615280x^{12} + 11771075936x^{10} + 65162455680x^8$
 $+ 203987266560x^6 + 362640015360x^4 + 342535495680x^2$
 $+ 133827821568,$
- $v(\delta) = 1,$
- $Q(x^2, \delta) = 16(3x^3 - 10x^2 - 2x - 4)$
 $(3x^3 + 10x^2 - 2x + 4)(3x^4 + 86x^2 + 168)^4\delta^5$
 $- 8(459x^{10} - 13002x^8 - 278300x^6 - 493880x^4$
 $- 521792x^2 - 153216)(3x^4 + 86x^2 + 168)^3\delta^4$

$$\begin{aligned}
& + 2(13365 x^{14} - 1993644 x^{12} - 49787448 x^{10} \\
& \quad - 413971040 x^8 - 856551216 x^6 - 2454506176 x^4 \\
& \quad - 2982352128 x^2 - 1143862272)(3 x^4 + 86 x^2 + 168)^2 \delta^3 \\
& - (261711 x^{18} - 38399238 x^{16} - 826873704 x^{14} \\
& \quad - 8277404112 x^{12} - 71496360592 x^{10} - 282382737504 x^8 \\
& \quad - 2419380097792 x^6 - 5180973643776 x^4 \\
& \quad - 5061866471424 x^2 - 2149514551296)(3 x^4 + 86 x^2 + 168) \delta^2 \\
& + (4644459 x^{22} + 118731420 x^{20} + 9162912348 x^{18} \\
& \quad + 97916779440 x^{16} - 4033206784368 x^{14} - 82159591566528 x^{12} \\
& \quad - 550884952288448 x^{10} - 2075017277585152 x^8 \\
& \quad - 3910594860533760 x^6 - 4248923686576128 x^4 \\
& \quad - 2981326231683072 x^2 - 1097681283514368) \delta \\
& - 8084610 x^{22} - 299382804 x^{20} - 7863071760 x^{18} \\
& \quad + 892227121920 x^{16} + 27828286253280 x^{14} \\
& \quad + 328935359770560 x^{12} + 1797276633574912 x^{10} \\
& \quad + 5352759561623040 x^8 + 8114867619102720 x^6 \\
& \quad + 7819726953185280 x^4 + 4892526662123520 x^2 \\
& \quad + 1607539792674816.
\end{aligned}$$

Le transformé de Mellin L_5^* de L_5 est l'opérateur

$$\begin{aligned}
L_5^* = & 3(m-6)(m-7)^2(m-8)^2 \tau^8 \\
& + (m-6)(86m^4 - 1542m^3 + 9959m^2 - 26925m + 24632) \tau^6 \\
& + (168m^5 - 1844m^4 + 5554m^3 + 5092m^2 - 49450m + 60544) \tau^4 \\
& + (336m^4 - 2856m^3 + 7724m^2 - 6340m - 1944) \tau^2 \\
& - 336(m-1)(m-4).
\end{aligned}$$

C'est un opérateur d'ordre 8. Il n'est donc pas multiple de l'opérateur Λ_{10} qui est d'ordre 10. De même que $\tilde{\Lambda}_{10}$ et L_5 vérifient la relation (3), les opérateurs L_5^* et Λ_{10} vérifient la relation (4).

On trouvera, écrites en MAPLE, les procédures correspondant à chacun des algorithmes présentés ci-dessus à l'adresse électronique indiquée dans [BCL]. À ce jour, celles-ci n'ont pas été optimisées et nous n'avons pas évalué leur complexité. Leur

exécution sur un certain nombre d'exemples donne une idée sans doute assez bonne de celles qui sont à retenir et à développer en priorité.

À titre indicatif, nous donnons ci-dessous les temps de calcul obtenus sur un ordinateur de type Alpha EV6.7 cadencé à 667 MHz pour chaque méthode et pour des valeurs de r allant de 2 à 10 sur l'exemple de l'équation de Ramis–Sibuya. La notation $A5(2)$ signifie «Algorithme : version 5 et variante 2». Les étoiles * indiquent un temps de calcul supérieur à 1 400 s et les doubles étoiles ** une mémoire saturée à 128 Mb. Dans le cas de l'algorithme $A2^*$ et pour $r = 3$ la saturation de la mémoire est obtenue après un temps de calcul de 230 s seulement.

Méthodes directes :

r	2	3	4	5	6	7	8	9	10
A1	0.9	51	151	*					
A2	2.4	*							
A3	.6	2.3	3.0	9.0	8.9	28.7	21.5	196.9	50.0
A4	.4	45.2	26.0	*					
A5(1)	.5	1.9	2.8	9.6	9.8	41.0	28.0	231.0	71.3
A5(2)	.5	4.0	6.6	89.0	56.0	*			

Méthodes duales :

r	2	3	4	5	6	7	8	9	10
A1*	1.5	**							
A2*	2.0	**							
A3*	.5	2.4	2.4	12.5	10.2	43.0	30.2	123.0	71.0
A4*	.8	121.2	1050.0	*					
A5*(1)	.3	1.6	1.5	6.4	4.7	19.1	11.6	47.1	25.0
A5*(2)	.9	4.6	6.0	30.1	25.3	150	88	642.0	341.3

Les meilleures méthodes apparaissent être ici sans conteste celles qui utilisent la réduction du rang sans recourir à l'emploi de nombres algébriques ($A5(1)$ et sa duale $A5^*(1)$) ou celles qui reposent sur le produit de Hadamard ($A3^*$ ou sa duale $A3$). Notons que l'écart entre $A3$ et $A5(1)$ par exemple n'est pas significatif. Ainsi la procédure $A5(1)$ écrite en utilisant les procédures générales d'élimination de Gauss disponibles sous MAPLE sans tenir compte de la forme spécifique des systèmes de rang réduit n'est certainement pas optimale. À titre anecdotique nous pouvons d'ailleurs ajouter que le même algorithme $A3$ exécuté sur une machine équipée d'un processeur de type Pentium cadencé à 2,2 GHz s'est avéré deux fois plus rapide sauf dans le cas $r = 9$ où il s'est avéré 5 fois plus lent !

4.3 Remarque sur l'utilisation d'équations différentielles linéaires affines

Nous avons choisi de travailler avec des équations différentielles linéaires. On peut tout aussi bien travailler avec des équations différentielles linéaires affines («avec second membre»).

Nous nous contentons ici d'observer l'exemple de la série de Ramis–Sibuya. Celle-ci satisfait à l'équation différentielle linéaire affine

$$x^5(2-x)y'' + x^2(4+5x^2-2x^3)y' + 2(2-x+x^2)y = 4x + 2x^2 + 10x^3 - 3x^4. \quad (6)$$

En adaptant les méthodes développées au paragraphe 2 on obtient les équations différentielles linéaires affines

$$\begin{aligned} L_3(y) &= 6x^6 + 56x^4 && \text{qui admet pour solution } \hat{f}^0 \\ \text{et } L_3(y) &= -x^5 - 14x^3 - 8x && \text{qui admet pour solution } \hat{f}^1 \end{aligned}$$

où L_3 est l'opérateur

$$\begin{aligned} L_3 &= (x^9 + 4x^7) \partial^3 + (34x^6 + 8x^4 + 6x^8) \partial^2 \\ &\quad + (12x^3 + 6x^7 + 43x^5) \partial - 10x^2 - 8. \end{aligned}$$

(Une base de solutions de l'équation $L_3(y) = 0$ est donnée par $e^{1/x}$, $e^{-1/x}$ et e^{1/x^2}).

En dérivant respectivement $(6x^6 + 56x^4)^{-1}L_3(y)$ et $(-x^5 - 14x^3 - 8x)^{-1}L_3(y)$ on obtient les équations différentielles linéaires («sans second membre»)

$$L'_4(y) = 0 \quad \text{qui admet pour solution } \hat{f}^0$$

et

$$L''_4(y) = 0 \quad \text{qui admet pour solution } \hat{f}^1$$

où

$$\begin{aligned} L'_4(y) &= x^8(3x^2 + 28) \partial^4 + x^5(27x^4 + 314x^2 + 56) \partial^3 \\ &\quad + 3x^4(18x^4 + 251x^2 + 28) \partial^2 + x(18x^6 + 303x^4 - 146x^2 - 56) \partial \\ &\quad + 120x^2 + 224 \end{aligned}$$

et

$$\begin{aligned} L''_4(y) &= x^8(x^4 + 14x^2 + 8) \partial^4 + 2x^5(5x^6 + 85x^4 + 70x^2 + 8) \partial^3 \\ &\quad + x^4(24x^6 + 485x^4 + 478x^2 + 72) \partial^2 \\ &\quad + 2x(6x^8 + 144x^6 + 153x^4 + 2x^2 - 8) \partial \\ &\quad + 30x^4 + 60x^2 + 16. \end{aligned}$$

Ces équations sont d'ordre 4 ce qui prouve, en particulier, que l'équation $L_5(y) = 0$ n'est pas l'équation d'ordre minimal vérifiée par l'une ou l'autre des sous-séries \hat{f}^0 ou \hat{f}^1 . On vérifie que l'opérateur L_5 qui admet simultanément les deux séries \hat{f}^0 et \hat{f}^1 pour solutions est, en fait, le ppcm de L'_4 et de L''_4 . En remarquant que la série \hat{f}^1 est la sous-série des termes impairs de la série d'Euler et en appliquant la même méthode à l'équation d'Euler $x^2y' + y = x$ on obtient pour \hat{f}^1 l'équation avec second membre $x^4y'' + 2x^3y' - y = -x$ ou encore, sous forme homogène, l'équation $x^5y''' + 5x^4y'' + x(4x^2 - 1)y' + y = 0$. Celle-ci est d'ordre 3 et non pas 4 ce qui montre que l'équation $L''_4y = 0$ n'est pas l'équation d'ordre minimal vérifiée par \hat{f}^1 .

5 Appendice : invariants formels d'une équation ou d'un système

Dans ce paragraphe, nous indiquons comment accéder de façon explicite ou algorithmique aux invariants formels des équations ou systèmes différentiels et, en particulier, à leur différents niveaux. Le cas d'une équation est beaucoup plus facile que celui d'un système : tous les invariants formels se lisent sur un nombre fini de polygones de Newton associés à l'équation moyennant la résolution d'un certain nombre d'équations algébriques qui sont génériquement de degré un.

Quitte à réduire les systèmes à une forme compagnon on peut ainsi, en théorie, calculer tous les invariants formels d'un système. Malheureusement, une telle réduction par un algorithme de vecteur cyclique conduit souvent à des équations dont les coefficients sont énormes voire incalculables en pratique. Il est donc justifié de s'attacher à la recherche d'algorithmes opérant directement sur les systèmes.

Les invariants formels d'un système se lisent sur une solution fondamentale formelle et caractérisent chaque classe formelle. Rappelons que la classification formelle des systèmes est la classification modulo transformation de jauge méromorphe formelle $Y = T(x)Z$, i.e., dans laquelle T est une matrice inversible à coefficients méromorphes formels. La notion d'équivalence formelle en termes d'équations s'énonce moins simplement. On renvoie à [Ore33] ou à [S96] Cor. 2.6.

5.1 Invariants formels des équations différentielles

Considérons, comme au paragraphe 2, une équation différentielle

$$Dy = 0 \tag{1}$$

où l'opérateur

$$D = a_n(x)\partial^n + a_{n-1}(x)\partial^{n-1} + \cdots + a_1(x)\partial + a_0(x)$$

est à coefficients méromorphes convergents ou formels en $x = 0$ (il n'est pas utile ici de se restreindre au cas polynomial) et notons $v(a)$ la valuation en x d'une série méromorphe a .

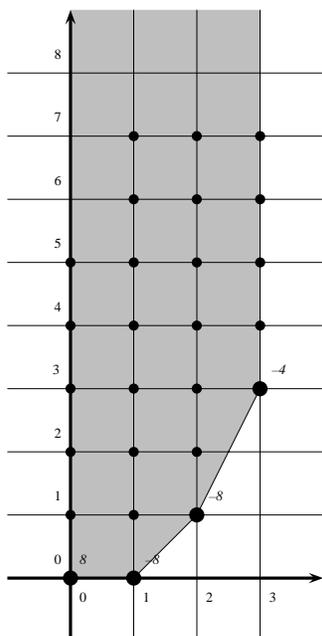
Les algorithmes de détermination des solutions formelles reposent sur les notions de polygones de Newton et d'équations caractéristiques qui leur sont attachées (voir ([BJ97] ou [M79]) par exemple). Nous esquissons ces notions ci-dessous.

- Le polygone de Newton de l'équation (1) elle-même est défini comme suit : Dans le demi-plan $\mathbb{R}^+ \times \mathbb{R}$ on considère l'ensemble des deuxièmes quadrants

$$Q_j = \{(x, y) \in \mathbb{R}^+ \times \mathbb{R} \mid x \leq j \text{ et } y \geq v(a_j) - j\}$$

de sommets chacun des points $(j, v(a_j) - j)$, $j = 0, \dots, n$ et $a_j \neq 0$. Le polygone de Newton $N(D)$ de D est l'enveloppe convexe de la réunion $Q = \cup_{j=0,1,\dots,n} Q_j$ de tous ces quadrants.

La figure ci-dessous représente le polygone de Newton de l'équation de Ramis–Sibuya.



On a indiqué, en italique à côté de chaque point marqué sur le bord du polygone de Newton, le coefficient à prendre en compte dans les équations indicelle ou caractéristiques.

Polygone de Newton de l'équation de Ramis–Sibuya

Les pentes de $N(D)$ sont des nombres rationnels positifs ou nuls. Celles qui sont strictement positives sont appelées *les niveaux de D*. Quitte à utiliser une extension finie $x = t^p$ de la variable x , on peut se ramener au cas de pentes entières. Nous

supposons donc désormais, pour simplifier les écritures, que les pentes de $N(D)$ sont entières.

• Les équations caractéristiques sont relatives à chaque côté de $N(D)$. Pour un côté de pente $k > 0$ on parle d'équation k -caractéristique ; pour le côté de pente nulle, s'il existe, d'équation indicielle. Ces équations sont définies par le symbole principal de D associé à la filtration par le k -poids w_k ainsi défini : on choisit $w_k(x) = \frac{1}{k+1}$ et $w_k(\partial) = -1$ de sorte que $w_k(x^{k+1}\partial) = 0$ et on définit le poids $w_k(D)$ comme le plus petit des k -poids de ses monômes.

Pour $k = 0$ et si le côté horizontal de $N(D)$ est de longueur $n_0 > 0$ la partie homogène de plus bas 0-poids de D est de la forme

$$\sigma_0(D) = x^m \sum_{j=0}^{n_0} a_{m+j,j} x^j \partial^j \quad \text{avec } a_{m+n_0,n_0} \neq 0.$$

On obtient le polynôme indiciel de D en négligeant le facteur x^m et en substituant à $x^j \partial^j$ le produit $X(X-1)\dots(X-j+1)$ d'où l'équation indicielle de D :

$$\sum_{j=0}^{n_0} a_{m+j,j} X(X-1)\dots(X-j+1) = 0$$

Pour $k > 0$ et si $N(D)$ admet un côté de pente k de longueur horizontale n_k la partie homogène de plus bas k -poids de D s'écrit (modulo des termes de k -poids supérieur si on écrit $(x^{k+1}\partial)^j$ au lieu de $x^{(k+1)j}\partial^j$)

$$\begin{aligned} \sigma_k(D) &\equiv x^{m'} \sum_{j=N'_k}^{N_k} a_{m'+(k+1)j,j} (x^{k+1}\partial)^j \\ &\equiv x^{m'} (x^{k+1}\partial)^{N'_k} \sum_{j=0}^{n_k} a_{m'+(k+1)(N'_k+j),N'_k+j} (x^{k+1}\partial)^j \end{aligned}$$

où on a noté $N'_k = n_0 + \dots + n_{k-1}$ et $N_k = N'_k + n_k$. Ici, les deux termes extrêmes $a_{m'+(k+1)N'_k,N'_k}$ et $a_{m'+(k+1)N_k,N_k} (x^{k+1}\partial)^{n_k}$ sont non nuls. On obtient le polynôme k -caractéristique de D en négligeant le facteur $x^{m'} (x^{k+1}\partial)^{N'_k}$ et en substituant X à $x^{k+1}\partial$ d'où l'équation k -caractéristique de D :

$$\sum_{j=0}^{n_k} a_{m'+(k+1)(N'_k+j),N'_k+j} X^j = 0.$$

Compte tenu de la remarque précédente, cette équation n'admet pas $X = 0$ pour solution.

Décrivons à présent comment on obtient les solutions formelles de D .

- À un côté horizontal de longueur $n_0 \geq 0$ correspondent n_0 solutions formelles-logarithmiques de la forme

$$\Phi(x) = x^\mu \sum_{j=0}^{s-1} \varphi_j(x) (\ln x)^j$$

où $\mu \in \mathbb{C}$, $s \in \mathbb{N}^*$ et où les φ_j sont des séries formelles en x .

Celles-ci sont calculables par la méthode de Frobenius. Les différentes valeurs de μ sont les racines de l'équation indicelle comptées avec leur multiplicité. s est inférieur ou égal à la multiplicité de l'exposant μ correspondant modulo \mathbb{Z} . La méthode de Frobenius consiste à chercher des solutions de la forme $x^\mu \sum_{m \geq 0} u_m x^m$ par identification et à introduire autant de puissances successives de $\ln x$ que nécessaire pour contourner les incompatibilités. Pour les détails on renvoie à [Ince] Chap. XVI.

- À un côté de pente $k > 0$ et de longueur horizontale n_k correspondent n_k solutions de la forme

$$y = \Phi(x) e^{q(1/x)}$$

où Φ est une combinaison linéaire de séries formelles-logarithmiques du type précédent et où q est de la forme $q\left(\frac{1}{x}\right) = \frac{-\alpha_k}{kx^k} + q_1\left(\frac{1}{x}\right)$, le terme $q_1\left(\frac{1}{x}\right)$ étant un polynôme en $\frac{1}{x}$ ou en une puissance fractionnaire de $\frac{1}{x}$, de degré en $\frac{1}{x}$ strictement inférieur à k . Les différents α_k qui apparaissent sont les racines de l'équation k -caractéristique de D comptées avec leur multiplicité.

Pour obtenir les termes suivants de q il suffit d'opérer le changement d'inconnue $y = e^{-\alpha_k/kx^k} z$ et de considérer l'équation différentielle en z . Dans le polygone de Newton de cette équation on ne prend en compte que les pentes inférieures à k . On construit ainsi une suite finie d'opérateurs jusqu'à l'obtention d'une pente nulle comme seule pente admissible. À cette dernière, si elle existe, on applique la méthode de Frobenius pour déterminer le facteur Φ .

Les invariants méromorphes formels de (1) sont les triplets $(\mu \bmod \mathbb{Z}, s, q)$.

Les différents $\mu \bmod \mathbb{Z}$ sont les exposants de monodromie formelle. Les nombres entiers s associés à μ sont les dimensions des blocs de Jordan de la monodromie formelle pour la valeur propre $e^{2\pi i \mu}$. La partie exponentielle $e^{q(1/x)}$ des solutions est aussi appelée leur partie irrégulière. Dans le cas de l'équation de Ramis-Sibuya les invariants méromorphes formels sont ainsi les triplets $(0, 1, 0)$, $(0, 1, \frac{1}{x})$, $(0, 1, \frac{1}{x^2})$.

Des procédures de calcul de ces invariants sont implantées dans le code *desir2*; on trouvera un pointeur sur *desir2* dans [BCL].

5.2 Invariants formels des systèmes différentiels

Nous considérons maintenant un système

$$\frac{dY}{dx} - A(x)Y = 0 \quad (2)$$

de rang de Poincaré r et nous nous proposons de déterminer les invariants de ce système sans passer par l'intermédiaire d'une forme compagnon. Naturellement, dans le cas où l'on dispose d'une forme compagnon, comme dans celui où une telle forme compagnon est facile à obtenir, il est préférable de déterminer les invariants formels sur l'équation associée.

En écrivant $A(x) = \frac{1}{x^{r+1}} \sum_{m \geq 0} A_m x^m$, nous supposons $A_0 \neq 0$ sinon r ne serait pas le rang de Poincaré et nous supposons $r \geq 0$ sinon l'origine ne serait pas un point singulier de (2), toutes les solutions seraient analytiques à l'origine et il n'y aurait pas d'invariant à déterminer.

Rappelons qu'un tel système admet une solution fondamentale formelle de la forme

$$\widehat{Y}(x) = \widehat{F}(x)x^L e^{Q(1/x)} \quad (3)$$

où $Q(1/x)$ est une matrice diagonale de polynômes en $\frac{1}{x}$ ou en une puissance fractionnaire de $\frac{1}{x}$, L est une matrice constante complexe et \widehat{F} une matrice de séries méromorphes formelles ([BJL79], [L-R01] Theorem 2.2).

Les invariants formels du système (2) sont donnés par $Q(1/x)$ et L , plus précisément, par la forme de Jordan de L , les valeurs propres étant prises modulo \mathbb{Z} .

Lorsque $Q \equiv 0$ on dit que l'origine est une *singularité régulière* : dans ce cas, en particulier, la matrice \widehat{F} est convergente dès que A l'est. Lorsque $Q \not\equiv 0$ on dit que l'origine est une *singularité irrégulière* et la matrice \widehat{F} est, en général, une série divergente.

Le degré κ (éventuellement fractionnaire) de $Q(1/x)$ en la variable $\frac{1}{x}$ est appelé *le rang de Katz* du système. On montre que le rang de Katz κ est toujours inférieur ou égal au rang de Poincaré r et que le vrai rang de Poincaré est la partie entière supérieure de κ (i.e., le plus petit nombre entier qui lui est supérieur ou égal). Ainsi, un système est à singularité régulière si et seulement si son vrai rang de Poincaré est nul. En particulier, les systèmes de rang de Poincaré 0, quelquefois appelés *systèmes de première espèce*, sont toujours singuliers réguliers. On dispose, pour ces systèmes de première espèce, de bons algorithmes de calcul d'une solution fondamentale formelle $\widehat{F}(x)x^L$ ([CL], [S86]). On peut, dans ce cas, choisir pour L une forme de Jordan de la matrice de tête A_0 de xA . En particulier, les valeurs propres de L sont celles de la matrice A_0 et l'équation indicelle est ici remplacée par l'équation caractéristique de A_0 . La matrice $\widehat{F}(x)$ n'a pas de partie polaire ; la matrice $\widehat{F}(0)$ est inversible et ses colonnes forment une base des sous-espaces caractéristiques de A_0 . Le reste de la matrice $\widehat{F}(x)$ peut

alors être calculé par identification dans le système et on obtient ainsi des relations de récurrence pour calculer de proche en proche toutes les parties homogènes $F_j x^j$ de $F(x)$.

Pour plus de précision, si nécessaire, nous noterons $\kappa(A), r(A), \dots$ le rang de Katz, de Poincaré, ... du système (2) : $\frac{dY}{dx} - A(x)Y = 0$.

Une première étape importante dans la recherche des invariants formels est la détermination du rang de Katz. Commençons par voir comment on peut décider si ce rang est nul, c'est-à-dire, si la singularité 0 est régulière en suivant une méthode due à Moser ([Mo60]). Nous traiterons ensuite du cas singulier irrégulier.

5.2.1 Détermination de la régularité, rang de Moser et calcul des solutions régulières. La méthode de Moser consiste à faire chuter le rang de Poincaré du système jusqu'à l'obtention du vrai rang de Poincaré et ceci au moyen de transformations de jauge polynomiales de degré au plus 2. Dans le cas où le système est singulier régulier on obtient ainsi un système de première espèce dont on sait calculer les solutions ([CL], [S86]). Pour tester si le rang de Poincaré est minimal, Moser attache au système une nouvelle quantité aujourd'hui communément appelée rang de Moser et qui est aisément lisible sur le système. Nous en rappelons la définition.

Définition 5.1. (i) Le rang de Moser du système (2) : $\frac{dY}{dx} - A(x)Y = 0$ est le nombre rationnel

$$m(A) = \max\left(0, r + \frac{r_0}{n}\right)$$

où r désigne le rang de Poincaré du système (2) en 0, r_0 le rang de la matrice de tête A_0 de $x^{r+1}A$ et n la dimension du système (2).

(ii) Le polynôme de Moser attaché au système (2) est le polynôme de la variable λ défini par

$$\mathcal{B}(A, \lambda) = x^{r_0} \det \left(\lambda I + \frac{A_0}{x} + A_1 \right) \Big|_{x=0}$$

đ
in next
display
correct ????

Comme c'est le cas du rang de Poincaré le rang de Moser n'est pas invariant par transformation de jauge méromorphe mais il admet une plus petite valeur possible $\mu(A)$ que nous appellerons *vrai rang de Moser* du système.

Clairement, 0 est une singularité régulière du système (2) si et seulement si le vrai rang de Moser $\mu(A)$ du système en 0 vérifie la condition

$$\mu(A) \leq 1.$$

Ainsi, si le rang de Moser $m(A)$ est ≤ 1 le système est singulier régulier en 0. Dans le cas où $m(A) > 1$, le théorème de Moser ci-dessous donne une condition

nécessaire et suffisante, simple à vérifier, pour déterminer si $m(A)$ est égal ou non au vrai rang de Moser $\mu(A)$. Dans le cas où $m(A) \neq \mu(A)$, Moser fournit, en outre, un algorithme construisant à l'aide de transformations de jauge adéquates sur le système (2) un système $\frac{dY}{dx} - B(x)Y = 0$ de rang de Moser minimal $m(B) = \mu(A)$ donc de rang de Poincaré minimal.

Théorème 5.2 ([Mo60]). (i) *Le rang de Moser $m(A)$ du système (2) est égal au vrai rang de Moser $\mu(A)$ si et seulement si le polynôme de Moser $\mathcal{B}(A, \lambda)$ n'est pas identiquement nul.*

(ii) *Si $\mu(A) < m(A)$ on peut réduire le rang de Moser du système par des transformations de jauge de la forme*

$$T(x) = (P_0 + P_1x)\Delta$$

où P_0 et P_1 sont des matrices constantes, P_0 est inversible et où Δ est une matrice diagonale dont les termes diagonaux ne prennent que les valeurs 1 ou x .

Pour plus de précisions sur l'algorithme on renvoie à [H87] et [B95]. Signalons, en outre, que l'algorithme de Moser est implanté dans la procédure **moser_reduce** du paquetage DEtools de MAPLE.

Exemple 5.3. Considérons le système $Y' = AY$ où $A = \begin{bmatrix} 4x^{-1} & -4 \\ 2x^{-2} & -3x^{-1} \end{bmatrix}$.

La matrice A est de la forme $A = \frac{1}{x^2}(A_0 + A_1x + A_2x^2)$ où

$$A_0 = \begin{bmatrix} 0 & 0 \\ 2 & 0 \end{bmatrix}, \quad A_1 = \begin{bmatrix} 4 & 0 \\ 0 & -3 \end{bmatrix} \quad \text{et} \quad A_2 = \begin{bmatrix} 0 & -4 \\ 0 & 0 \end{bmatrix}$$

Le rang de Poincaré r vaut 1. La matrice A_0 est nilpotente de rang $r_0 = 1$ et rang de Moser $m(A)$ du système vaut donc $m(A) = 1 + 1/2 = 3/2 > 1$. Comme

$$x \det \left(\lambda I + \frac{A_0}{x} + A_1 \right) = x \begin{vmatrix} \lambda + 4 & 0 \\ 2x^{-1} & \lambda - 3 \end{vmatrix} = x(\lambda + 4)(\lambda - 3)$$

le polynôme de Moser $\mathcal{B}(A, \lambda)$ est identiquement nul et le vrai rang de Moser du système est strictement inférieur à $\frac{3}{2}$. Comme ce rang est un nombre rationnel de dénominateur 2 on peut, ici, d'ores et déjà conclure que $\mu(A) \leq 1$, donc que le système est singulier régulier en 0.

La transformation de jauge de matrice $T = \begin{bmatrix} x & 0 \\ 0 & 1 \end{bmatrix}$ réduit le rang de Moser à 1.

En effet, le nouveau système s'écrit $\frac{dY}{dx} - B(x)Y = 0$ où

$$B = T^{-1}AT - T^{-1}T' = \frac{1}{x} \begin{bmatrix} 3 & -4 \\ 2 & -3 \end{bmatrix}.$$

Comme c'est un système de première espèce en 0 on arrête là la procédure de réduction : le système donné est un système singulier régulier en 0 équivalent au système $\frac{dY}{dx} - B(x)Y = 0$ ci-dessus.

Dans cet exemple, on a pu conclure que le système est singulier régulier sans avoir à effectuer la réduction du système par des transformations de Moser mais, en général, et chaque fois que la procédure de Moser requiert plusieurs étapes, on ne peut éviter le calcul effectif de réduction du système à chaque étape.

5.2.2 Cas des singularités irrégulières. Turriffin ([T55]) et Wasow ([Was]) ont, les premiers, donné en toute généralité une matrice fondamentale de solutions formelles d'un système (2). Leur démonstration conduit à une version faible de la formule (3) dans laquelle la série \widehat{F} peut contenir les puissances fractionnaires de x nécessaires à l'écriture de Q .

Dans [BJL79] W. Balsler, W. B. Jurkat et D. A. Lutz montrent qu'une bonne symétrisation des formules obtenues par Turriffin et Wasow permet d'obtenir la forme fine (3) puis, dans [L-R01], on exprime cette symétrisation en termes de réduction du rang de certains systèmes simples exhibant les invariants formels. Ainsi peut-on déjà lire tous les invariants formels du système sur la forme faible de Turriffin-Wasow.

La démonstration de Turriffin-Wasow est algorithmique. Elle repose sur une triple récurrence portant sur le rang et la dimension du système et sur le nombre de blocs de Jordan de A_0 . Elle opère en alternance des opérations du type suivant :

- Découplage : si la matrice A_0 est somme directe de deux matrices sans valeurs propres communes alors on peut scinder de façon analogue le système par une transformation de jauge formelle ([Was] Theorem 11.1).
- Changement de vecteur inconnu de la forme $Y = e^{\frac{-\alpha}{rx^r}} Z$.
- Mise sous forme réduite de Turriffin-Wasow : la matrice de tête A_0 de $x^{r+1}A$ étant sous forme de Jordan nilpotente il s'agit de faire apparaître dans les termes suivants A_j des zéros sur toutes les lignes non nulles de A_0 ([Was] Lemma 19.2).
- Cisaillement : ce sont des transformations de jauge diagonales de la forme

$$\text{diag}(1, x^g, x^{2g}, \dots, x^{(n-1)g})$$

pour un g convenable à déterminer.

- Extension algébrique convenable $x = t^s$ de la variable.

Cette construction a le mérite d'établir la partie centrale du théorème de classification formelle. Malheureusement, dans la pratique, elle s'avère très peu efficace pour calculer les invariants d'un système donné. En particulier, la mise sous forme réduite de Turritin-Wasow est extrêmement lourde et aboutit difficilement. Or, il est en général nécessaire d'y recourir plusieurs fois.

Nous présentons ci-dessous un protocole dans lequel on s'intéresse en priorité à la détermination de la partie irrégulière d'une solution fondamentale ([B97]); on sait alors qu'il suffit de garder les nr premiers termes de chaque coefficient du système pour garantir la validité des résultats ([BV83], [LS85]). On n'a pas de borne semblable sur le nombre de termes permettant de calculer sans erreur la partie régulière et un nombre donné de termes dans \widehat{F} . Un algorithme de calcul des solutions régulières d'un système singulier irrégulier permet de contourner cette difficulté ([BP99]).

Les procédures sont implantées sous Maple dans le code *isolde*. On trouvera un pointeur vers *isolde* dans [BCL].

Protocole pour le calcul de la partie irrégulière :

1. On réduit le système à son vrai rang de Moser (cf. Théorème 5.2).

On note encore (2) : $\frac{dY}{dx} - A(x)Y = 0$ avec $A(x) = \frac{1}{x^{r+1}} \sum_{j \geq 0} A_j x^j$ le système ainsi réduit, n sa dimension, r son rang de Poincaré, $m = m(A)$ son rang de Moser, $\kappa = \kappa(A)$ son rang de Katz et r_0 le rang de A_0 .

2. Si $m \leq 1$, le système est à singularité régulière : sa partie irrégulière Q est nulle.
3. Si $n = 1$ et $m > 1$, la partie irrégulière Q s'obtient par intégration terme à terme de la partie polaire de $A(x)$.
4. Si A_0 admet au moins deux valeurs propres distinctes, on scinde le système (2) par la procédure de découplage de Turritin-Wasow et on reprend en **1** pour chacun des systèmes obtenus.
5. Si A_0 admet la seule valeur propre $\alpha \neq 0$ on effectue le changement de vecteur inconnu $Y = e^{\frac{\alpha}{rx}} Z$ (qui transforme le système (2) en un système de rang de Poincaré $< r$ et de matrice de tête $A_0 - \alpha I$ nilpotente) et on revient en **1**.
6. Si la matrice A_0 est nilpotente, alors

- 6.1. Si $r + r_0 > n$ le rang de Katz est un nombre rationnel non entier $\kappa(A) = \frac{r'}{s'}$, $r - 1 < \kappa(A) < r$, donné par (cf. [B97] Theorem 1)

$$\kappa(A) = \max \left(0, \max_{0 \leq j < n} \frac{-n + j - v(c_j)}{n - j} \right) \quad (4)$$

où les c_j sont les coefficients du polynôme caractéristique de A ,

$$\det(\lambda I - A) = \lambda^n + c_{n-1} \lambda^{n-1} + \dots + c_0,$$

et où v désigne la valuation.

- 6.2. Si $r + r_0 \leq n$ on augmente la valeur de r de façon à satisfaire à la condition $r + r_0 > n$ par une extension algébrique $x \mapsto x^s$ de la variable x de degré s suffisant ; on peut alors appliquer 6.1.
- 6.3. On opère l'extension $x = t^{s'}$ de la variable. On obtient un nouveau système $\frac{dZ}{dt} - B(t)Z = 0$, où $B = \frac{1}{t^{rs'+1}} \sum_{j \geq 0} s' A_j t^{js'}$, qui est de rang de Poincaré rs' et de rang de Katz entier $r' < rs'$. (Le vrai rang de Poincaré est donc égal à r'). Après avoir réduit le système à son rang de Moser minimal on est ramené au point 4 ci-dessus.

Remarque 5.4. La formule pour le rang de Katz donnée à l'étape 6.1 étend la notion de rang de Katz d'une équation différentielle : si le système (2) est le système compagnon d'une équation différentielle, cette valeur est bien égale à la plus grande pente du polygone de Newton de celle-ci.

Exemple 5.5. Considérons le système $\frac{dY}{dx} = A(x)Y$ où

$$A(x) = \frac{1}{x^4} \begin{bmatrix} 0 & 0 & x & 0 \\ 1 & -x^2 & x^2 & -x^2 \\ 0 & 1 & x^2 & 0 \\ x^2 & x^2 & 0 & -x^2 \end{bmatrix}.$$

Le rang de Poincaré vaut $r = 3$.

La matrice de tête $A_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ est nilpotente de rang $r_0 = 2$.

Le polynôme de Moser $\mathcal{B}(A, \lambda) = \lambda$ n'étant pas identiquement nul, le rang de Moser $m(A) = 3 + \frac{2}{4} = \frac{7}{2}$ est minimal.

La condition $r + r_0 > n$ étant satisfaite (étape 6.1 ci-dessus) on peut calculer $\kappa(A)$ en utilisant la formule (4) : comme

$$\det(\lambda I - A(x)) = \lambda^4 + \frac{\lambda^3}{x^2} - \frac{\lambda^2}{x^6} + \frac{(-2x^7 - x^2 - x^5)\lambda}{x^{13}} + \frac{x^2 - 1}{x^{13}},$$

on a $v(a_3) = -2$, $v(a_2) = -6$, $v(a_1) = -11$, $v(a_0) = -13$ (v désigne la valuation) et, par suite,

$$\kappa(A) = \max \left\{ 0, 1, 2, \frac{8}{3}, \frac{9}{4} \right\} = \frac{8}{3}.$$

En posant $x = t^3$, on obtient le système $\frac{dY}{dt} = B(t)Y$ où

$$B(t) = \frac{3}{t^{10}} \begin{bmatrix} 0 & 0 & t^3 & 0 \\ 1 & -t^6 & t^6 & -t^6 \\ 0 & 1 & t^6 & 0 \\ t^6 & t^6 & 0 & -t^6 \end{bmatrix}.$$

La méthode de Moser conduit à appliquer à ce système la transformation de jauge $Y = SZ$ où $S = \text{diag}(t^2, t, 1, 1)$. On obtient alors le système

$$\frac{dZ}{dt} = C(t)Z \quad \text{où} \quad C(t) = \frac{1}{t^9} \begin{bmatrix} -2t^8 & 0 & 3 & 0 \\ 3 & -3t^5 - t^8 & 3t^4 & -3t^4 \\ 0 & 3 & 3t^5 & 0 \\ 3t^7 & 3t^6 & 0 & -3t^5 \end{bmatrix}$$

dont le rang de Poincaré est bien égal à 8.

La matrice de tête $C_0 = \begin{bmatrix} 0 & 0 & 3 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ est non nilpotente et admet les valeurs

propres distinctes $0, 3, 3j$ et $3j^2$ (où $j^3 = 1$). Le système peut donc être découpé en 4 équations scalaires.

On obtient les invariants formels $e^{1/x}$ et $\frac{1}{t}e^{-3/(8t^8)-1/(4t^4)}$ pour chacune des trois déterminations $t = x^{1/3}$, $t = jx^{1/3}$ et $t = j^2x^{1/3}$ d'une racine cubique de x . Une solution fondamentale formelle s'écrit alors sous la forme

$$\widehat{Y}(x) = \widehat{F}(x) \begin{bmatrix} e^{1/x} & 0 \\ 0 & x^J U e^{Q(1/x)} \end{bmatrix}$$

où $\widehat{F}(x)$ est une série méromorphe formelle en x ,

$$J = -\frac{1}{3} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/3 & 0 \\ 0 & 0 & 2/3 \end{bmatrix},$$

$$U = \begin{bmatrix} 1 & 1 & 1 \\ 1 & j & j^2 \\ 1 & j^2 & j^4 \end{bmatrix}$$

et

$$Q(1/x) = \begin{bmatrix} q(1/x) & 0 & \\ 0 & q(1/(jx)) & 0 \\ 0 & 0 & q(1/(j^2x)) \end{bmatrix}$$

avec $q\left(\frac{1}{x}\right) = -\frac{3}{8x^{8/3}} - \frac{1}{4x^{4/3}}$.

5.3 Vecteur cyclique

Certains des algorithmes que nous proposons utilisent une méthode dite «de vecteur cyclique». Rappelons brièvement en quoi elle consiste et comment la mener à bien.

Étant donné un système différentiel $\partial Y = AY$ à coefficients dans un corps différentiel (K, ∂) on se propose de mettre ce système sous forme compagnon à l'aide de transformations de jauge sur K . On pourra alors identifier le système à l'équation différentielle dont il est le compagnon. Plusieurs réponses algorithmiques ont été données à ce problème, d'abord par Cope sous l'hypothèse que K soit un corps de fonctions, puis, sous des hypothèses plus générales, par P. Deligne, N. Katz,...

Il existe plusieurs variantes algorithmiques de cette méthode ; en voici une ([B93]).

On choisit $y = \Lambda Y$ où $\Lambda = [\lambda_1, \dots, \lambda_n] \in K^n$ une forme linéaire en les composantes y_1, \dots, y_n de Y . Compte tenu de la relation $\partial Y = AY$ les dérivées successives de y sont, elles aussi, des formes linéaires en les composantes y_1, \dots, y_n de Y soit

$$\partial^j y = \Lambda_j Y \tag{5}$$

où les matrices lignes Λ_j sont définies de proche en proche par les relations $\Lambda_j = \partial \Lambda_{j-1} + \Lambda_{j-1} A$ à partir de $\Lambda_0 = A$.

La méthode du vecteur cyclique consiste à chercher la relation linéaire d'ordre minimal satisfaite par y et ses formes dérivées successives. On obtient ainsi une équation différentielle d'ordre au plus n en l'inconnue y

$$a_n y^{(n)} + a_{n-1} y^{(n-1)} + \dots + a_0 y = 0. \tag{6}$$

Si celle-ci est d'ordre n on dit que Λ est un *vecteur cyclique* pour le système $\partial Y = AY$.

En posant $Z = \begin{bmatrix} y \\ \partial y \\ \vdots \\ \partial^{n-1} y \end{bmatrix}$ les relations (5) ci-dessus s'écrivent $Z = PY$ où la

matrice P a pour lignes successives $\Lambda_0, \Lambda_1, \dots, \Lambda_{n-1}$. Le vecteur Λ est cyclique si

et seulement si la matrice P est inversible. Z vérifie alors un système $\partial Z = BZ$, où $B = \partial P \cdot P^{-1} + P A P^{-1}$, qui n'est autre que le système compagnon de l'équation (6). Remarquons, en particulier, que l'ensemble des vecteurs cycliques est dense dans K^n .

On trouvera une implantation en MAPLE de cet algorithme, y compris lorsque le vecteur n'est pas cyclique, dans [BCL].

Références

- [BCL] Adresse des programmes ou pointeurs : http://www.unilim.fr/pages_perso/moulay.barkatou/
- [Co] P. M. Cohn, *Free Rings and their Relations*, London Math. Soc. Monogr. 2, Academic Press, 1971.
- [L-R01] M. Loday-Richaud, Rank reduction, Normal Forms and Stokes Matrices, *Exposition. Math.* 19 (2001), 229–250.
- [Le99] A. H. M. Levelt, Formal solutions of linear differential equations, Cathode meeting (June 1999), preprint, Nijmegen.
- [Lu72] D. A. Lutz, On the reduction of the rank of linear differential systems, *Pacific J. Math.* 42 (1972), 153–164.
- [N95] F. Naegelé, *Autour de quelques équations fonctionnelles analytiques*, Thèse de l'Université Joseph Fourier, Grenoble, France, Décembre 1995.
- [Ore33] O. Ore, Theory of non-commutative polynomials, *Ann. of Math.* (2) 34 (1933), 480–508.
- [S96] M.F Singer, Testing Reducibility of Linear Differential Operators : A Group Theoretic Perspective, *Appl. Algebra Engrg. Comm. Comput.* 7 (1996), 77–104.
- [SZ94] B. Salvy, P. Zimmermann, Gfun : a Maple package for the manipulation of generating and holonomic functions in one variable, *ACM Trans. Math. Software* 20 (1994), 163–177; <http://algo.inria.fr/libraries/>
- [T63] H. L. Turriffin, Reducing the rank of ordinary differential equations, *Duke Math. J.* 30 (1963), 271–274.

Sur la sommation des séries divergentes et le calcul des invariants méromorphes

- [Ba-1] W. Balser, *From divergent Power Series to Analytic Functions*, Lecture Notes in Math. 1582, Springer-Verlag, 1994.
- [Br91] B. L. J. Braaksma, Multisummability and Stokes multipliers of linear meromorphic differential equations, *J. Differential Equations* 92 (1991), 45–75.
- [JNT96] F. Jung, F. Naegelé, J. Thomann, An algorithm of multisummation of formal power series solutions of linear ODEs, *Math. Comput. Simulation* 42 (1996), 409–425.
- [JLP76] W. Jurkat, D. Lutz, A. Peyerimhoff, Birkhoff invariants and effective calculations for meromorphic linear differential equations I, *J. Math. Anal. Appl.* 53 (1976), 438–470.

- [L-R90] M. Loday-Richaud, Solutions formelles des systèmes différentiels linéaires méromorphes et sommation, *Exposition. Math.* 13 (1995), 116–162.
- [L-R90] M. Loday-Richaud, Calcul des Invariants de Birkhoff des Systèmes d'Ordre Deux, *Funkcial. Ekvac.* 33 (1990), 161–225.
- [LS97] D. Lutz, R. Schäfke, Calculating connection coefficients for meromorphic differential equations, *Complex Variables Theory Appl.* 34 (1–2) (1997), 145–170.
- [M95] B. Malgrange, Somme des séries divergentes, *Exposition. Math.* 13 (1995), 163–222.
- [MR92] B. Malgrange, J.-P. Ramis, Fonctions multisommables, *Ann. Inst. Fourier* 42 (1992), 353–368.
- [MR82] J. Martinet, J.-P. Ramis, Problèmes de modules pour des équations différentielles non linéaires du premier ordre, *Inst. Hautes Études Sci. Publ. Math.* 55 (1982), 63–164.
- [NT96] F. Naegelé, J. Thomann, Algorithmic approach of the multisummation of formal power series solutions of linear ODE applied to the Stokes phenomena, in : *The Stokes phenomenon and Hilbert's 16th problem*, Groningen 1995, World Sci. Publishing, River Edge, N.J., 1996, 197–213.
- [Th90] J. Thomann, Resommation des séries formelles solutions d'équations différentielles linéaires ordinaires du second degré dans le champ complexe au voisinage de singularités irrégulières, *Numer. Math.* 58 (1990), 503–535.

Sur les formes normales et les invariants formels

- [BV83] D. G. Babbitt, V. S. Varadarajan, Formal reduction of meromorphic differential equations : a group theoretic view, *Pacific J. Math.* 109 (1) (1983), 1–80.
- [Ba-2] W. Balser, *Formal power series and linear systems of meromorphic ordinary differential equations*, Springer-Verlag, 2000.
- [BJL79] W. Balser, W. B. Jurkat, D. A. Lutz, A General Theory of Invariants for Meromorphic Differential Equations ; Part I, Formal Invariants, *Funkcial. Ekvac.* 22 (1979), 197–221.
- [B89] M. A. Barkatou, Rational Newton algorithm for computing formal solutions of linear differential equations, in : *Symbolic and algebraic computation* Rome, 1988, Lecture Notes in Comput. Sci. 358, Springer-Verlag, 1989, 183–195.
- [B95] M. A. Barkatou, A rational version of Moser's algorithm, in : *Proceedings of the 1995 international symposium on symbolic and algebraic computation*, ISSAC '95, Montréal, Canada, Academic Press, 1995, 297–302.
- [B97] M. A. Barkatou, An algorithm to compute the exponential part of a formal fundamental matrix solution of a linear differential system, *Appl. Algebra Engrg. Comm. Comput.* 8 (1997) 1–23.
- [BJ97] M. A. Barkatou, F. Richard-Jung, Formal Solutions of Linear Differential and Difference Equations, *Programming and Computer Software* 23 (1997), 17–30.
- [BP99] M. A. Barkatou, E. Pflügel, An algorithm Computing the Regular Formal Solutions of a System of Linear Differential Equations, *J. Symbolic Comput.* 28, (1999), 569–587.

- [CL] E. A. Coddington, N. Levinson, *Theory of Ordinary Differential Equations*, McGraw-Hill Book Company, New York 1955.
- to be completed [Co3] E. Corel, Algorithmic computation of exponents for linear differential systems, ce volume !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.
- [DDT82] J. Della Dora, C. Dicrescenzo, E. Tournier An algorithm to obtain formal solutions of a linear homogeneous differential equation at an irregular singular point, in : *Computer algebra*, Marseille 1982, Lecture Notes in Comput. Sci. 174, Springer-Verlag, 273–280.
- [H87] A. Hilali, *Solutions formelles de systèmes différentiels linéaires au voisinage d'un point singulier*, Thèse de l'Université Joseph Fourier, Grenoble, France, Juin 1987.
- [Ince] E. L. Ince, *Ordinary Differential Equations*, 1926 ; réédition : Dover Publications Inc., 1956.
- [Le91] A. H. M. Levelt, Stabilizing differential operators : a method for computing invariants at irregular singularities, in : *Differential equations and computer algebra*, Comput. Math. Appl., Comput. Math. and Appl., Academic Press, London 1991, 181–228.
- [LS85] D. A. Lutz, R. Schäfke, On the identification and stability of formal invariants for singular differential equations, *Linear Algebra Appl.* 72 (1985) 1–46.
- [M79] B. Malgrange, Sur la réduction formelle des équations différentielles à singularités irrégulières, Prépublication de l'Université Joseph Fourier, Grenoble, France, 1979.
- [Mo60] J. Moser, The order of a singularity in Fuchs' theory, *Math. Z.* 72 (1960), 379–398
- [S86] L. Sauvage, Sur les solutions régulières d'un système d'équations différentielles, *Ann. Sci. École Norm. Sup.* (3) (1886), 391–404.
- [Sib] Y. Sibuya, *Linear differential equations in the complex domain : problems of analytic continuation*, translated from Japanese by the author, Transl. Math. Monogr. 82, Amer. Math. Soc., Providence, R.I., 1990.
- [T55] H. L. Turrittin, Convergent solutions for ordinary linear homogeneous differential equations in the neighborhood of an irregular singular point, *Acta Math.* 93 (1955), 27–66.
- [Was] W. Wasow, *Asymptotic expansions for ordinary differential equations*, Interscience, New York 1965 ; reprint R. E. Krieger Publishing Co., 1976.

Sur l'algorithme du vecteur cyclique

- [B93] M. A. Barkatou, An algorithm for computing a companion block diagonal form for a system of linear differential equations, *Appl. Algebra Engrg. Comm. Comput.* 4 (1993), 185–195.
- [Bi30] G. D. Birkhoff, Formal theory of irregular linear difference equations, *Acta Math.* 54 (1930), 205–246.
- [CK00] R. C. Churchill, J. J. Kovacic, *Cyclic vectors*, Differential Algebra and related topics, Newark, N.J., 2000, World Sci. Publishing, 2002, 191–218.
- [C36] F. T. Cope, Formal solutions of irregular differential equations, Part II, *Amer. J. Math.* 58 (1936), 130–140.

- [D70] P. Deligne, *Équations différentielles à points singuliers réguliers*, Lecture Notes in Math. 163, Springer-Verlag, 1970.
- [K87] N. M. Katz, A simple algorithm for cyclic vectors, *Amer. J. Math.* 109 (1987), 65–70.
- [R84] J.-P. Ramis, *Théorèmes d'indices Gevrey pour les équations différentielles ordinaires*, Mem. Amer. Math. Soc. 296, vol. 48, Amer. Math. Soc., Providence, R.I., 1984.