

CALCUL FORMEL OU CALCUL NUMÉRIQUE? QUELQUES COMMENTAIRES À CE SUJET

Jean THOMANN ¹

Irma

Avertissement: Le but de cet article est de mettre en évidence, sur quelques exemples d'abord élémentaires, puis un peu plus élaborés, l'articulation entre différentes méthodes de calcul appelées traditionnellement formelles ou numériques.

L'idée est de montrer que ces méthodes font partie d'un tout avec comme finalité de rendre effective une partie des mathématiques.

Les renvois sont destinés à permettre au lecteur de préciser et d'approfondir les questions abordées.

1 Préliminaires

Comment définir le Calcul Formel?

Le rêve serait de définir sur ordinateur des objets mathématiques munis de leurs classifications, de leurs structures et d'automatiser les différents opérateurs et applications sur ces objets par des algorithmes.

Cette ambition est destinée à rencontrer assez rapidement des limites, ne serait-ce que parce que la machine n'est qu'un automate fini. Pour illustrer l'évolution dans la réalisation partielle et progressive de ce rêve, il faut d'abord mettre en évidence quelques caractéristiques de ce qu'on appelle le *calcul numérique classique*.

Les méthodes de ce domaine ont évolué de manière spectaculaire en gain de précision, de vitesse par la vectorisation, la parallélisation, mais sont restées basées sur deux types de données :

- les entiers négatifs, positifs, nuls, représentés par un nombre fixe de positions binaires (32 par exemple);
- les nombres représentés par le système de *Virgule Flottante Normalisée* (VFN).

Un nombre en VFN binaire est structuré en *mot* contenant un nombre fixe de positions binaires (32 ou 64 en général) organisés schématiquement suivant la liste (0 ou 1, m positions binaires, 0 ou 1, c positions binaires) qui représente (le signe du nombre, la *mantisse* de longueur m ou partie fractionnaire, le signe de l'exposant de 2, la *caractéristique* de longueur c ou exposant) par exemple (1,101110,0,101) donne $-0,101110 \times 2^{+101}$, la première position binaire de la mantisse étant toujours égale à 1.

Cette dernière normalisation permet d'avoir toujours une précision maximale de m positions binaires significatives de la mantisse. En effet, la possibilité de faire *flotter* la virgule en translatant la valeur de l'exposant permet ainsi d'avoir cette configuration de 0,..., puis le premier chiffre significatif en tête.

1. © L'OUVERT 98 (2000)

Ce système de représentation, un peu simplifié pour des raisons de concision (on peut entre autres utiliser le système hexadécimal à la place du binaire; d'autre part, on utilise la complémentarité à 2 pour les nombres négatifs) entraîne des limitations évidentes que nous allons illustrer sur quelques exemples.

Une première évidence est que ce système ne permet pas de représenter le corps des nombres rationnels $\mathbb{Q}$.

Comme les nombres, représentés en binaire en machine, sont visibles sur écran sous forme convertie en système décimal, par exemple sous forme VFN:

$$-0,2372 \times 10^{+23},$$

nous allons illustrer quelques comportements en utilisant la VFN décimale, en n'oubliant pas qu'avant les calculs, ces nombres sont convertis en VFN binaire.

D'autre part, les seuls opérateurs utilisés sont les opérateurs d'addition, de multiplication, de division, de test de positivité ou de négativité; le test d'égalité n'étant significatif que sur les entiers (on va voir pourquoi). Ces premiers exemples permettent de voir comment le calcul formel permet de franchir ces premiers obstacles, puis d'évoluer dans un système de données infiniment plus varié sur lequel on peut établir des structures algébriques et fonctionnelles.

2 Quelques lacunes du système de virgule flottante

Exemple 1

Dans un système VFN décimal avec une mantisse de longueur 2, les nombres positifs accessibles sont:

$$\begin{array}{cccccc} \vdots & 0,10 \times 10^{-1} & \underline{0,10 \times 10^0} & 0,10 \times 10^{+1} & \underline{0,10 \times 10^{+2}} & \vdots \\ \vdots & 0,11 \times 10^{-1} & 0,11 \times 10^0 & 0,11 \times 10^{+1} & 0,11 \times 10^{+2} & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & 0,99 \times 10^{-1} & 0,99 \times 10^0 & 0,99 \times 10^{+1} & 0,99 \times 10^{+2} & \vdots \end{array}$$

La première constatation est que la représentation de $\mathbb{R}^+$ par ces nombres est de plus en plus dense en s'approchant de l'origine et de moins en moins dense en s'éloignant vers l'infini.

La deuxième remarque concerne la somme des 2 éléments $0,10 \times 10^0$ et $0,10 \times 10^2$, qui est $0,10 \times 10^2$ dans ce système. En effet, pour effectuer cette somme le système aligne les 2 nombres sur l'exposant 2 et l'opération se déroule ainsi:

$$\begin{array}{r} 0,10 \times 10^2 \\ + \quad 0,0010 \times 10^2 \\ \hline 0,10 \times 10^2 \end{array}$$

Même si le résultat dans le registre d'addition peut comporter provisoirement 4 chiffres significatifs, le résultat sera ensuite tronqué à $0,10 \times 10^2$.

Autrement dit, l'addition est inopérante sur ces deux nombres.

Ce phénomène, appelé l'erreur de *troncature* ou d'*arrondi*, est à l'origine de beaucoup de catastrophes, notamment quand on opère sur des nombres d'ordres de grandeur très différents et quand le nombre d'opérations est élevé.

Exemple 2

Le premier exemple peut être critiquable par le fait que ce système à deux chiffres significatifs est vraiment très fruste. On peut rétorquer que l'évaluation du polynôme pourtant peu excentrique:

$$1 + x + x^2 + \cdots + x^7 + x^8 + x^9$$

en $x = 0,1$ est égale à l'évaluation du polynôme:

$$1 + x + x^2 + \cdots + x^6$$

en cette même valeur de x (même en utilisant le schéma de Horner), avec un langage évolué comme FORTRAN, en précision ordinaire.

En effet, avec une mantisse de longueur 6, ce qui correspond à ce qu'on appelle la *simple précision*, l'addition de 1 à 10^{-7} donne 1, suivant le principe du système VFN exposé dans l'exemple 1.

Ces difficultés apparaissant dans les expressions polynomiales laissent présager des bien plus importantes dans les expressions exponentielles! On voit dès maintenant quelle sera la difficulté de reproduire et de contrôler la croissance exponentielle.

Exemple 3

Suivant le même principe, on voit que l'opération d'addition:

$$1 + 0,009 + 0,009 + 0,009 + 0,009 + 0,009$$

dans un système VFN à trois chiffres significatifs (mantisse de longueur 3) donne comme résultat 1 si les additions sont effectuées de gauche à droite et 1,04 si les additions sont effectuées de droite à gauche.

Ce qui met fin à la propriété de commutativité de l'opération d'addition dans le système VFN.

Exemple 4

Il n'est pas difficile de trouver la solution $y(x) = \frac{1}{e}e^{-x}$ de l'équation différentielle très élémentaire:

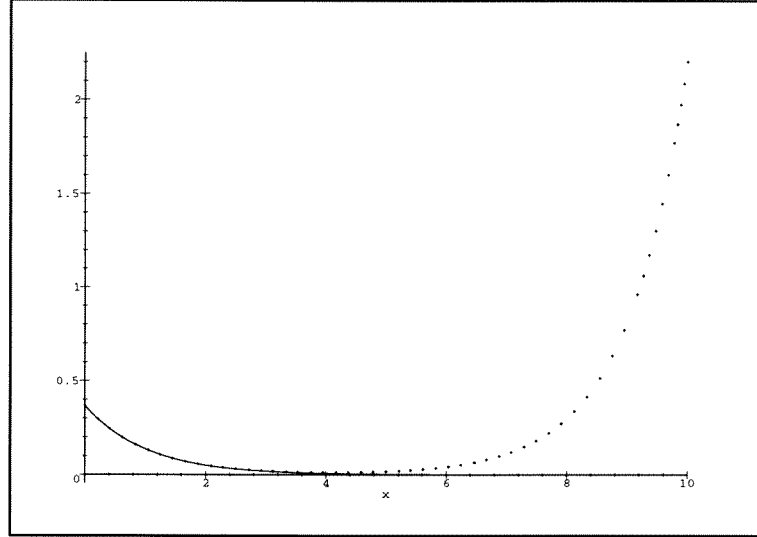
$$y'' - y = 0$$

avec les conditions initiales:

$$\begin{cases} y(0) &= 1/e \\ y'(0) &= -1/e \end{cases}$$

Si on veut vérifier l'efficacité d'une méthode numérique par *pas*, par exemple les méthodes d'Euler, de Runge-Kutta, etc., il faut commencer par donner une valeur numérique à la valeur initiale $1/e$, par exemple 0,3679 (valeur approchée par excès) puis une valeur numérique à $-1/e$, par exemple $-0,3677$ (valeur approchée

par défaut), puis on effectue le calcul *numérique*. On obtient le résultat représenté graphiquement:



où la courbe obtenue numériquement est dessinée par points et la courbe exacte en trait plein.

Même si $1/e$ et $-1/e$ sont approchés avec plus de chiffres significatifs, avec respectivement une erreur par excès et par défaut sur le dernier chiffre significatif, et si on applique une méthode de Runge-Kutta de plus en plus élaborée, le lecteur peut vérifier que ce phénomène (en fait de bifurcation retardée) se reproduira toujours, avec plus ou moins de retard.

Si on regarde cette équation d'un peu plus près on voit que la solution générale de l'équation homogène est:

$$y(x) = ae^{-x} + be^{+x}$$

Les valeurs des paramètres a et b , qui correspondent à la solution particulière avec les conditions initiales imposées, sont solutions du système linéaire:

$$\begin{cases} y(0) &= a + b = 0,3679 \\ y'(0) &= -a + b = -0,3677 \end{cases}$$

c.à d. $a = 0,3678$ $b = 0,0001$.

Donc la solution est:

$$y(x) = 0,3678e^{-x} + 0,0001e^{+x}.$$

Nous constatons que la croissance de la solution numérique vers $+\infty$ est due uniquement à l'erreur de représentation sur $1/e$, qui entraîne dans la solution la présence d'un coefficient b non nul (même s'il est très petit) de la composante fondamentale dominante e^{+x} quand x tend vers $+\infty$.

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

La *vraie* solution $\frac{1}{e}e^{-x}$ est perturbée par la composante ϵe^{+x} .

En posant: $\xi = c - x$, où c est une constante $c \in \mathbb{R}^+$, la solution générale devient: $y(\xi) = ae^{-c+\xi} + be^{+c-\xi}$ et la solution particulière $\frac{1}{e}e^{-x}$ devient $y(\xi) = \frac{1}{e}e^{-c+\xi}$. Ce qui inverse la situation précédente: la solution vraie $\frac{1}{e}e^{-c}e^{+\xi}$ *dominante* cette-fois, sera perturbée éventuellement par une composante *minimale* $\epsilon e^c e^{-\xi}$ qui s'effacera devant la vraie solution quand ξ sera croissante.

L'analyse de ce comportement numérique très répandu dans les équations différentielles (ex: les fonctions spéciales) relève typiquement d'une démarche qui peut s'effectuer à l'aide du calcul formel. Nous y reviendrons.

Exemple 5

Considérons l'équation de récurrence souvent citée de J.M.Muller [9]:

$$a_{n+1} = 111 - \frac{1130}{a_n} + \frac{3000}{a_n a_{n-1}}$$

avec les conditions initiales:

$$\begin{cases} a_0 = 11/2 \\ a_1 = 61/11 \end{cases}$$

Les calculs effectués en VFN avec 8, 16 ou 30 chiffres significatifs montrent que a_n tend toujours vers 100, avec une très bonne stabilité quand n croît. Les nombres rationnels sont alors représentés par des valeurs approchées en VFN.

Plus loin nous constaterons que si on effectue le calcul avec des nombres rationnels, la limite sera 6 quand n tend vers l'infini. Ce qui pose un problème.

Cet exemple est cité par D.Duval dans un article, paru deux fois dans la revue *La Recherche*, et donnant une très intéressante introduction au Calcul symbolique[18,19].

Exemple 6

Un exemple trivial, mais significatif est donné par l'évaluation de l'expression:

$$y : = d \times (a - b + c)$$

avec un système VFN à 3 chiffres significatifs où:

$$d = 100, a = 321, b = 321, c = 0,88.$$

En procédant comme dans l'exemple 1, $a - b = 0$, $a - b + c = 0,88$ d'où $d \times (a + b - c) = 88$.

Dans cette expression, l'ordre des opérations peut être effectué différemment de l'exploration gauche-droite, par exemple si on utilise un *optimiseur*. Si le programme comporte l'expression $h : = a + c$ puis l'expression $y : = d \times (a - b + c)$, l'optimiseur évalue d'abord $h : = 321 + 0.88$, c.à.d. $h = 321$, puis $y : = d \times (h - b)$ c.à.d. $d \times (321 - 321)$ et le résultat est 0.

Ce phénomène d'amplification d'une erreur de troncature par un grand coefficient est bien connu dans le calcul de récurrences à 3 termes, très utiles et souvent utilisées pour les calculs de fonctions spéciales, par exemple de polynômes orthogonaux.

Essayons de calculer les fonctions de Bessel $j_n(x)$ (fractionnaires) au voisinage de l'origine.

Pour $n = 0$ et $n = 1$, les conditions initiales:

$$\begin{cases} j_0(x) = \frac{\sin(x)}{x} \\ j_1(x) = \frac{\sin(x)}{x^2} - \frac{\cos(x)}{x} \end{cases}$$

permettent d'amorcer l'équation de récurrence sur n :

$$j_{n+1}(x) = \frac{(2n+1)}{x} j_n(x) - j_{n-1}(x)$$

En fait, cette équation de récurrence est satisfaite à la fois par la fonction de Bessel fractionnaire $j_n(x)$ et par la fonction de Neumann $y_n(x)$ (deux solutions fondamentales d'une équation aux différences finies d'ordre 2). Au voisinage de $x = 0$, par exemple pour $x \leq 0.1$, la valeur de $j_0(x)$ est entachée d'une erreur importante en VFN (indétermination), la valeur de $j_1(x)$ est entachée d'une erreur encore plus importante (singularité en 0). Dans la récurrence, la valeur de $j_n(x)$, qui provient d'une différence fortement tronquée en VFN, est multipliée par le terme $\frac{2n+1}{x}$ qui devient de plus en plus grand quand n croît, tout comme dans le petit exemple trivial ci-dessus. La propagation de cette erreur a pour effet que $j_n(x)$ va tendre vers $+\infty$ si n tend vers $+\infty$, alors que $j_n(x)$ doit théoriquement tendre vers 0 si n tend vers $+\infty$ [1 p.438].

En fait, la propagation de l'erreur d'origine propre à la VFN, est amplifiée par une instabilité essentielle du même type que dans l'exemple 4.

En effet la récurrence, valable également pour la deuxième solution fondamentale $y_n(x)$, qui, elle, tend vers $-\infty$ quand n tend vers $+\infty$ au voisinage de 0, introduit une *perturbation dominante*: $j_n(x) + \epsilon y_n(x)$ où $\epsilon \neq 0$.

Il est bien connu, qu'une fois de plus, il faut inverser la situation, en évaluant les $j_n(x)$ par récurrence descendante et normalisation.

3 Ces lacunes revues sous l'angle du calcul formel

Le premier point qui ressort des exemples précédents est la nécessité de pouvoir effectuer un calcul *exact*. En 1966, au colloque national d'analyse numérique (Lille), le professeur d'astronomie P. Bachus (professeur à Strasbourg puis à Lille) a présenté le premier algorithme effectif sur ordinateur d'inversion exacte de la matrice de Hilbert.

Cette matrice H , dont les éléments: $a_i^j = \frac{1}{i+j-1}$ sont rationnels, et qui apparaît naturellement dans bien des résolutions numériques (par exemple les équations normales dans les approximations au sens des moindres carrés de fonctions par des polynômes: $a_n x^n + \dots + a_0$ sur l'intervalle $(0,1)$) est très mal conditionnée. On sait que la résolution numérique d'un système linéaire avec une matrice de Hilbert d'ordre 10 peut entraîner une erreur relative de l'ordre de 10^{13} fois l'erreur sur le second membre! La solution exacte consiste évidemment à travailler avec des nombres rationnels, donc dans le *corps* $\mathbb{Q}$ pour trouver les éléments de H^{-1} , qui sont d'ailleurs

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

des entiers. En 1966, implanter une structure de corps $\mathbb{Q}$ était loin d'être facile (avec tous les problèmes de simplification des fractions, donc de factorisation dans $\mathbb{Z}$).

Depuis, le fait d'évoluer dans une vraie structure d'anneau ou de corps, par conséquent de manière exacte, a constitué le progrès essentiel apporté par le calcul formel. Du calcul approché en VFN, on est passé au calcul algébrique (Computer Algebra). Cette fois les entiers sont de vrais éléments de l'anneau euclidien $\mathbb{Z}$ (avec une petite restriction toutefois: l'ordre de grandeur des éléments étant limité par la capacité mémoire de la machine et non plus par la longueur du mot binaire fixe).

Les polynômes ne sont pas seulement évalués numériquement, mais traités comme des éléments par exemple de l'anneau euclidien $\mathbb{Z}[x]$. On peut chercher le PGCD de deux polynômes. Un processeur formel *rigoureux* comme ALDOR interdira d'ailleurs l'inversion d'un tel élément, tant qu'on ne sera pas placé dans la structure de corps des fractions de polynômes.

La démarche d'analyse du comportement numérique de l'exemple 4 consistait en fait à effectuer un calcul formel sur un exemple trivial. En effet, un processeur formel donne comme solution de l'équation: $y'' - y = 0$, la liste de 2 solutions fondamentales sous forme de fonctions élémentaires (e^{-x}, e^{+x}). Chercher la solution particulière vérifiant les conditions initiales:

$$\begin{cases} y(0) &= 1/e \\ y'(0) &= -1/e \end{cases}$$

consiste à chercher les coefficients a et b de la solution générale:

$$y(x) = ae^{-x} + be^{+x}$$

en dérivant d'abord formellement $y(x)$:

$$y'(x) = -ae^{-x} + be^{+x}$$

puis en résolvant le système linéaire:

$$\begin{cases} y(0) &= 1/e = a+b \\ y'(0) &= -1/e = -a+b \end{cases}$$

avec le *paramètre* e et les variables inconnues a et b . D'où: $a = \frac{1}{e}$ et $b = 0$.

Ainsi le calcul formel donne le résultat exact:

$$y(x) = \frac{1}{e}e^{-x} + 0e^{+x},$$

mais nous renseigne également sur l'origine de l'instabilité mathématique due à la confrontation d'une solution *dominante* e^{+x} (pour $x \geq 0$) et d'une solution *minimale* e^{-x} . Il nous indique également le remède de stabilisation des calculs par changement de variable: $x = c - \xi$. Nous avons donc pu *contrôler* le phénomène de croissance et de décroissance exponentielle. Il est important de noter que c'est la connaissance des deux fonctions explicites données par le calcul formel, à la place de courbes intégrales calculées par points, qui permet d'analyser le *comportement* des solutions.

Une fois de plus, on peut reprocher à cet exemple le fait d'être simpliste, mais cette démarche d'analyse de comportement, de recherche de fonctions élémentaires ou spéciales explicites, peut se pratiquer sur des exemples bien plus sophistiqués, comme on le verra par la suite dans l'analyse d'une singularité.

À titre d'exercice on peut essayer de résoudre une équation différentielle un peu plus évoluée:

$$y''(x) + a\left(\frac{1}{3} - 2a\right)y'(x) - \frac{2a^3}{3}y(x) = 0$$

où a est un paramètre. Un processeur formel donne deux solutions fondamentales:

$$[e^{2a^2x}, \quad e^{-\frac{1}{3}ax}].$$

En introduisant les conditions initiales:

$$\begin{cases} y(0) = a - 2 + \pi \\ y'(0) = 2a^2(a - 2) - a\pi/3 \end{cases}$$

formellement, puis numériquement pour $a = 2$, on constatera le même phénomène de perturbation par la solution fondamentale dominante e^{2a^2x} .

On remarquera qu'une recette de vérification de stabilité de méthode d'intégration numérique est de pratiquer l'intégration à *rebours* à partir de l'extrémité de l'intervalle d'intégration et de constater le recouvrement ou le non-recouvrement des deux courbes intégrales.

L'exemple de la récurrence de J.M.Muller [18,19] est encore plus significatif. Cette récurrence détermine les termes de la suite a_n :

$$a_{n+1} = 111 - \frac{1130}{a_n} + \frac{3000}{a_n a_{n-1}}$$

avec les conditions initiales:

$$\begin{cases} a_0 = 11/2 \\ a_1 = 61/11 \end{cases}.$$

En disposant d'un processeur formel, on peut obtenir la limite 6 en calculant dans la structure du corps $\mathbb{Q}$, mais on peut aussi déterminer que ce système à 3 points fixes en factorisant le polynôme $P(x) = x^3 - 111x^2 + 1130x - 3000 = 0$ en produit $(x - 5)(x - 6)(x - 100)$.

Nous remercions J.M.Muller, de l'ENS Lyon, de nous avoir communiqué l'origine et la description complète de son exemple. Il est construit à partir de la récurrence linéaire:

$$u_{n+1} = 111u_n - 1130u_{n-1} + 3000u_{n-2}.$$

Les solutions d'un tel système sont de la forme:

$$u_n = \alpha_0(\lambda_0)^n + \alpha_1(\lambda_1)^n + \alpha_2(\lambda_2)^n$$

où $\lambda_0, \lambda_1, \lambda_2$ (on suppose que $|\lambda_0| > |\lambda_1| > |\lambda_2|$) sont les racines $\lambda_0 = 100$, $\lambda_1 = 6$, $\lambda_2 = 5$, du polynôme $P(x)$.

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

La suite initiale est obtenue par le rapport de 2 termes consécutifs:

$$a_n = \frac{u_{n+1}}{u_n} = \frac{\alpha_0 100^{n+1} + \alpha_1 6^{n+1} + \alpha_2 5^{n+1}}{\alpha_0 100^n + \alpha_1 6^n + \alpha_2 5^n}.$$

On voit que, si $\alpha_0 \neq 0$, $a_n \rightarrow 100$ si $n \rightarrow +\infty$. Il suffit alors de choisir a_0 et a_1 de sorte que: $\alpha_0 = 0$ (dans notre cas: $a_0 = 11/2$ et $a_1 = 61/11$), et les valeurs de α_1 et de α_2 correspondantes (ici $\alpha_1 = 1$ et $\alpha_2 = 1$).

Tant qu'on effectue les calculs dans le corps $\mathcal{Q}$, on reste sur la trajectoire *exacte* parcourue par a_n , mais dès qu'on commet une erreur (aussi petite soit-elle) de représentation ou d'arrondi sur les nombres rationnels (notamment en VFN), on est *dévié* sur une trajectoire où α_0 est peut-être très petit mais non nul. D'où la convergence vers 100.

J.M.Muller suggère de fabriquer autant d'exemples qu'on veut à partir d'un choix de: $\lambda_0, \lambda_1, \lambda_2$ où λ_0 est grand devant λ_1 et λ_2 .

4 Le calcul algébrique

1. Le calcul algébrique exact

Des différents exemples précédents il découle la nécessité d'avoir des outils qui permettent non seulement de travailler de manière exacte, mais encore dans de vraies structures de groupes, d'anneaux, de corps, qui permettent l'utilisation de paramètres ou de nombres algébriques, et qui permettent d'utiliser des fonctions et applications connues ou d'en fabriquer de nouvelles.

La difficulté est de faire correspondre à ces objets mathématiques abstraits des objets qui ont une *forme* de représentation qui peut ensuite être réduite en structures de données sur machine.

On peut imaginer assez facilement comment on peut passer de la structure des entiers à longueur binaire fixe à une structure d'anneau intègre commutatif des entiers à longueur variable, puis construire un algorithme de factorisation unique et un algorithme d'Euclide, encore faut-il utiliser des techniques appropriées.

Un pas décisif du calcul formel a été franchi avec la manipulation des listes de symboles (notamment avec le langage LISP). Soit, par exemple, l'objet symbolique $\frac{2}{3}x^3$ c.à.d. un monôme en x d'ordre 3 avec le coefficient $\frac{2}{3}$. Cet objet mathématique peut avoir une représentation par la liste $(\frac{2}{3}, x, 3)$. On peut lui associer une nouvelle liste

$$(2, x, 2)$$

où le premier élément est $2 = 3 \times \frac{2}{3}$ dans le corps $\mathcal{Q}$ et le troisième élément est $2 = 3 - 1$ dans l'anneau $\mathbb{Z}$.

Cette règle: $(a, x, n) \rightarrow (na, x, n-1)$ appliquée à la liste de 3 éléments (a, x, n) où $a \in \mathcal{Q}$, $n \in \mathbb{Z}$ et x est un symbole choisi, définit la règle de dérivation des monômes en x , qui à ax^n où $a \in \mathcal{Q}$, fait correspondre nax^{n-1} .

Une autre règle qu'on peut appliquer est de faire correspondre à la chaîne de symboles $\text{Log}(x)$ une autre chaîne: $1/x \dots$. Le calcul formel permet ainsi de manipuler, suivant des règles ou des consultations de dictionnaires, des listes de symboles et d'éléments définis précédemment.

Parmi différentes formes *équivalentes*, il faut choisir des représentants particuliers des objets algébriques. Pour le polynôme: $(x+1)(x-1)$ on peut choisir $x^2 - 1$ ou plutôt $-1 + x^2$. Ce choix consiste à définir une *forme normale*. Si l'on veut effectuer l'addition de deux polynômes de $A[x]$ où A est un anneau déjà défini, on peut alors effectuer directement la somme deux à deux des éléments correspondants dans la *structure de données* des deux listes des coefficients appartenant à A .

Par exemple, la somme des 2 polynômes de $\mathbb{Z}[x]$: $(x+1)(x-1)$ et $2x(x-3) + 5$ consistera à calculer la somme: $-1 + x^2 + 5 - 6x + 2x^2$; c.à.d. dans les 2 listes de coefficients de $\mathbb{Z}$: $(-1, 0, 1)$ et $(5, -6, 2)$ on effectuera la somme des éléments 2 à 2 et on trouvera: $(4, -6, 3)$. Donc la somme des 2 polynômes sera: $4 - 6x + 3x^2$.

Le choix d'une forme normale, associée à un objet mathématique abstrait, permet de constater que deux objets sont *égaux*, c.à.d. que le *test de nullité* de leur différence est possible [3].

On peut définir ainsi des formes normales de polynômes à plusieurs variables, par exemple: $x^2 - y^2$ à la place de $(x-y)(x+y)$. Cette fois la forme normale correspond à l'ordre lexicographique sur les indéterminées.

Beaucoup d'expressions se ramènent à une forme polynomiale. Par exemple, l'expression: $(\cos a - \sin b)(\cos a + \sin b) = \cos^2 a - \sin^2 b$ n'est autre que le polynôme précédent où $x = \cos a$ et $y = \sin b$.

Une des difficultés en calcul formel est d'ailleurs la *simplification* des expressions. En effet, $x - 1$ est plus *simple* que $(x^2 - 1)/(x + 1)$, mais $x^{999} - x^{998} + x^{997} \dots - 1$ n'est pas plus *simple* que $(x^{1000} - 1)/(x + 1)$.

En pratique, il existe 2 types de structures de données pour les formes polynomiales: la structure *creuse* et la structure *dense*:

- la structure creuse consiste à représenter: $1 + 3x^2$ comme $((0, 1), (2, 3))$, c.à.d. tous les couples (i, a_i) pour les termes $a_i x^i$ non nuls;
- la structure dense représente le même polynôme par $(1, 0, 3)$, c.à.d. par la liste de tous les coefficients a_i .

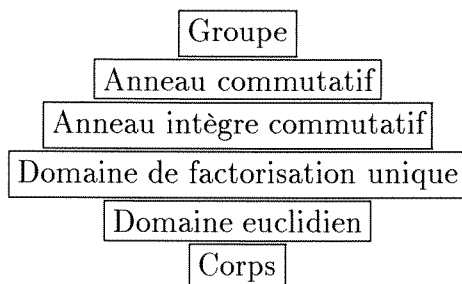
Si on veut vérifier que: $(x^{1000} + 1)(x^{1000} - 1) = x^{2000} - 1$, une méthode dense nécessite un million de multiplications, tandis qu'une méthode creuse n'en nécessite que quatre, comme dans $(x+1)(x-1) = x^2 - 1$.

Lorsque les représentations des objets mathématiques sont définies, on peut construire les algorithmes, par exemple l'algorithme d'Euclide. Mais cet algorithme doit être construit une fois pour toutes, c.à.d. il doit permettre de calculer le $\text{PGCD}(a, b)$ où a et b sont des éléments d'un domaine euclidien quelconque.

Dans le schéma ainsi défini, les exécutions des opérations élémentaires de cet algorithme sont ensuite concrétisées selon la spécificité du domaine euclidien choisi. Par exemple, si a et b appartiennent à $\mathbb{Q}$, les multiplications se font dans $\mathbb{Q}$, tandis que si a et b appartiennent à $\mathbb{Q}[x]$, les multiplications se font dans la structure de $\mathbb{Q}[x]$,

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

qui fait d'ailleurs appel elle-même à la multiplication dans $\mathcal{Q}$. En fait, l'algorithme d'Euclide est construit pour deux éléments appartenant à la *catégorie* indifférenciée des domaines euclidiens. Quand cet algorithme se déroule avec deux éléments d'un domaine euclidien particulier (par exemple $\mathcal{Q}[x]$) les opérations se font dans ce domaine spécifique. Ceci implique une structure hiérarchique des catégories entre elles, suivant, par exemple, le schéma suivant:



Ce schéma n'est qu'une petite partie du schéma de structure général, que l'utilisateur peut agrandir à son gré. Ainsi, si D est un domaine de factorisation unique, alors $D[x]$ est un domaine de factorisation unique dont les éléments premiers sont des polynômes irréductibles.

Une partie essentielle d'un processeur formel est évidemment constitué par les algorithmes de factorisation, par exemple la factorisation du polynôme de Wilkinson:

$$x^{20} + 210x^{19} + \cdots + 20! = (x+1)(x+2) \cdots (x+20),$$

qui jouent un rôle central dans la réduction des fonctions rationnelles.

Dans l'étude des singularités, nous utiliserons des algèbres plus évoluées, notamment des algèbres non commutatives dont les éléments sont des opérateurs différentiels linéaires à coefficients polynomiaux, ou rationnels.

Par exemple, si on note un opérateur élémentaire de dérivation par D , où

$$D = \frac{d}{dx},$$

un premier opérateur est défini par

$$op1 = 3x^2 D^2 + 2D + \frac{1}{x}.$$

D^2 est en fait l'opérateur $D \times D$, c.à.d. d^2/dx^2 . Un deuxième opérateur est défini par

$$op2 = op1 \times (5xD + 7).$$

L'élément $op2$ est évalué formellement:

$$op2 = 15x^3 D^3 + (51x^2 + 10x) D^2 + 29D + \frac{7}{x}.$$

On peut faire agir ces opérateurs sur des polynômes, des fraction rationnelles, ou des séries formelles. Si on fait agir les opérateurs précédents sur:

$$f = \frac{x^4 + 1}{x^2},$$

on trouve que:

$$(op1 \times op2 - op2 \times op1)(f) = \frac{75x^4 - 540x + 75}{x^4}$$

et on constate que la loi de multiplication des opérateurs n'est pas commutative. On voit quel est l'intérêt d'un algorithme de factorisation à gauche ou à droite dans une telle algèbre non commutative.

Les séries formelles peuvent être considérées elles-mêmes comme des éléments d'une algèbre particulière, comme l'algèbre des séries Gevrey: $\sum_n a_n x^n \in \mathcal{Q}[i][[x]]$, dont les coefficients a_n sont tels que:

$$|a_n| < C A^n \Gamma(1 + \frac{n}{k}) \quad n \in \mathbb{N}, \quad A \in \mathbb{R}^+, \quad C \in \mathbb{R}^+, \quad k \in \mathbb{Q}.$$

Dans l'analyse des singularités, qui sera exposée plus loin, les séries formelles seront construites de la manière suivante: une liste de m conditions initiales: $(a_0, \dots, a_{m-1})$, où $a_k \in \mathcal{Q}[i]$ (i désigne l'élément imaginaire), suivi d'un opérateur aux différences finies: $\Delta = p_0(j)a_j + \dots + p_m(j)a_{j+m}$ où $p_k(j) \in \mathcal{Q}[i][j]$.

Un coefficient a_n , où $n \in \mathbb{N}$ est ainsi engendré par m valeurs précédentes (au début par les conditions initiales), à l'aide de l'équation aux différences finies:

$$p_0(n-m)a_{n-m} + \dots + p_m(n-m)a_n = 0.$$

Les opérateurs des équations différentielles linéaires et les opérateurs des équations aux différences linéaires peuvent être traités dans un cadre unifié. Ils sont considérés comme éléments d'algèbres particulières non commutatives d'Ore [10,11,12], où les transformations de Mellin constituent le pont entre ces 2 types d'opérateurs et d'algèbres.

Un cas particulier d'algèbre d'Ore est l'algèbre de Weyl des opérateurs aux dérivées partielles à coefficients polynomiaux. Il existe bien d'autres structures d'algèbres (algèbres de Lie, de Grassmann, etc...) dans les bibliothèques de calcul formel.

Dans ce cadre algébrique, on peut également inclure l'utilisation des *paramètres*. Un exemple significatif sur l'apport du calcul formel au calcul numérique *classique* est le système linéaire suivant [2]: soit la matrice

$$\begin{bmatrix} 1 - \epsilon & 2 & -4 \\ 3/2 - \epsilon & 3 & -5 \\ 5/2 + \epsilon & 5 & -7 \end{bmatrix}$$

où ϵ est un paramètre. La résolution formelle du système: $Ax = b$ où: $b = [1, 2, 3]$, donne comme résultat:

$$\begin{cases} x_1 = \frac{-1}{2\epsilon} \\ x_2 = \frac{1+7\epsilon}{4\epsilon} \\ x_3 = \frac{3}{4} \end{cases}$$

En évaluant le résultat pour $\epsilon = 10^{-20}$, on trouve:

$$x_1 = -50000000000000000000 \quad x_2 = 100000000000000000007/4 \quad x_3 = 3/4$$

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

En effectuant la résolution numériquement en VFN (normalement avec une mantisse de 18 chiffres décimaux en *double precision*), il faudrait d'abord évaluer les éléments de matrice en remplaçant ϵ par sa valeur et, par l'effet des erreurs d'arrondi, la matrice serait singulière.

Un autre exemple de calcul avec paramètre consiste en l'évaluation du rang de la matrice:

$$\begin{bmatrix} 1 & 1 \\ a & 1 \end{bmatrix}$$

en fonction de a .

Ce rang vaut 1 si $a = 1$, sinon le rang vaut 2.

La gestion des nombres algébriques en calcul formel est analogue à la gestion des paramètres. En effet, l'équation $x^2 - 2 = 0$ admet $+\sqrt{2}$ et $-\sqrt{2}$ comme solutions. En pratique, on définit ces solutions par le paramètre a qui est tel que, si $x = a$, $x^2 - 2 = 0$, sinon $x^2 - 2 \neq 0$.

La représentation des racines par les radicaux $\sqrt[n]{}$ est en fait une représentation symbolique particulière, qui peut être utilisée pour les équations polynomiales d'ordre inférieur à 5, d'après la théorie de Galois. Mais dans ce cas, comme dans le cas d'ordre supérieur, un nombre algébrique est caractérisé par le seul fait qu'il vérifie une équation polynomiale à coefficients appartenant à $\mathbb{Z}$ ou eux-mêmes algébriques.

La gestion des nombres algébriques doit être effectuée à plusieurs niveaux. En effet, ils peuvent être racines d'un premier polynôme dont les coefficients sont eux-mêmes racines d'un deuxième polynôme, dont les coefficients sont racines d'un troisième polynôme à coefficients dans $\mathbb{Z}$, etc... Pour effectuer des calculs qui dépendent d'un certain nombre de paramètres ou de nombres algébriques, il faut faire appel à des algorithmes élaborés permettant de réaliser des évaluations dans une *clôture algébrique* ou plus généralement dans une *clôture dynamique constructible* [14,20,21]. Ces algorithmes, qui, depuis une vingtaine d'années, font l'objet de recherches actives (voir le code D5[14] puis D7 [14,20,21], les travaux des équipes de recherche du LACO de Limoges, du LMC de Grenoble), sont en constante évolution.

En effet, une équation différentielle très simple:

$$(x^2 - 2)y'' + p_1(x)y' + p_2(x) = 0$$

qui a comme équation caractéristique $x^2 - 2 = 0$, a des singularités pour les racines de $x^2 - 2$.

Les nombres algébriques $+\sqrt{2}$ ou $-\sqrt{2}$ vont se retrouver dans toutes les expressions (parties exponentielles, séries, etc...) constitutives des solutions formelles au voisinage de ces 2 singularités. Il faut donc *explorer* toutes les branches de l'arborescence créée par les *cas* constitués par $+\sqrt{2}$ et $-\sqrt{2}$ ou toutes les racines d'un polynôme caractéristique d'ordre éventuellement plus élevé, c-à-d les nombres algébriques en présence.

Peuvent s'y rajouter un certain nombre de paramètres.

Les algorithmes de *clôture dynamique constructible* permettent de construire cette arborescence et de dérouler les algorithmes (par exemple de calcul de rang d'une matrice, de solutions d'équations différentielles, etc...) dans tous les *cas*.

À la fin des exécutions pendant lesquelles interviennent les factorisations, les simplifications des expressions rationnelles, etc...les résultats sont formels, par exemple une série formelle $\sum a_n x^n$ dont les coefficients sont exprimés en fonction de ces nombres algébriques.

2. Les limites du calcul algébrique

À ce moment, se pose un problème fondamental, celui de l'évaluation numérique. En effet, si on est dans le cas où une série formelle à coefficients algébriques est convergente pour une valeur de x (par ex. $x = 0.1$) il faut d'abord *évaluer numériquement* les termes de la série, puis *évaluer* la somme de la série.

Il faut donc franchir la barrière entre l'évaluation exacte, algébrique et l'évaluation numérique approchée avec l'aide de la VFN. En fait, au niveau du calcul scientifique, nous sommes à la limite de l'algèbre et de l'analyse. Le problème immédiat est le problème de la représentation de $\mathbb{R}$ et de $\mathbb{C}$.

Comme on l'a vu, on peut manipuler e ou π comme paramètres dans les expressions algébriques, mais comment évaluer numériquement ces éléments?

Il y a deux solutions possibles: soit évaluer par encadrement de nombres rationnels avec un intervalle d'écart prédéfini, ce qui risque d'être assez lourd pour la suite des calculs *par encadrements*, soit *approcher en VFN* (dont on choisit le nombre de chiffres significatifs) les éléments de $\mathbb{R}$ ou $\mathbb{C}$ (par $a+ib$ dans le cas de $\mathbb{C}$, où $a, b \in \mathbb{R}$). Par exemple, dans le cas de e ou de π , on peut approcher ces valeurs en évaluant des séries formelles tronquées. Mais cette *conversion numérique* doit être effectuée et *contrôlée* proprement.

On voit maintenant à quel point le test de nullité en VFN est aléatoire. En effet, il n'y a aucune chance, même si toutes les précautions sont prises, que $\log(e) - 1$ soit nul, si le $\log$ et e sont calculés par des séries tronquées. Un exemple nous montre immédiatement quelles peuvent être les difficultés rencontrées, notamment quand il s'agit de convertir un nombre algébrique en nombre numérique VFN.

Reprenons le polynôme:

$$W(x) = (x+1)(x+2)\cdots(x+20).$$

Wilkinson a développé ce polynôme:

$$W(x) = x^{20} + 210x^{19} + \cdots + 20!,$$

puis a introduit une perturbation sur le coefficient de x^{19} de l'ordre de *l'erreur de représentation machine* en VFN à mantisse de 24 positions binaires:

$$\overline{W}(x) = W(x) + 2^{-23}x^{19}.$$

Par un procédé de calcul *exact*, Wilkinson a trouvé que $\overline{W}(x)$ avait 10 racines réelles et que les parties imaginaires des autres racines étaient comprises entre 0.8 et 3.

Cet exemple montre bien quelle est l'instabilité des racines d'un polynôme par rapport à la définition des coefficients. Ceci prouve bien qu'un algorithme conçu exclusivement en VFN est voué à l'échec, à cause de la propagation de l'erreur d'approximation numérique qu'accentue ce phénomène d'instabilité mathématique.

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

Le problème très important de calcul des racines réelles de polynômes peut être résolu par un algorithme *mixte formel-numérique* dont nous exposons un schéma résumé. La première phase *formelle* consiste à *isoler* les racines.

En supposant que le polynôme $p[x]$ d'ordre n à coefficients rationnels, ou à coefficients $\in \mathbb{Z}$ (la multiplication des coefficients rationnels par le ppcm des dénominateurs donne une équation polynomiale équivalente) ne comporte pas de racines multiples, on peut construire formellement la *séquence de Sturm* des polynômes. Sinon, on peut réduire formellement la multiplicité des racines à 1.

Cette séquence est au signe près celle obtenue par l'algorithme d'Euclide:

$$\begin{cases} p_0[x] = p[x] \\ p_1[x] = p'[x] \\ \vdots \\ p_i[x] = -\text{reste}(p_{i-2}[x], p_{i-1}[x]) \\ \vdots \\ p_k[x] = \text{constante} \end{cases}$$

où $(\ , \)$ indique la division et $k \leq n$.

En évaluant ces polynômes en une valeur rationnelle de x (qui n'est pas racine), on peut décompter *exactement* le nombre $V(x)$ de changements de signe de cette séquence.

En utilisant le théorème de Sturm, qui permet de dire qu'il y a exactement $|V(p) - V(q)|$ racines réelles entre 2 valeurs rationnelles p et q , on peut par un procédé de dichotomie de l'intervalle (A, B) de 2 nombres rationnels de $\mathbb{R}$, isoler toutes les racines réelles comprises entre A et B .

Ce procédé formel permet, en un nombre fini d'étapes, de déterminer des encadrements rationnels de ces racines simples même si elles sont très proches les unes des autres; ce qui est irréalisable numériquement pour les raisons exposées plus haut.

Ce travail étant achevé, on peut passer à la phase *numérique* qui permet d'une manière rapide d'approcher ces racines isolées auparavant, avec une précision déterminée, par une méthode de type Newton-Raphson. On aurait pu arriver au même résultat en poursuivant le procédé de dichotomie, mais il est bien connu que la convergence de la méthode de Newton est quadratique, alors que la convergence de la méthode dichotomique est très lente.

Nous disposons donc d'une conversion *contrôlée* d'un nombre algébrique en une approximation numérique en VFN, dans le cas réel. Il existe des méthodes analogues dans le cas de racines complexes.

À partir de ce moment on peut évaluer des sommes de séries (par exemple pour évaluer e ou π) ou toute expression contenant un nombre algébrique. L'exemple suivant nous illustre cette double démarche, algébrique et analytique.

Soit un système différentiel non linéaire autonome:

$$\begin{cases} \dot{x}_1 = 3x_1 + x_1x_3 \\ \dot{x}_2 = 7x_2 + 2x_3 + x_1^2 \\ \dot{x}_3 = 3x_2 + 4x_3 + x_2^2 + x_1^3 \end{cases}$$

où x_1, x_2, x_3 sont des fonctions du temps t et où le point indique la dérivation par rapport à t . Ce système a la forme: $\dot{X} = AX + F(X)$ où $F(X)$ désigne la partie non linéaire en x_1, x_2, x_3 .

Pour réduire ce système à une forme plus simple, nous effectuons une première transformation linéaire non singulière: $X = TY$ sur les coordonnées. Donc: $\dot{X} = T\dot{Y}$ et $\dot{Y} = T^{-1}ATY + T^{-1}F(X)$ afin de réduire A à la forme de Jordan: $T^{-1}AT = J$.

Nous obtenons ainsi un nouveau système dans un nouveau système de coordonnées:

$$\dot{Y} = JY + G(Y)$$

où $G(Y)$ est la partie non linéaire. Dans notre cas, J est diagonale:

$$J = \begin{bmatrix} 3 & 0 & 0 \\ 0 & \frac{11}{2} + \frac{1}{2}\sqrt{33} & 0 \\ 0 & 0 & \frac{11}{2} - \frac{1}{2}\sqrt{33} \end{bmatrix}$$

Les éléments diagonaux λ_i , $i = 1, 2, 3$ de J (racines de l'équation caractéristique) sont des nombres algébriques. Dans notre cas, où le nombre d'équations est 3, ils peuvent être exprimés par radicaux.

Si J est diagonale on peut essayer de linéariser ce système ou le mettre sous forme d'un système *le moins non linéaire possible*, grâce à un changement si possible *régulier*. Pour trouver une telle transformation, concrétisée par une série formelle en x_1, x_2, x_3 , la série de Poincaré, convergente ou divergente, il faut analyser si le système est *résonnant* ou non.

On dit qu'un système d'ordre n est résonnant s'il existe des relations du type:

$$\sum_{i=1}^n m_i \lambda_i = \lambda_j \quad \text{où } m_i \in \mathbb{N}, \quad j \in 1, \dots, n, \quad \text{et } \sum_{i=1}^n m_i \geq 2.$$

Or ce test de présence ou d'absence de résonance n'est possible qu'en évaluant ces expressions exactement, c.à.d. avec les nombres algébriques λ_i et des entiers m_i .

Dans notre exemple, il est facile de constater qu'il n'y a pas de résonance, mais dans le cas où la dimension du système est ≥ 4 , ces tests d'égalité, donc de nullité, impossibles en VFN, comme on l'a constaté, ne peuvent se faire que dans le corps des nombres algébriques.

En supposant qu'il n'y a pas de résonances et qu'on n'est pas en présence de *petits dénominateurs* (pour plus de détails, il faudrait développer la théorie des singularités de systèmes dynamiques cf[16,24]), on peut construire une transformation régulière qui linéarise ce système.

Dans ce cas, les solutions du système linéarisé peuvent s'exprimer explicitement sous forme de combinaisons linéaires d'exponentielles du type: $x(t) = \sum_{i=1}^n a_i e^{-\lambda_i t}$. Cette fois, il s'agit de trouver une bonne approximation numérique des valeurs algébriques de λ_i . Nous sommes donc ramenés au problème précédent d'évaluation numérique des racines du polynôme caractéristique.

5 Le calcul contrôlé

Le but de ce paragraphe est de montrer, dans une étude mathématique particulière, quelle est la part respective de ces différents ingrédients *numériques, formels, symboliques, algébriques*.

Nous allons voir aussi comment les résultats formels permettent de prévoir des comportements numériques et finalement de contrôler et de structurer l'ensemble des calculs.

Nous nous proposons d'étudier la singularité à l'origine de l'équation différentielle:

$$4x^5y'' + 2x^2y' + y = 0.$$

Cette étude peut se faire à plusieurs niveaux d'approfondissement:

- Classer la singularité, chercher les invariants formels.
- Calculer les solutions formelles au voisinage de cette singularité.
- Analyser le comportement des solutions au voisinage de la singularité, plus particulièrement dans les différentes directions autour de cette singularité.
- Calculer efficacement ces solutions avec l'aide des informations recueillies aux trois points précédents, c.à.d. établir une correspondance entre les solutions formelles et les vraies fonctions holomorphes qui sont solutions dans des secteurs déterminés, puis évaluer ces fonctions. Dans ce but nous utiliserons les éléments de théorie de la k -sommabilité établie par J.P.Ramis[25,26,28].

1. Classification

Un instrument spécifique formel, lié à cette équation, le *polygone de Newton* (défini par B.Malgrange et J.P.Ramis) nous renseigne sur la nature de la singularité et la forme des solutions.

Définition du polygone de Newton (Newton, Ramis, Malgrange) en $x = 0$, d'un opérateur différentiel à coefficients dans $\mathcal{C}[x]$:

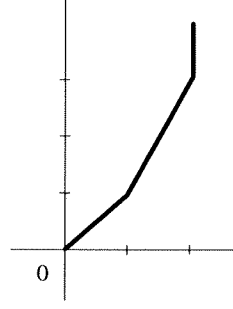
$$L = \sum_{i=0}^{i=n} \sum_{j=0}^m p_{i,j} x^j \frac{d^i}{dx^i}.$$

Si $Q^+(u, v) = \{(x, y) \in \mathbb{R}^2, \ x \leq u, y \geq v\}$, est le second quadrant de $\mathbb{R}^2$ translaté en (u, v) , on définit:

$$M^+(L) = \bigcup_{p_{i,j} \neq 0} Q^+(i, j - i).$$

Le *polygone de Newton* de L est l'enveloppe convexe inférieure de $M^+(L)$. Toutes ses pentes finies sont rationnelles. Voici le polygone de Newton de l'équation:

$$4x^5y'' + 2x^2y' + y = 0 :$$



En-dehors des demi-droites de pente 0 et infinie, il y a 2 segments respectivement de pentes 1 et 2. Le fait qu'il y a au moins une pente (en-dehors de la pente infinie) non nulle, nous indique que nous sommes en présence d'une *singularité irrégulière*: ce qui veut dire, entre autres, que l'une des deux solutions fondamentales formelles indépendantes comportent une partie exponentielle d'ordre 2 (comme la pente la plus forte 2), c.à.d. en $e^{1/ax^2+1/bx}$ $(a, b) \in \mathbb{C}^2$ et que ces solutions comportent également, en général, au moins une série formelle divergente.

Ces séries sont *Gevrey* de classe k et k -sommables (k prenant la valeur 2 de la plus forte pente du polygone de Newton). Rappelons qu'une série formelle: $\sum_n a_n x^n \in \mathbb{C}[[x]]$ est une série formelle de classe Gevrey k (cf. par.4.1) si les coefficients a_n sont tels que:

$$|a_n| < C A^n \Gamma(1 + \frac{n}{k}) \quad n \in \mathbb{N}, \quad A \in \mathbb{R}^+, \quad C \in \mathbb{R}^+, \quad k \in \mathbb{Q}.$$

Si le polygone de Newton n'avait que des segments de pente nulle (en dehors de la demi-droite de pente infinie), nous serions en présence d'une *singularité régulière*, ce qui signifierait que les solutions ne comporteraient pas de parties exponentielles et que les séries apparaissant dans ces solutions seraient toutes convergentes.

Par un calcul purement formel, nous obtenons ainsi un ensemble de renseignements essentiels sur la nature de la singularité, sur la forme des solutions en cette singularité, sur leurs parties exponentielles et sur la structure de leurs séries formelles.

2. Calcul des solutions formelles au voisinage de la singularité

À présent, les solutions fondamentales formelles peuvent être calculées de manière exacte. Ce calcul est effectué en deux temps:

– L'algorithme formel de Newton [3,10,15,36]

La substitution: $\hat{y}(x) = e^{1/ax^2} \hat{\varphi}(x)$ dans l'équation différentielle:

$$4x^5 \hat{y}'' + 2x^2 \hat{y}' + \hat{y} = 0$$

aboutit à la nouvelle équation différentielle:

$$4x^6 \hat{\varphi}'' + 2x^3(1 - \frac{8}{a})\hat{\varphi}' + (\frac{24x^2}{a} + x + \frac{16}{a^2} - \frac{4}{a})\hat{\varphi} = 0$$

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

qui dépend du paramètre a .

Afin de *relever* au maximum la pente du premier segment du polygone de Newton, c.à.d. de la rendre plus proche de l'horizontale, on essaie de maximiser la borne inférieure de $j - i$ (voir la définition du polygone de Newton) pour $i = 0$ dans cette équation de la forme $\sum_{i=0}^n \sum_{j=0}^m p_{i,j} x^j \frac{d^i}{dx^i} = 0$.

Comme $\sum_{j=0}^m p_{0,j} x^j \frac{d^0}{dx^0} = (\frac{24x^2}{a} + x + \frac{16}{a^2} - \frac{4}{a})\hat{\varphi}$, il faut annuler la partie $\frac{16}{a^2} - \frac{4}{a}$, d'où la valeur $a = 4$. Et l'équation différentielle devient *l'équation réduite*:

$$4x^5 \hat{\varphi}'' - 2x^2 \hat{\varphi}' + (6x + 1)\hat{\varphi} = 0$$

vérifiée par $\hat{\varphi}$;

Comme la pente du premier segment n'est encore pas nulle, on réitère dans cette équation réduite la substitution: $\hat{\varphi}(x) = e^{1/bx} \hat{\psi}(x)$, qui conduit à la valeur: $b = -2$ et la nouvelle équation différentielle *réduite*:

$$4x^4 \hat{\psi}'' + (4x^2 - 2x)\hat{\psi}' - (4x - 7)\hat{\psi} = 0,$$

vérifiée par $\hat{\psi}(x)$.

Nous avons donc obtenu, par un algorithme formel, la *partie irrégulière* de $\hat{y}(x) = e^{1/4x^2 - 1/2x} \hat{\psi}(x)$, c.à.d. $e^{1/4x^2 - 1/2x}$ et l'équation différentielle vérifiée par $\hat{\psi}(x)$. Le polygone de Newton de cette équation réduite a maintenant un segment de pente nulle.

Nous constatons que les paramètres a et b sont solutions d'équations polynomiales (appelées *équations caractéristiques associées* aux différentes pentes) dont les coefficients appartiennent à $\mathbb{Q}$. Dans le cas général, ces racines sont des nombres algébriques.

– L'algorithme formel de Frobenius [3,10,15,36]

Cet algorithme consiste à calculer la solution formelle correspondant au segment de pente nulle du polygone de Newton. Il s'applique d'emblée dans le cas d'une singularité régulière.

La substitution $\hat{\psi}(x) = x^\mu \hat{g}(x)$ dans l'équation différentielle vérifiée par $\hat{\psi}(x)$ conduit à la résolution algébrique de l'équation polynomiale *indicielle* $7 - 2\mu = 0$, dont la racine μ est, dans le cas général, un nombre algébrique, et, dans notre cas particulier un rationnel: $\mu = 7/2$.

L'algorithme nous donne également *l'équation réduite* par substitution:

$$4x^3 \hat{g}'' + 2(14x^2 + 2x - 1)\hat{g}' + 5(7x + 2)\hat{g} = 0,$$

qui admet comme solution la série formelle:

$$1 + 5x + (105/4)x^2 + (525/4)x^3 + \dots$$

Les coefficients de la série formelle sont obtenus par l'équation de récurrence:

$$(7 + 2j)(5 + 2j)a_j + 2(7 + 2j)a_{j+1} - (4 + 2j)a_{j+2} = 0 \quad (j = 0, 1, 2, \dots),$$

où $a_0 = 1$ et $a_1 = 5$, associée à l'équation différentielle *génératrice*:

$$4x^3\hat{g}'' + 2(14x^2 + 2x - 1)\hat{g}' + 5(7x + 2)\hat{g} = 0.$$

L'équation de récurrence peut être obtenue par identification des coefficients a_n , après substitution de la série formelle $\sum_n a_n x^n$ à $\hat{g}(x)$ dans l'équation différentielle génératrice.

D'une manière générale, en termes plus techniques, l'opérateur de récurrence (ou de différences-finies) est obtenu à partir de l'équation différentielle génératrice par une transformation formelle de Mellin, qui à un élément de l'algèbre d'Ore des opérateurs différentiels fait correspondre un élément de l'algèbre d'Ore des opérateurs aux différences finies.[10,11,12]

Ces substitutions purement formelles ont conduit à une solution formelle complète:

$$\hat{y}_1(x) = e^{\frac{1}{4x^2} - \frac{1}{2x}} x^{7/2} (1 + 5x + (105/4)x^2 + (525/4)x^3 + \dots)$$

L'autre solution formelle fondamentale calculée suivant les mêmes algorithmes est:

$$\hat{y}_2(x) = e^{\frac{1}{2x}} x^{1/2} (1 - 3x + (3/4)x^2 - (1/4)x^3 - (15/32)x^4 + \dots).$$

Dans le cas général, les solutions formelles peuvent être plus compliquées.

L'algorithme de Frobenius peut en effet introduire des combinaisons de séries avec des puissances de $\text{Log}(x)$ si les racines de l'équation indiciale diffèrent de nombres entiers.

D'autre part, les pentes rationnelles du polygone de Newton peuvent introduire des ramifications du type: $x = t^q$ où $q \in \mathbb{N}^+$.

Nous constatons ici, malgré la simplicité de la place de la singularité (l'origine), l'apparition possible de nombres algébriques dans les racines a et b des équations caractéristiques et les racines μ de l'équation indiciale, puis, par conséquent, dans les coefficients des séries formelles (ici $a = 4$, $b = -2$, $\mu = 7/2$ sont rationnels, donc il en est de même pour les coefficients des séries).

Il faut ajouter que, si la singularité est située ailleurs qu'à l'origine, comme dans l'équation différentielle:

$$(x^2 - 2)y'' + p_1(x)y' + p_2(x) = 0$$

du paragraphe 4.1, il faut dès le départ, effectuer une translation pour ramener la singularité à l'origine; ce qui a pour effet d'introduire dès le départ des nombres algébriques dans les coefficients de l'équation différentielle, puis dans les différents algorithmes de calcul des solutions.

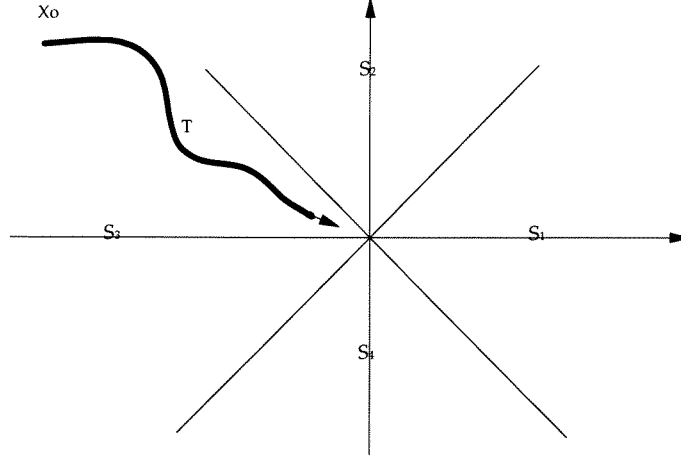
3. Comportement des solutions au voisinage de la singularité

Les résultats formels des 2 sous-paragraphe précédents nous permettent maintenant de procéder à une étude de *comportement*.

Les parties exponentielles $e^{-1/2x}$ et $e^{1/4x^2 - 1/2x}$ nous donnent explicitement le comportement des vraies solutions quand x tend vers 0 dans les différents secteurs

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

du plan complexe $\mathcal{C}$ centrés à l'origine (en sachant, d'après la suite, que les autres parties sont à croissance ou à décroissance modérée).



La solution $y_1(x)$ correspondant à $\hat{y}_1(x)$ tend exponentiellement vers 0 strictement à l'intérieur des secteurs S_2 et S_4 et exponentiellement vers l'infini strictement à l'intérieur des secteurs S_1 et S_3 . La solution $y_2(x)$ correspondant à $\hat{y}_2(x)$ tend exponentiellement vers 0 dans le demi-plan de $\mathbb{R}^-$ et exponentiellement vers l'infini dans le demi-plan de $\mathbb{R}^+$.

Nous pouvons faire l'étude du comportement des solutions du type $\alpha y_1(x) + \beta y_2(x)$ (nous supposons que $y_1(x)$ et $y_2(x)$ sont les vraies fonctions qui seront définies plus loin et qui correspondent aux solutions formelles $\hat{y}_1(x)$ et $\hat{y}_2(x)$) sur une trajectoire dans $\mathcal{C}$, partant d'un point initial X_0 et s'approchant de O .

Par exemple, en partant de X_0 , à l'intérieur de S_3 , en s'approchant de O suivant la trajectoire T de la figure, et en essayant de calculer une solution du type $y_2(x) + \epsilon y_1(x)$ où $\epsilon \in \mathbb{R}^+$ est petit, on s'apercevrait que $y_2(x)$, solution *minimale* devant tendre vers 0, serait rapidement annihilée par la solution *dominante* $y_1(x)$, comme dans l'exemple 4 du par.2. Il est donc inutile d'essayer de s'approcher *numériquement* de la singularité, à cause de ces comportements exponentiels, en plus opposés dans des secteurs adjacents.

En outre, les solutions formelles nous indiquent quelles sont, en vue du calcul des *sommes* des séries divergentes, les *directions singulières*. En effet, les directions de plus forte décroissance, autour de l'origine, de la partie exponentielle $e^{1/4x^2}$ nous donnent le *spectre singulier* ou *directions anti-Stokes*.

Ces directions sont *infranchissables* lors du procédé de sommation de Borel-Laplace (nous verrons pourquoi dans le paragraphe suivant) et donnent la configuration *polarisée*, autour de l'origine, du comportement des solutions. Toute la structure des solutions autour de la singularité est ainsi mise en place et toutes les données exactes utiles sont à notre disposition pour planifier les approximations numériques.

Cette étude formelle aurait pu être menée avec des nombres algébriques plus *sophistiqués* et également avec un certain nombre de paramètres qui auraient pu apparaître dans l'équation différentielle du départ.

4. L'isomorphisme de Borel-Laplace

La théorie de la k -sommabilité de J.P.Ramis [23,25,26,28] nous donne une méthode de calcul des vraies fonctions holomorphes solutions dans des secteurs délimités avec l'aide du spectre singulier. Elle est basée sur les isomorphismes de Borel-Laplace, qui, à une série formelle $\hat{g}(x) = \sum_{n=0}^{\infty} a_n x^n$ solution k -sommable d'une équation différentielle, fait correspondre une *somme* $g(x)$, fonction holomorphe solution, dans un secteur d'ouverture plus grande que π/k , de l'équation différentielle, et dont le développement de Taylor est la série formelle $\hat{g}(x)$.

Cet isomorphisme se décompose en 2 temps:

- une *descente formelle* de l'espace des x dans le *plan de Borel* des ξ par une transformation de Borel formelle;
- une *remontée numérique* du plan de Borel dans l'espace initial par une transformation numérique de Laplace.

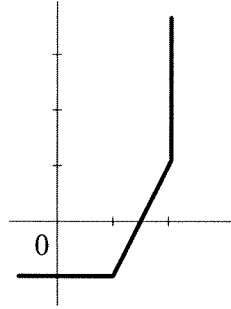
Dans l'exemple que nous avons choisi, nous avons obtenu la série formelle:

$$\hat{g}(x) = 1 + 5x + 105/4x^2 + 525/4x^3 \dots$$

qui est l'une des 2 solutions de l'équation différentielle:

$$4x^3 \hat{g}'' + 2(14x^2 + 2x - 1)\hat{g}' + 5(7x + 2)\hat{g} = 0.$$

Le polygone de Newton de cette équation est composé d'un segment de pente nulle et d'un segment de pente 2:



Nous pouvons en déduire que la série formelle $\hat{g}(x)$, correspondant à la pente nulle, est 2 -sommable.

La deuxième solution fondamentale de cette équation différentielle, calculée par la méthode de Newton, est de la forme $e^{-\frac{1}{4x^2} + \frac{1}{x}} x^\mu \hat{h}(x)$; donc les directions singulières sont $\mathbb{R}^+$ et $\mathbb{R}^-$.

La 2 -sommabilité entraîne que la *somme* $g(x)$ de $\hat{g}(x)$ dans toute direction non singulière d , issue de l'origine est définie par la représentation intégrale de Laplace:

$$g(x) = \frac{1}{x^2} \int_d \varphi(\xi) e^{-\frac{\xi^2}{x^2}} d\xi^2$$

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

où $\varphi(\xi)$ est le prolongement analytique de la somme de la série formelle $\hat{\varphi}(\xi)$ convergente au voisinage de l'origine et où $\hat{\varphi}(\xi)$ est la transformée de Borel formelle d'ordre $k = 2$ de $\hat{g}(x)$.

Chaque terme de la série transformée de Borel formelle est la transformée de Borel d'ordre k du terme correspondant de la série $\hat{g}(x)$:

$$\hat{B}_k(a_n x^n) = \frac{a_n \xi^n}{\Gamma(1 + \frac{n}{k})}.$$

Dans notre cas:

$$\hat{B}_2(a_n x^n) = \frac{a_n \xi^n}{\Gamma(1 + \frac{n}{2})}.$$

Remarque. Pour des raisons de compatibilité avec l'article référencé [34], nous utilisons la définition: $f(x) = \frac{1}{x} \int_d \varphi(\xi) e^{-\frac{\xi}{x}} d\xi$ pour la transformée de Laplace et la définition: $\hat{B}_k(a_n x^n) = \frac{a_n \xi^n}{\Gamma(1+n/k)}$ pour la transformée de Borel. Plus récemment, la théorie de la k -sommation fait usage de la transformée de Laplace:

$$f(x) = \int_d \varphi(\xi) e^{-\frac{\xi}{x}} d\xi$$

et de la transformée de Borel: $\hat{B}_k(a_n x^n) = \frac{a_n \xi^{n-k}}{\Gamma(\frac{n}{k})}$.

Il y a donc un obstacle au calcul exact des fonctions Gamma de nombres fractionnaires (valeurs transcendantes). Pour éviter cet inconvénient, nous scindons $\hat{g}(x)$ en deux sous-séries:

$$\hat{\psi}_1(u) + x\hat{\psi}_2(u) \quad u = x^2.$$

Il est facile de démontrer que chaque sous-série est *1-sommable* dans les directions d_1 correspondant à d dans la transformation $u = x^2$.

Cette fois les transformées de Borel

$$\hat{B}_1(b_n u^n) = \frac{b_n \xi^n}{\Gamma(1+n)}$$

ne comportent que des valeurs entières de la fonction Gamma.

Les coefficients des transformées de Borel formelles: $\hat{\varphi}_1(t)$ et $\hat{\varphi}_2(t)$ respectivement de $\hat{\psi}_1(u)$ et de $\hat{\psi}_2(u)$ seront solutions des équations de récurrence:

$$(5+4j)(7+4j)(9+4j)(11+4j)a_j - 2(7+4j)(9+4j)(11+4j)(j+1)a_{j+1}$$

$$+(6+4j)(8+4j)(j+1)(j+2)a_{j+2} = 0 \quad \text{et}$$

$$(7+4j)(9+4j)(11+4j)(13+4j)a_j - 2(9+4j)(11+4j)(13+4j)(j+1)a_{j+1}$$

$$+(8+4j)(10+4j)(j+1)(j+2)a_{j+2} = 0$$

Si nous calculons formellement l'équation différentielle génératrice de la première équation (toujours par une transformation de Mellin entre 2 algèbres d'Ore), nous trouvons:

$$16(16t^2 - 8t + 1)t^2 \frac{d^4 U}{dt^4} + (3584t^2 - 1248t + 72)t \frac{d^3 U}{dt^3} \\ + (13920t^2 - 2904t + 48) \frac{d^2 U}{dt^2} + (15840t - 1386) \frac{dU}{dt} + 3465U = 0$$

La série formelle $\hat{\varphi}_1(t)$ est solution de cette équation.

L'équation caractéristique: $16t^2 - 8t + 1 = (4t - 1)^2 = 0$ a une racine double en $1/4$, qui est donc la seule singularité à distance finie de l'équation différentielle, donc la seule possible à distance finie pour $\varphi_1(t)$. On peut situer de la même manière la seule singularité possible à distance finie de $\varphi_2(t)$ en $1/4$.

En analysant formellement cette équation différentielle, après avoir effectué une translation du point $+1/4$ en 0 , par les mêmes algorithmes formels de Newton et de Frobenius, par exemple par le code DESIR [15,17,36], nous constatons, grâce au polygone de Newton, que nous nous trouvons en face d'une singularité irrégulière.

Nous savons donc dans quelles directions nous pouvons effectuer le prolongement analytique [13] de $\varphi_1(t)$ afin de pouvoir calculer:

$$\psi_1(u) = \frac{1}{u} \int_{d_1} \varphi_1(t) \exp\left(-\frac{t}{u}\right) dt.$$

La direction d_1 pourra donc être toute direction sauf $\mathbb{R}^+$ qui contient $1/4$.

Comme nous pouvons faire tourner d_1 dans toutes les directions, sauf la direction singulière $\mathbb{R}^+$, et que pour chaque direction, l'intégrale de Laplace est définie pour tout u situé à l'intérieur d'un secteur de rayon fini, d'ouverture π et bissecté par d_1 , on peut montrer que les valeurs de $\psi_1(u)$ se recollent en une fonction holomorphe ainsi définie sur un secteur d'ouverture 3π , en forme d'oeil, et bissecté par $\mathbb{R}^-$, sur la surface de Riemann du logarithme.

Nous pouvons maintenant procéder à l'évaluation numérique des intégrales de Laplace. Dans ce but, il faut d'abord calculer ce prolongement analytique de $\varphi_1(t)$ [13,34,35], qui, d'après la théorie de la *1-sommabilité* est convergente dans le disque centré à l'origine, délimité par cette première singularité en $1/4$. Ce prolongement peut s'effectuer numériquement par différents méthodes.

On voit maintenant que la direction singulière de d_1 correspond à $\mathbb{R}^+$, direction dans laquelle l'intégrand devient singulier en $1/4$, c.à d. $\mathbb{R}^+$ et $\mathbb{R}^-$ pour d après le changement de variable $u = x^2$.

Ces directions singulières sont bien celles qui sont données par le spectre singulier de l'équation

$$4x^3 \hat{g}'' + 2(14x^2 + 2x - 1) \hat{g}' + 5(7x + 2) \hat{g} = 0,$$

c.à.d. par la direction de plus forte décroissance de la partie exponentielle $e^{-\frac{1}{4x^2} + \frac{1}{x}} x^\mu \hat{h}(x)$ de sa 2^e solution.

6 Conclusion

Cet article n'est exhaustif à aucun titre. C'est une première approche de ce qu'apporte le calcul formel au calcul scientifique. À l'aide de quelques exemples, certains

problèmes sont posés et des ouvertures prometteuses d'un domaine en pleine évolution, celui du calcul scientifique, sont mis en évidence. La structure informatique très complexe et les nombreux domaines d'application ne sont de loin pas tous abordés. Evidemment, tous les groupes de recherche, qui sont nombreux en France et dans le monde, ne sont pas mentionnés. D'ailleurs, les références bibliographiques, qui sont citées pour approfondir certains aspects présentés, ne reflètent pas la totalité du travail publié dans ces domaines.

D'autre part, il est indispensable de mentionner une troisième composante qui complète les différentes possibilités de calcul, d'étude et d'analyse de comportement, celle de l'analyse graphique. Beaucoup de résultats, de comportements d'objets mathématiques ne peuvent être compris, interprétés ou même vérifiés par recoupements, superpositions, etc., que lorsqu'ils sont *visualisés*. En effet, comment analyser le comportement de l'image, par une fonction complexe, solution d'une équation différentielle, d'un cercle entourant un point singulier du plan complexe, sinon par des trajectoires où la couleur (et même le temps en film vidéo [17]) permet d'ajouter une quatrième dimension? L'étude et les algorithmes élaborés dans ce domaine, notamment au LMC-Grenoble [17,30,31,32], nous donnent souvent des images non seulement indispensables à l'étude mathématique, mais aussi très esthétiques.

Remerciements. L'auteur tient à remercier Bruno Bernardoff et Alain Sartout pour leur aide et Émile Urlacher pour ses suggestions.

7 Adresses utiles

- <http://www-lmc.imag.fr>

À l'adresse du Laboratoire de Modélisation et Calcul de Grenoble, on peut accéder aux différentes activités du laboratoire et, sous la rubrique *calcul formel*, au serveur CATHODE (projet européen Esprit: Computer Algebra for Handling Ordinary Differential Equations). La finalité de ce projet est d'élaborer et de diffuser des outils théoriques et algorithmiques dans le domaine des ODE (équations différentielles ordinaires), et de donner accès à de nombreuses références bibliographiques, ainsi qu'à différents logiciels concernant ce domaine.

- <http://www-rocq.inria.fr>

À l'INRIA-Rocquencourt, notamment dans le groupe *projet Algorithmes* (<http://www-rocq.inria.fr/algo>), on peut trouver beaucoup de références bibliographiques, de nombreux articles sur les domaines des équations différentielles, des algèbres d'Ore, de la combinatoire et des fonctions génératrices, etc..., ainsi que les algorithmes correspondants.

- <http://www-irma.u-strasbg.fr>

Le réseau donne accès à toutes les activités de l'IRMA. Le texte de cet article est disponible dans la rubrique *Prépublications*.

8 Bibliographie

Ouvrages d'intérêt général

1. ABRAMOWITZ M.,STEGUN I.A.: *Handbook of Mathematical Functions*, Dover Publications.
2. CZAPOR S.R.,GEDDES K.O.,LABAHN G.: *Algorithms for computer algebra*, Éd. Kluwa 1992.
3. DAVENPORT J.H.,SIRET Y.,TOURNIER E.: *Calcul formel*, Éd. Masson 1995.
4. DAVENPORT J.H.,SIRET Y.,TOURNIER E.: *Computer algebra*, Éd. Academic Press 1993.
5. GOMEZ C.,SALVY B.,ZIMMERMANN P.: *Calcul formel: mode d'emploi*, Éd. Masson 1995.
6. MACCALLUM M.A.H.: *Symbolic and algebraic computation*, à paraître dans the Encyclopaedia of computer science and technology, Éd. A.Kent and J.G.Williams, Marcel Dekker, New-York.
7. MIGNOTTE M.: *Mathématiques pour le calcul formel*, Éd. Presses universitaires de France 1989. *Mathematics for computer algebra*, Éd. Springer 1992.
8. MIGNOTTE M.,STEFANESCU D.: *Polynomials, an algorithmic approach*, Éd. Springer 1999.
9. MULLER J-M: *Arithmétique des ordinateurs*, Éd. Masson 1989.

Publications concernant les différentes méthodes

10. BARKATOU M.A.: *Contribution à l'étude des équations différentielles et aux différences dans le champ complexe*, Thèse à l'INP de Grenoble 1989.
11. BARKATOU M.A.,DUVAL A.: *Sur les séries formelles solutions d'équations aux différences polynomiales*, Pub.IRMA Lille vol.31, n°15 1993.
12. BRONSTEIN M.,PETKOVSEK M.: *An introduction to pseudo-linear algebra*, Theoretical Computer Science 157 1996, p.3-33.
13. CHAFFY C.: *The analytic continuation process, From computer algebra to numerical analysis*, in ISSAC'92 Éd. P.S. Wang p.69-79, ACM Press, New-York 1992.
14. DELLA-DORA J., DICRESCENZO C., D.Duval: *About a New Method for Computing in Algebraic Number Fields*, Eurocal'85 vol.2, Springer Lecture Notes in Computer Science 204 éd. G.Goos, J.Hartmanis p. 289-290 1985.
15. DELLA-DORA J., DICRESCENZO C., TOURNIER E.: *An algorithm to obtain formal solutions of a linear homogenous differential equation at an irregular singular point*, in Calmet J. Lecture Notes in Computer Science Springer, 1982.

CALCUL FORMEL OU CALCUL NUMÉRIQUE?

16. DELLA-DORA J.,STOLOVITCH L.: *Normal forms of differential systems*, in Computer Algebra and Differential Equations, Éd. E.Tournier London Math. Society, Lecture Note Series 193, Cambridge University Press 1994.
17. DI CRESCENZO C., JUNG F.: *Un logiciel nommé DÉsir*, film Hi8 de 10mn 1993.
18. DUVAL D.: *Calcul symbolique: automatisation en cours*, La Recherche n°291 oct. 1996 p.64-71.
19. DUVAL D.: *Calcul symbolique: automatisation en cours*, La Recherche Hors-Série 2 (l'univers des nombres) août 1999 p. 98-105.
20. GOMEZ-DIAZ T.: *Quelques applications de l'évaluation dynamique*, Thèse Université de Limoges 1994.
21. GOMEZ-DIAZ T.: *Examples of using dynamic constructible closure*, Mathematics and Computers in Simulation n°42 1996.
22. JUNG F., NAEGELE F., THOMANN J.: *An algorithm of multisummation of formal power series, solutions of linear ODE equations*, Proceedings ISSAC, Lille 1993.
23. LODAY-RICHAUD M.: *Introduction à la Multisommabilité*, Gazette des Mathématiciens SMF n°44 avril 1990.
24. LOUIES S.,BRENIG L.: *Structure and convergence of Poincaré-like normal forms*, Physics Letters A 233 1997 p.184-192.
25. MALGRANGE B.: *Resommation des Séries Divergentes*, Expo.Math.13 1995.
26. MARTINET J.,RAMIS J.P.: *Théorie de Galois différentielle et Resommation* in Tournier E. *Computer Algebra and Differential Equations*, p. 117-214 Academic Press 1988.
27. NAEGELE F.: *Autour de quelques équations fonctionnelles analytiques*, Thèse de l'INPG (1995).
28. RAMIS J.P.: *Les séries k-sommables et leurs applications*, Lecture Notes in Physics **126** Springer 1980.
29. RAMIS J.P.,THOMANN J.: *Some comments about the numerical utilization of factorial series*,
in Della-Dora J.,Demongeot J.,Lacolle B.: *Numerical Methods in the study of critical phenomena*, Springer 1980.
30. RICHARD,F.: *Graphical Analysis of Complex ODE Solutions*, Computer Graphics Forum vol.6 n°4 déc. 1987.
31. RICHARD-JUNG F.: *Représentations graphiques de solutions d'équations différentielles dans le champ complexe*, thèse de l'Université de Strasbourg 1988.
32. RICHARD-JUNG F.: *Le phénomène de Stokes en image*, RT 65 LMC Grenoble 1991.

- 33. SIBUYA Y.: *Linear Differential Equations in the Complex Domain: Problems of Analytic Continuation*, Translations Math. Monographs, vol. 82 AMS Providence, RI 1990.
- 34. THOMANN J.: *Resommation des séries formelles Solutions d'équations différentielles linéaires ordinaires du second ordre dans le champ complexe au voisinage de singularités irrégulières*, Numer.Math. **58**, p.503-535 1990.
- 35. THOMANN J.: *Procédés formels et numériques de sommation de séries solutions d'équations différentielles*, Expo.Math.13 1995.
- 36. TOURNIER E.: *Solutions formelles d'équations différentielles*, Thèse d'Etat de l'Université de Grenoble 1988.