Sur la résolution des systèmes polynomiaux à paramètres

Éric Schost

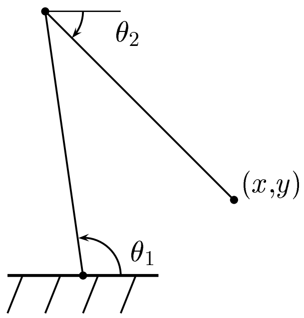
21 janvier 2001

Première partie

Introduction

Préambule

Ce texte traite de la résolution des systèmes polynomiaux à paramètres et plus précisément de la question d'une description des solutions *valable pour une valeur générique des paramètres.*
Illustrons notre propos sur l'exemple d'un bras articulé constitué de deux tiges de longueur 1, inspiré de [87] :



Les coordonnées (x,y) de l'extrémité sont vues comme des paramètres, à partir desquelles on souhaite retrouver les angles θ_1 et θ_2. Pour $i = 1,2$, posons $c_i = \cos\theta_i$ et $s_i = \sin\theta_i$; toutes ces grandeurs sont liées par le système :

$$\left\{\begin{array}{rcl} c_1 + c_2 &=& x, \\ s_1 + s_2 &=& y, \\ c_1^2 + s_1^2 &=& 1, \\ c_2^2 + s_2^2 &=& 1. \end{array}\right.$$

Notre objectif sera par exemple de décrire les solutions sous la forme :

$$s_2^2 - ys_2 + \frac{1}{4}\frac{x^4 + 2x^2y^2 - 4x^2 + y^4}{x^2 + y^2} = 0 \text{ et } \left\{\begin{array}{rcl} c_1 &=& \frac{y}{x}s_2 + \frac{1}{2}\frac{x^2-y^2}{x}, \\ c_2 &=& \frac{y}{x}s_2 + \frac{1}{2}\frac{x^2+y^2}{x}, \\ s_1 &=& -s_2 + y. \end{array}\right.$$

Cette description se spécialise en toute valeur rationnelle de (x,y) qui n'annule pas les dénominateurs ; la variable s_2 devient solution d'une équation du second degré sur $\mathbb{Q}$, et les valeurs des variables (c_1,c_2,s_1) sont données en fonction de ces solutions.

On considérera en priorité des situations à autant d'équations que d'inconnues, où la problématique sera d'obtenir une description des zéros à coefficients dans le corps des fractions rationnelles en les paramètres. En particulier, on se désintéressera des valeurs de dégénérescence des paramètres, et de ce qui peut arriver aux solutions dans ce cas.

⋆ ⋆ ⋆

Mise en contexte

Mon travail se situe dans la filiation de travaux initiés par Giusti et Heintz, et poursuivis dans la série d'articles du groupe TERA [4] qui ont culminé avec la mise au point d'un algorithme dit de *résolution géométrique* d'un système polynomial. Citons entre autres les articles [82, 123, 81, 83, 84], dûs à Krick, Lecerf, Morgenstern, Montaña, Morais, Pardo et Salvy.

Une première spécificité de leurs travaux est la représentation des équations d'entrée par un *calcul d'évaluation*, c'est-à-dire un programme qui les évalue en un point du corps de base, ne comportant ni test ni branchement et n'effectuant que des additions et des multiplications. Une deuxième caractéristique réside dans la nature incrémentale du processus de résolution. Après un changement de variables générique, le cœur de la boucle utilise un *opérateur de Newton-Hensel formel*, suivi d'une phase d'intersection.

En l'état, leurs résultats ne permettaient pas de répondre à mon problème : la simple distinction entre les inconnues et les paramètres interdit la mise en position générique. Néanmoins, la combinaison des outils utilisés dans ces travaux sous-tend toujours mon approche. La condition de généricité mentionnée en préambule est un corollaire de ce choix : on ne peut pas retrouver les fibres pathologiques par ce processus. De même, on devra se limiter à l'étude des solutions *sans multiplicité* : l'utilisation d'un opérateur de Newton nécessite d'inverser une matrice jacobienne.

L'article de Heintz, Krick, Puddu, Sabia et Waissbein [100] aborde également la question des systèmes à paramètres, toujours dans la continuité des travaux du groupe TERA, sous l'hypothèse plus forte d'une variété en position suffisamment générique (position de Noether), ce qui est difficile à demander en pratique. Cette démarche et celle présentée ici ont malgré tout de nombreux points communs, même si la sortie des calculs de [100], représentée par des calculs d'évaluation, offre un résultat de complexité meilleur.

⋆ ⋆ ⋆

De manière générale, les méthodes d'élimination basées sur des calculs de bases de Gröbner [35, 69, 70] ou de triangularisation [137, 111, 205, 153, 11, 57] permettent l'étude des systèmes paramétrés, mais le travail dans un anneau de polynômes regroupant les paramètres et les variables entraîne le traitement des situations dégénérées, ce qui produit des sorties difficilement exploitables. La complexité de ces méthodes est imparfaitement connue.

Les techniques d'*évaluation dynamique* [109, 61, 62, 86, 57] sont bien adaptées à la résolution de systèmes à paramètres, et donnent une description complète de toutes les solutions possibles. La résolution est effectuée pour une valeur des paramètres a priori libre, et les tests à zéro rencontrés au cours du calcul provoquent des scindages, en imposant des conditions sur les paramètres. La complexité, en particulier conditionnée par le nombre de scindages effectués, n'est pas connue. Dans une veine analogue, les "*comprehensive Gröbner bases*", d'après Weispfenning [206] décrivent toutes les possibilités de spécialisation d'un système paramétré donné ; leur calcul est au moins aussi coûteux que celui d'une base de Gröbner. Récemment, Grigoriev et Vorobjov [90] proposent un algorithme de calcul d'une base de "Gröbner paramétrée" reposant sur la résolution paramétrée d'un système linéaire. La complexité est d'ordre $d^{O(mn^2)}$ pour des polynômes d'entrée de degré d, à m paramètres et n variables. Enfin, Rojas [182] propose un algorithme de calcul de la dimension dont la complexité est fonction du volume mixte du système et de sa *hauteur*.

Le parti-pris d'une description générique évite le parcours des branches de calcul, et ramène à une situation de *dimension zéro* sur le corps des fractions rationnelles en les paramètres. À ce titre, tout algorithme de résolution en dimension zéro, *effectué sur le corps des fractions en les paramètres*, résout notre problème. Outre l'algorithme de résolution géométrique, citons notamment le calcul d'une Résolution Univariée Rationnelle [9, 184, 185] (objet similaire à ce que nous allons baptiser résolution de Kronecker), les méthodes de triangularisation déjà citées, les méthodes matricielles [40, 66]... : nous renvoyons au chapitre 1 pour une discussion plus détaillée.

⋆ ⋆ ⋆

Voici maintenant le contenu de ce mémoire. Les résultats de complexité ne sont pas donnés dans leur forme la plus précise, qui nécessite davantage de notations ; on consultera notamment les introductions des parties V, VI et VII pour plus de détails. Des précisions concernant l'implantation et les comparaisons effectuées se trouvent à la fin de cette introduction.

II : Systèmes de dimension zéro

Cette courte partie regroupe les définitions de deux formes de représentation des solutions d'un système de dimension zéro : la *représentation de Kronecker* et la *représentation triangulaire*. Dans la suite, je les utiliserai notamment avec des coefficients dans le corps des fractions rationnelles en les paramètres ; elles portent alors le qualificatif de "*générique*".

Soient $(x_1,\ldots,x_n)$ des indéterminées sur un corps K et $\mathcal{V}$ une variété algébrique de dimension zéro de $\mathbb{A}^n(\overline{K})$. La représentation de Kronecker est la donnée d'un élément primitif t, de son polynôme éliminant $q(t)$, et des paramétrisations $(v_1,\ldots,v_n)$ des points de $\mathcal{V}$ sous la forme :

$$q(t)=0, \quad \left\{ \begin{array}{rcl} \frac{dq(t)}{dt}x_1 & = & v_1(t), \\ & \vdots & \\ \frac{dq(t)}{dt}x_n & = & v_n(t). \end{array} \right.$$

L'exemple de résolution donné en préambule aboutit à une présentation similaire, à la différence du terme de tête associé à chaque paramétrisation. *Le calcul d'une résolution de Kronecker générique sera mon premier objectif.*

La représentation triangulaire prend la forme

$$\left\{ \begin{array}{l} t_1(x_1,x_2,\ldots,x_n)=0, \\ t_2(x_2,\ldots,x_n)=0, \\ \vdots \\ t_n(x_n)=0, \end{array} \right.$$

où pour tout i, le polynôme t_i est unitaire en sa variable principale, réduit par rapport aux polynômes $(t_{i+1},\ldots,t_n)$, et "sans carré", en un sens qui sera précisé. Dans ce mémoire, la représentation triangulaire sera utilisée pour calculer le polynôme minimal générique de x_n.

Cette partie présente brièvement des algorithmes permettant de calculer ces objets, l'algorithme de résolution géométrique déjà mentionné pour le premier cas, les algorithmes `Triangular` et `LexTriangular` de Lazard [138].

III : Rappels

Cette partie résume des résultats techniques souvent déjà connus, qui seront utilisés dans les parties V, VI et VII. La présentation est rapide, ces résultats serviront essentiellement de références.

Les deux premiers chapitres présentent les notions de *degré* et de *hauteur* d'un système de polynômes, puis d'une variété algébrique. Je m'attacherai principalement à fournir de quoi borner ces quantités, à l'aide en particulier de formes géométrique et arithmétique du théorème de Bézout et du Nullstellensatz, tirées essentiellement de [98, 125, 124]. Tous ces outils permettront d'estimer la *taille de la sortie* des algorithmes qui suivront, et d'en contrôler la probabilité d'erreur.

Le troisième chapitre est un chapitre d'algorithmique. Je rappelle la complexité de diverses opérations "élémentaires", à l'aide desquelles sera mesuré le *coût* des algorithmes. Dans ce chapitre, un résultat est à ma connaissance nouveau, il concerne la reconstruction d'une fraction rationnelle multivariée à partir de son développement de Taylor.

⋆ ⋆ ⋆

La suite contient l'essentiel de mes contributions. Je présente pour commencer un opérateur de Newton-Hensel formel valable dans une représentation triangulaire. Les trois parties suivantes présentent des utilisations de cet outil pour calculer d'abord une représentation de Kronecker puis un polynôme minimal génériques, dont les complexités sont *polynomiales en la taille de la sortie* ; à la différence de l'article [100], la sortie est écrite sur la base monomiale. La fin de ce mémoire décrit l'étude d'une situation surdéterminée dépendant de paramètres, tiré de l'article [85] écrit avec M. Giusti, puis des applications des algorithmes décrits ici en théorie des courbes hyperelliptiques, en géométrie réelle et en théorie des invariants, menées en collaboration avec I. Bershenko, P. Gaudry, R. Harley, G. Lecerf et M. Safey el Din.

IV : Opérateur de Newton-Hensel

Je détaille dans cette partie le fonctionnement de l'opérateur de Newton-Hensel formel, sur lequel sont fondés les algorithmes des trois parties suivantes. Il s'agit d'une nouvelle déclinaison d'un principe général d'approximation quadratique ; il tient à la fois de l'opérateur de Newton numérique "usuel" pour la formulation des calculs, et des différentes versions du lemme de Hensel [210] pour son caractère "non archimédien". Cet opérateur est valable dans un cadre plus général que celui des systèmes à paramètres, puisqu'il permet l'approximation d'un ensemble triangulaire modulo les puissances successives d'un idéal de l'anneau de définition :

Opérateur de Newton-Hensel : *Soient $\mathcal{A}$ un anneau commutatif unitaire, $\mathcal{I}$ un idéal de $\mathcal{A}$, $\mathbf{f} = (f_1, \ldots, f_n)$ des polynômes de $\mathcal{A}[x_1, \ldots, x_n]$ et $\mathbf{t}$ un ensemble triangulaire de $\mathcal{A}[x_1, \ldots, x_n]$, tels que :*

- *le système $\mathbf{f}$ se réduit à zéro modulo $\mathbf{t}$;*
- *la matrice jacobienne $\mathbf{Jac}(\mathbf{f})$ est inversible modulo $\mathbf{t}$.*

L'opérateur de Newton-Hensel considéré ici calcule de manière itérative la suite $(\mathbf{t}_i = \mathbf{t} \bmod \mathcal{I}^{2^i})$ à partir de la donnée de $\mathbf{t} \bmod \mathcal{I}$, par une suite de produits matriciels.

Ce résultat englobe le cas particulier d'une résolution univariée, le schéma de calcul étant alors semblable, dans une présentation simplifiée, à celui proposé dans [84]. Le calcul repose sur l'*évaluation* du vecteur $\mathbf{Jac}(\mathbf{f})^{-1}\mathbf{f}$, sa complexité s'énonce donc naturellement en termes du coût d'évaluation du système $\mathbf{f}$. Si celui-ci est donné par un calcul d'évaluation effectuant L opérations, le coût du passage de la précision $\mathcal{I}^{2^{i-1}}$ à la précision $\mathcal{I}^{2^i}$, exprimé en nombre d'opérations de $\mathcal{A}/\mathcal{I}^{2^i}$, est linéaire en L et en le coût de l'arithmétique modulo l'ensemble triangulaire considéré, et polynomial en n.

L'utilisation d'un opérateur de Newton-Hensel nécessite des bornes sur la "taille" des objets que l'on cherche à approximer. Dans l'étude des systèmes à paramètres, cet opérateur servira principalement à l'approximation modulo les puissances de l'idéal $(x_1,\ldots,x_n)$ de l'anneau des séries $k[[x_1,\ldots,x_n]]$; il faudra donc des bornes en degré. Par ailleurs, dans le cas particulier où $k = \mathbb{Q}$, des techniques de calcul modulaire nécessiteront des bornes de hauteur.

On peut obtenir des bornes fines dans le cas particulier d'une représentation de Kronecker : c'est le sujet des deux parties suivantes. Dans le cas d'un système triangulaire général, je n'ai pas obtenu de majorations équivalentes : dans ce contexte, l'opérateur de Newton sera utilisé pour calculer uniquement un polynôme minimal, dans la partie VII.

V : Calcul d'une résolution de Kronecker générique

Soient $(f_1,\ldots,f_n)$ des polynômes dépendant de n inconnues $\mathbf{x} = (x_1,\ldots,x_n)$ et m paramètres $(p_1,\ldots,p_m)$, à coefficients dans un corps k. On note $\mathbf{jac}(\mathbf{f})$ le déterminant jacobien du système $\mathbf{f}$ par rapport aux variables $\mathbf{x}$. L'objet géométrique qui m'intéresse est la réunion des composantes des zéros de $(f_1,\ldots,f_n)$ dans $\mathbb{A}^{m+n}(\overline{k})$ sur lesquelles $\mathbf{jac}(\mathbf{f})$ n'est pas identiquement nul ; elle est notée $\mathcal{U}$.

Cette partie propose un algorithme probabiliste de calcul d'une résolution de Kronecker générique des points de $\mathcal{U}$. Le point de départ est une résolution du système spécialisé en une valeur suffisamment générique des paramètres. À partir de ce point initial, l'opérateur de Newton permet d'approximer la résolution générique par des résolutions à coefficients dans l'anneau de séries formelles $k[[p_1,\ldots,p_m]]$; on arrête le processus de remontée dès qu'il est possible de reconstruire les coefficients recherchés à partir de leur développement en série.

Cette approche permet un calcul itératif, sans branchement, avec un test qui intervient pour décider l'arrêt du calcul et dont la taille des expressions intermédiaires n'explose pas. En revanche, l'algorithme est probabiliste : ce caractère apparaît lors du choix de la valeur de spécialisation et lors du test d'arrêt de la phase de remontée.

Pour contrôler la probabilité d'erreur, il faut des bornes a priori sur la taille des solutions, déduites du degré des objets géométriques sous-jacents. Soient d un majorant des degrés des polynômes d'entrée, $\mathcal{D}$ le degré de la variété $\mathcal{U}$, au sens de Heintz [98] et $\mathcal{D}(t)$ le maximum des degrés en $(p_1,\ldots,p_m)$ des expressions intervenant dans la résolution de Kronecker générique pour l'élément primitif t. Le premier chapitre de cette partie consiste en une étude géométrique qui montre que $\mathcal{D}(t)$ est inférieur à $\mathcal{D}$, lui-même borné par d^n. Sans perte de généralité, on suppose $d > 1$.

Le calcul de la résolution spécialisée est confié à un algorithme de résolution en dimension zéro, tel l'algorithme de résolution géométrique. Sur la base de l'estimation précédente de degré, le chapitre 10 donne une estimation de la complexité et de la probabilité d'erreur de la phase de remontée (théorème 7 page 77). Les calculs dans k sont comptés à coût constant, le modèle est celui de la complexité arithmétique. L'énoncé utilise la notation $\mathcal{T}(t,s)$ qui désigne le nombre de monômes de degré t en s variables, et le nombre $\mathcal{D}_{\mathbf{x}}$ de solutions génériques du système ; enfin, le système $\mathbf{f}$ est supposé donné par un calcul d'évaluation effectuant L opérations — ces notations restent valables dans la suite de l'introduction.

Théorème *Le coût de la phase de remontée est polynomial en* $(\mathcal{D}_{\mathbf{x}},\mathcal{T}(4\mathcal{D}(t),m),L,n,m)$*, pour l'élément primitif* t *retenu. Les choix des valeurs aléatoires qui mènent à un résultat erroné sont enfermées dans une hypersurface de* $\mathbb{A}^{3m-1}(\overline{k})$ *de degré inférieur à* $80nd^{4n}$*.*

L'algorithme nécessite de tirer $3m-1$ valeurs aléatoires dans k, ce qui explique que l'hypersurface à éviter est incluse dans $\mathbb{A}^{3m-1}(\overline{k})$.

La partie s'achève par la description d'une implantation dans le langage Magma version 2.6. Afin de mettre en évidence la complexité arithmétique de l'algorithme, ces premières expériences ont lieu sur des corps finis. L'implantation est comparée aux fonctions de calculs de bases de Gröbner de Magma ; la méthode développée ici permet de résoudre des systèmes sur lesquels échouent ces calculs — cependant, n'oublions pas que notre algorithme est probabiliste.

Pour d'autres méthodes, mentionnons de nouveau [206, 137, 69, 70, 84, 185, 12, 57, 90].

VI : Considérations de complexité binaire

Cette partie traite le cas particulier du corps de base des rationnels, et propose une version de l'algorithme spécialisée pour ce cadre. À nouveau, il faut éviter la croissance des expressions intermédiaires : le calcul est d'abord effectué sur le modèle précédent modulo un nombre premier, une seconde application de l'opérateur de Newton mène ensuite au résultat sur les rationnels.

Ces techniques modulaires augmentent la probabilité d'erreur. Pour l'estimer, il faut contrôler la hauteur des entiers qui peuvent apparaître dans les calculs. Notons $\mathcal{H}(t)$ la hauteur des coefficients de la résolution de Kronecker générique associée à un élément primitif t de hauteur h_t (sous une condition de normalisation naturelle, donnée à la page 120). Dans le chapitre 13, je donne une majoration explicite de $\mathcal{H}(t)$, d'ordre $(n+m)(h+h_t+\log(n+m+1))d^n$, où h est une borne sur la hauteur des coefficients des polynômes $(f_1,\ldots,f_n)$.

L'étude des dégénérescences possibles lors du calcul modulaire, basée en particulier sur le Nullstellensatz arithmétique rappelé dans la partie III, mène à des estimations explicites (théorème 17 page 17) de la taille du nombre premier à utiliser pour le calcul spécialisé et de la complexité du calcul. L'estimation ci-dessous est surestimée par rapport aux résultats du chapitre 15, mais offre une présentation concise :

Théorème *Soit* $H = 10^4hn^2(n+m+10)d^{5n+2}$*, et tirons au hasard un nombre premier* q *compris entre* H *et* $2H$*. Après avoir effectué le calcul modulo* q *d'une résolution spécialisée, le nombre d'opérations* binaires *de la phase de remontée est polynomial en* $(\mathcal{D}_{\mathbf{x}},\mathcal{T}(\mathcal{D}(t),m),\mathcal{H}(t),L,n,m,\log h)$*, pour l'élément primitif* t *retenu, et la probabilité d'erreur du processus est inférieure à* $1/2$*.*

À nouveau, cette partie s'achève sur la description d'une implantation de l'algorithme et son comportement sur une série d'exemples. Les conclusions sont les mêmes que précédemment, l'implantation réussit souvent là où les calculs de bases de Gröbner de Magma échouent.

Outre les méthodes mentionnées dans le paragraphe précédent, citons à ce sujet [123, 79, 84] et [182], dont les algorithmes prennent également en compte une notion de hauteur du système.

VII : Vers une extension aux représentations triangulaires

J'aborde enfin le cadre d'une représentation triangulaire. À nouveau, il faut une majoration sur les degrés des expressions en $(p_1,\ldots,p_m)$ pouvant apparaître dans cette représentation. Le chapitre 18 traite cette question, sans résultat aussi déterminant que dans le cas d'une représentation de Kronecker. J'obtiens une majoration d'ordre $d^{O(n^2)}$, valable pour *chaque* composante de $\mathcal{U}$, en m'inspirant des travaux de Gallo et Mishra [72].

Soit $\mathrm{m}_{x_n} \in \mathbb{Q}(p_1,\ldots,p_m)[t]$ le polynôme minimal générique de la variable x_n. Les résultats de la partie V montrent que le degré $\mathcal{D}(x_n)$ en $(p_1,\ldots,p_m)$ de ses coefficients est borné par d^n. Je propose un algorithme calculant seulement le polynôme m_{x_n} et qui repose sur une représentation triangulaire. Le calcul utilise toujours l'opérateur de Newton formel, appliqué à partir d'un calcul spécialisé. Le théorème 21, page 152, donne le coût et la probabilité d'erreur de la phase de remontée de Newton, en voici une version simplifiée :

Théorème *Le coût arithmétique de la phase de remontée est polynomial en* $(C^n,\mathcal{D}_{\mathbf{x}},\mathcal{T}(4\mathcal{D}(x_n),m), L,m)$*, pour une certaine constante* C *liée au coût de la multiplication modulo un ensemble triangulaire. Les choix des valeurs aléatoires tirées au cours de l'algorithme qui mènent à un résultat erroné sont enfermés dans une hypersurface de* $\mathbb{A}^{3m-1}(\overline{k})$ *de degré inférieur à* $80nd^{4n}$.
J'énonce également sans démonstration un résultat de complexité binaire propre au cas des rationnels, dans la lignée de celui obtenu dans la partie précédente.

Ces techniques permettent en fait le calcul de *tous* les polynômes des ensembles triangulaires génériques. En l'absence d'une borne meilleure que $d^{O(n^2)}$ sur le degré de leurs coefficients en $(p_1,\ldots,p_m)$, je n'ai pas mis en œuvre cette possibilité. Dans le chapitre 18, l'étude d'un cas simplifié ainsi qu'une série d'exemples indiquent que l'introduction d'initiaux bien choisis permet de réduire le degré des expressions en $(p_1,\ldots,p_m)$. Cette question importante sera l'objet de recherches futures.

Pour finir, je présente une implantation de cet algorithme en Magma. Les conclusions sont de même nature que dans les parties précédentes : cette implantation permet de résoudre des systèmes inaccessibles aux bases de Gröbner calculées par Magma.

VIII : Systèmes surdéterminés à paramètres

Cette partie est tirée de l'article [85], écrit avec Marc Giusti. Nous considérons une situation comportant désormais $n+1$ équations à n inconnues et m paramètres. Comme motivation, notons que de nombreuses questions physiques, en robotique [197] ou en vision articielle [128, 152], peuvent s'énoncer dans ce cadre : pour lever une indétermination, on multiplie les mesures.

Sous des hypothèses de généricité raisonnables, nous montrons que l'introduction de l'équation supplémentaire f_{n+1} a une double conséquence. Pour une valeur générique des paramètres, le système de $n+1$ équations est inconsistant ; ensuite, il n'a génériquement qu'une seule solution quand il est consistant. Cela se traduit par l'existence d'un polynôme de $\mathbb{Q}[p_1,\ldots,p_m]$ décrivant localement le *lieu de consistance*, et d'une *expression rationnelle* donnant les variables en fonction des paramètres, valable au-dessus de ce lieu.

De ce point de vue découle un algorithme pour approcher un système consistant à partir d'une situation initiale "bruitée", donc inconsistante : il consiste à faire tendre le jeu initial de paramètres vers l'hypersurface de consistance, par un procédé d'approximation numérique généralisant l'opérateur de Newton archimédien classique, utilisé et étudié dans [8, 201, 196, 56, 55]. Pour mettre en œuvre cet algorithme, le résultat de la partie précédente permet d'obtenir une représentation explicite du lieu de consistance et des paramétrisations ; parallèlement, je propose une approche basée uniquement sur des calculs numériques.

L'exemple d'un robot manipulateur "surdéterminé" vient montrer l'intérêt de notre approche. Partant d'une situation idéale artificiellement perturbée, notre processus de raffinement permet de reconstituer une solution proche de la solution initiale.

IX : Applications : calculs dans les courbes

Cette partie est composée de trois applications des algorithmes des parties V, VI et VII à l'étude des courbes hyperelliptiques, principalement en genre 1 et 2.

Le chapitre 23 décrit une application de l'algorithme de la partie V à la résolution d'un système qui code la division par 2 dans la Jacobienne d'une courbe de genre 2 : les paramètres sont les "coordonnées" du dividende. Ce calcul a été effectué en réponse à une question de P. Gaudry et R. Harley, et a contribué à faire aboutir le calcul du cardinal de cette Jacobienne [74].

Cette question nous a menés, avec P. Gaudry, à étudier les idéaux de torsion de la Jacobienne d'une courbe hyperelliptique de genre arbitraire. Dans le chapitre 24, nous définissons des équations modulaires comme polynômes minimaux d'éléments invariants sous l'action de groupes bien choisis. En genre 1, nous retrouvons une construction due à Charlap, Coley et Robbins [46], et l'extension en genre supérieur est à notre connaissance nouvelle. Les degrés des facteurs de ces polynômes permettent d'obtenir rapidement des informations de cardinalité, dans l'esprit de l'algorithme de Schoof [193] pour les courbes elliptiques.

Enfin, dans le chapitre 25, toujours en collaboration avec P. Gaudry, nous nous intéressons aux courbes de genre 2 dont la Jacobienne est (2,2)-isogène à un produit de courbes elliptiques. Ces courbes sont l'objet de nombreux travaux [108, 135, 129, 146, 130, 105] ; pour notre part, nous avons calculé explicitement le polynôme minimal du j-invariant de ces courbes elliptiques. Ce chapitre reprend l'essentiel de l'article [75], écrit avec P. Gaudry.

Dans ces deux derniers points, les paramètres sont les "modules" décrivant la famille de courbes considérées. Pour une courbe donnée, ces questions peuvent se résoudre par un calcul d'élimination, et notre objectif a été de produire des formules réutilisables indépendamment d'un tel processus. La plus grande part du travail est consacrée à trouver la bonne formulation des systèmes ; ces questions ont motivé la mise au point de l'algorithme de la partie VII.

X : Autres applications

Dans le chapitre 26, en collaboration avec M. Safey el Din, nous nous intéressons à un problème de géométrie réelle : détecter toutes les composantes connexes d'une hypersurface réelle. F. Rouillier, M.-F. Roy, et M. Safey el Din abordent la question dans [187] et proposent un algorithme, qui utilise en particulier une *résolution univariée dépendant d'un paramètre de déformation* ε si l'hypersurface considérée contient une infinité de singularités. L'algorithme de la partie VI permet ce calcul, nous l'avons mis en œuvre sur une série d'hypersurfaces issues du problème d'*interpolation de Birkhoff* [88], ce qui a mené à une première comparaison entre la méthode de [187] et l'algorithme proposé par P. Aubry, F. Rouillier et M. Safey el Din [14]. Cette comparaison est décrite dans [186], rédigé avec F. Rouillier et M. Safey el Din.

Enfin, dans le chapitre 27, avec G. Lecerf, nous répondons à une question d'I. Bershenko. En collaboration avec P.J. Olver [25, 26], elle pose la question d'une classification des orbites de l'action du groupe linéaire sur un espace de formes homogènes. Nous sommes intervenus pour trouver une relation liant des invariants associés à la forme $P^n + Q^n + R^n$. Cette question se ramène à déterminer l'équation d'une hypersurface, nous l'avons traitée comme le calcul d'un polynôme minimal dépendant de paramètres, en application de l'algorithme de la partie VII.

Annexes

Pour conclure, les annexes regroupent :
- un index des principales notations utilisées dans ce mémoire,
- les preuves de résultats de complexité annoncés dans le chapitre 5,
- une évaluation des performances arithmétiques de Magma version 2.6,
- la liste des systèmes étudiés, sous la forme de calculs d'évaluation directement issus des implantations.

⋆ ⋆ ⋆

Implantation

Cette thèse a été écrite parallèlement à l'implantation par G. Lecerf de l'algorithme de résolution géométrique dans le langage Magma [38, 32, 2], dans le prototype `kronecker` [140]. L'implantation des algorithmes des parties V et VI utilise l'opérateur de Newton "univarié" tiré du code de `kronecker` ; une partie des routines que j'ai écrites à cet effet fait maintenant partie du code de Kronecker. Dans la partie VII, l'opérateur de Newton est utilisé dans une représentation triangulaire ; j'ai écrit le code spécifique à cet effet, toujours en Magma.

Parmi les autres implantations disponibles permettant de répondre à mes questions, j'ai retenu les fonctions de calculs de bases de Gröbner fournies par Magma, qui semblent être parmi les versions disponibles les plus performantes ; ce choix permet également d'utiliser un environnement commun pour effectuer les comparaisons, bien que les domaines sur lesquels fonctionnent ces algorithmes ne soient pas le mêmes. Citons également l'implantation FGb [1] de l'algorithme F4 [70] de J.-C. Faugère, mentionnée dans le chapitre 20, et les implantations en Axiom de P. Aubry et M. Moreno Maza [13], S. Dellière [57] dans le domaine des ensembles triangulaires — l'implantation de M. Moreno Maza est disponible dans la version 2.2 d'Axiom.

Tous les calculs relatés dans ce document sont effectués sur une machine Compaq XP/1000 - Alpha EV6 de l'UMS MEDICIS [3].

⋆ ⋆ ⋆

Perspectives

Les méthodes proposées dans ce mémoire ont permis de mener à bien plusieurs applications, souvent inaccessibles aux solutions que propose un logiciel tel que Magma : calculs dans les courbes hyperelliptiques, étude de déformations d'hypersurfaces singulières et calculs de relations entre quantités invariantes sous une action de groupe.

Dans le domaine des courbes hyperelliptiques, l'étude des idéaux de torsion n'est pas close : il reste notamment à traiter le cas $\ell = 5$ en genre 2, ainsi que la possibilité d'une décomposition supplémentaire des extensions considérées. Le domaine de l'automatique semble également porteur d'applications pour les méthodes développées ici. Citons ainsi la question d'établir une relation liant les *paramètres de commande* et les *sorties* d'un système physique — le problème est ici d'éliminer les *variables internes* qui interviennent dans les équations régissant le système.

Ces bons résultats découlent en partie du choix d'une solution “faible”, en l'occurence valable pour “presque toute” valeur des paramètres seulement. Ce parti-pris permet un calcul direct, sans branchement, et n'a pas nui jusqu'ici à l'exploitation des résultats. Un second point-clé est l'utilisation d'algorithmes probabilistes, dont la probabilité d'erreur est contrôlée par des termes polynomiaux en des nombres de Bézout. Rappelons qu'en pratique, aucune erreur n'a été décelée, et que dans bien des cas, les résultats peuvent être aisément vérifiés, voir les chapitres 23, 25, 26 et 27. Enfin, l'utilisation d'une structure de données ad-hoc pour représenter les polynômes d'entrée, les calculs d'évaluation, permet de tenir compte d'une certaine “structure” des systèmes à résoudre, souvent rencontrée en pratique.

Au-delà de ces applications, l'algorithme de la partie VII peut se voir comme le premier pas vers un algorithme de résolution complet reposant sur des techniques d'évaluation dans une représentation triangulaire. Dans cette optique, une question de première importance semble être le contrôle des degrés (voire des hauteurs) des coefficients apparaissant dans un ensemble triangulaire.

Deuxième partie

Systèmes de dimension zéro

Chapitre 1

Systèmes de dimension zéro

Résumé

Ce chapitre est constitué de rappels concernant les systèmes polynomiaux de dimension zéro. Nous introduisons les deux types de représentation des zéros d'un tel système que nous utiliserons, la représentation de Kronecker et la représentation par une famille finie d'ensembles triangulaires. Nous présentons brièvement des algorithmes permettant leur calcul.

Sommaire

Introduction

Le cadre de ce mémoire est la description des zéros génériques d'un système polynomial dépendant de paramètres : le caractère générique de la description permet de se ramener en dimension zéro, en faisant passer les paramètres dans le corps de base. Nous commençons donc par des rappels sur la représentation des zéros d'un système polynomial de dimension zéro.

Soient donc k un corps, $(x_1,\ldots,x_n)$ des indéterminées sur k et $\mathcal{I}$ un idéal radical de $k[x_1,\ldots,x_n]$ de dimension zéro. Nous considérons dans ce chapitre la question d'une description de l'extension

$$k \to B = k[x_1,\ldots,x_n]/\mathcal{I}.$$

Nous supposons que B est un produit de corps séparables sur k. Cette hypothèse n'est pas très restrictive ; nous verrons dans le chapitre 8 des conditions assurant qu'elle est satisfaite.

Nous présentons dans ce chapitre deux représentations des points décrits par l'idéal $\mathcal{I}$ dans $\mathbb{A}^n(\overline{k})$: une représentation univariée qui sera baptisée *représentation de Kronecker*, et une représentation par une famille finie d'*ensembles triangulaires*.

La représentation de Kronecker code les coordonnées des points décrits par $\mathcal{I}$ par la donnée d'un *élément primitif* t de l'extension $k \to B$, du *polynôme minimal* q de t, et de *paramétrisations* $(v_1,\ldots,v_n)$ des coordonnées des solutions. L'existence d'un élément primitif n'est pas assurée pas notre hypothèse de séparabilité, il faut en outre que le cardinal de k soit suffisamment grand. À nouveau, nous donnerons dans le chapitre 8 des conditions sur le cardinal de k assurant l'existence de cet élément. Une telle représentation se présentera sous la forme

$$q(t) = 0, \quad \left\{ \begin{array}{rcl} \frac{dq(t)}{dt} x_1 & = & v_1(t), \\ & \vdots & \\ \frac{dq(t)}{dt} x_n & = & v_n(t), \end{array} \right.$$

ou en abrégé $\mathcal{R} = (t,q,\mathbf{v})$. Le polynôme éliminant q est premier avec sa dérivée, ce qui permet d'obtenir les valeurs des coordonnées x_i au-dessus de ses racines. Nous présenterons en particulier l'algorithme de *résolution géométrique* introduit dans la suite d'articles [82, 81, 83, 84], qui calcule une telle représentation pour l'ensemble des zéros simples d'un système $(f_1,\ldots,f_n)$ de $k[x_1,\ldots,x_n]$, et repose sur des techniques d'*évaluation*.

La représentation par ensembles triangulaires ne procède pas par adjonction d'un élément primitif. Un ensemble triangulaire décrit une suite d'extensions univariées de k, et le quotient B est décrit comme un produit de telles extensions. Nous noterons usuellement un ensemble triangulaire $\mathbf{t} = (t_1,\ldots,t_n)$, ou, de manière plus imagée, sous la forme

$$\left\{ \begin{array}{l} t_1(x_1,x_2,\ldots,x_n) = 0, \\ t_2(x_2,\ldots,x_n) = 0, \\ \vdots \\ t_n(x_n) = 0. \end{array} \right.$$

Les polynômes t_i seront pris *unitaires*, réduits les uns par rapport aux autres et *sans racines multiples*, et nous demanderons que deux ensembles $\mathbf{t}_i$ disjoints remplissent des conditions de non-redondance. Notre présentation est inspirée de celle de Lazard et Aubry, et Lazard et Moreno-Maza [138, 12], nous donnons d'autres références dans le corps du texte.

Dans toute la suite, nous coderons les systèmes de dimension zéro sous l'une ou l'autre de ces formes ; les cadres d'utilisation et d'application motiveront le choix de la représentation. Les algorithmes présentés ultérieurement dans ce mémoire ont d'abord été développés dans le cadre d'une représentation univariée, dans la lignée des travaux du groupe TERA et ce n'est que plus tard que l'extension au cadre triangulaire a été envisagée.

1.1 La représentation de Kronecker

1.1.1 Historique

L'idée de représenter les zéros d'un idéal de dimension zéro à partir d'un élément primitif a une longue histoire. On en trouve trace dès les travaux de Kronecker en 1882 [127], puis chez Macaulay en 1916 [150], van der Waerden [203], Narasimham [166], et certainement bien d'autres. Dans le milieu du calcul formel, ces techniques apparaissent dans les travaux de Chistov et Grigoriev [48], Kobayashi, Moritsugu et Hogan [104], le "Shape Lemma" de Gianni et Mora [78], les travaux de Canny [39], Renegar [179], Lakshman et Lazard [132], Giusti et Heintz [80], Alonso, Becker, Roy et Wörmann [9], Giusti, Hägele, Heintz, Lecerf, Montana, Morais, Morgenstern et Pardo, au sein du groupe TERA, [171, 82, 81, 83, 84], Rouillier [184, 185],...

Les travaux du groupe TERA [80, 171, 82, 81, 83, 84] traitent de variétés de dimension positive, pour lesquelles on procède à une mise en position de Noether, sous les hypothèses que l'idéal $(f_1, \dots, f_n)$ les définissant est donné par une *suite régulière réduite*, et que le corps de base est infini et parfait. Ces variétés de dimension positive sont codées à l'aide d'un élément primitif, et l'ensemble du processus prend le nom d'algorithme de *résolution géométrique*.
L'article [82] propose une première version de cet algorithme. Il repose sur le calcul incrémental des résolutions des variétés $\mathcal{V}(f_1, \dots, f_i)$, retrouvant en cela les idées de Kronecker [127] ; les résultats intermédiaires y sont décrits par des calculs d'évaluation. Dans [81] apparaît l'usage de l'opérateur de Newton formel, qui permet de réduire la taille des calculs d'évaluation intervenant dans le calcul. Il en découle une complexité arithmétique polynomiale, notamment en le degré des variétés intermédiaires $\mathcal{V}(f_1, \dots, f_i)$. Dans la suite, l'article [83] étend ces résultats au cas des polynômes définis sur le corps des rationnels, pour aboutir à une complexité binaire polynomiale, où intervient une notion de *hauteur* du système $(f_1, \dots, f_n)$.
Récemment, l'article [84] propose une version revue et simplifiée de cet algorithme, où les résultats intermédiaires ne sont plus codés par des calculs d'évaluation. Il en découle l'implantation par G. Lecerf, dans le langage Magma [38, 32, 2], d'un prototype appelé `kronecker` [140]. Nous présentons cette version de l'algorithme dans la section 1.1.3.

Citons également les travaux de Alonso, Becker, Roy, Wörmann et Rouillier [9, 184, 185]. Ceux-ci traitent le cas d'idéaux de dimension zéro, sans hypothèse de radicalité et proposent un algorithme pour calculer une *Représentation Univariée Rationnelle* ou RUR, en demandant explicitement que le dénominateur intervenant dans les paramétrisations soit la dérivée du polynôme minimal. Dans le cas d'un idéal radical de dimension zéro, les notions de résolution de Kronecker et de Représentation Univariée Rationnelle coïncident. Le calcul d'une Représentation Univariée Rationnelle repose sur la connaissance d'une table de multiplication du quotient correspondant, aussi cet algorithme prend-il habituellement en entrée une base de Gröbner.

Kobayashi, Fujise et Furukawa [120], Canny, Kaltofen et Lakshman [41], Canny et Emiris [40], Mourrain et Pan [163], Rojas [182], parmi d'autres références, s'attaquent au problème en revenant à la définition donnée par Macaulay [150] du u-résultant, en étendant progressivement ces

considérations au cadre des *polynômes creux.*
La définition du u-résultant donnée par Macaulay coïncide avec celle d'une forme de Chow, qui sera donnée dans le chapitre 4. Nous verrons dans le chapitre 8 qu'il fournit directement une représentation de Kronecker des zéros. Le u-résultant peut s'obtenir, à travers des formulations matricielles, par des calculs d'algèbre linéaire faisant intervenir des divisions ou des calculs de pgcd de polynômes multivariés. Pour une présentation plus complète de ces idées, on pourra consulter l'article de synthèse de Emiris et Mourrain [66].

1.1.2 Élément primitif

La donnée d'une représentation de Kronecker des points décrits par $\mathcal{I}$ passe par l'obtention d'un élément primitif pour l'extension B correspondante.

Définition 1 (Élément primitif) *Notons $\mathcal{D}$ la dimension de B comme k-espace vectoriel. Un élément t de $k[x_1,\dots,x_n]$ est primitif pour l'extension $k \to B$ si les puissances $\{1,t,\dots,t^{\mathcal{D}-1}\}$ forment une base de B.*
Dans la suite, nous considèrerons uniquement des éléments primitifs combinaisons linéaires des variables.

Puisque les puissances $\{1,t,\dots,t^{\mathcal{D}-1}\}$ sont linéairement indépendantes, le polynôme minimal de la multiplication par t a degré maximal $\mathcal{D}$; il coincïde avec son polynôme caractéristique.

Exemple L'exemple suivant est tiré de [156, Ex. 43 p.29], et met en défaut l'hypothèse de séparabilité. Soient p un nombre premier et $k = \mathbb{F}_p(X,Y)$. L'extension

$$k \to k[x,y]/(x^p - X, y^p - Y)$$

n'admet pas d'élément primitif.

L'hypothèse de séparabilité ne suffit pas à assurer l'existence d'un élément primitif pour l'extension $k \to B$. Si k est un corps infini, alors il existe un élément primitif, voir par exemple la preuve de [49, Theorem 2.1.5]. La preuve est donnée sur un corps de caractéristique zéro, mais utilise le caractère infini de k et le fait que le polynôme minimal de tout élément est sans racine multiple, mais si k est un corps fini, nous ne pouvons pas garantir qu'il existe un élément primitif.

Exemple Soit $\mathcal{I} = (x^2 - x, y^2 - y)$ dans $\mathbb{F}_2[x,y]$. On ne peut pas séparer les points de $\mathcal{V}(\mathcal{I})$ par une combinaison $\mathbb{F}_2$-linéaire de x et y. Dans ce cas, on peut néanmoins trouver des éléments primitifs à coefficients dans une extension algébrique de k de degré assez grand. Dans le chapitre 8, nous donnerons une borne sur le cardinal de k en fonction du degré de $\mathcal{V}(\mathcal{I})$ qui assure l'existence d'un élément primitif à coefficients dans k.

La proposition suivante donne une caractérisation des éléments primitifs, et justifie la dénomination alternative d'*élément séparant.*
Proposition 1 *Un élément t est primitif pour $k \to B$ si et seulement si les images par t de points distincts de $\mathcal{V}(\mathcal{I})$ diffèrent.*
Démonstration. Le polynôme minimal d'un élément primitif a degré maximal $\mathcal{D}$. L'hypothèse de séparabilité impose que ce polynôme minimal est séparable, et a donc $\mathcal{D}$ racines distinctes. □

Enfin, dans le cas particulier où k est le corps des rationnels, nous aurons besoin dans la partie VI d'une majoration sur la hauteur d'un élément primitif.
Proposition 2 [212, 185] *Il existe un élément primitif pour l'extension $\mathbb{Q} \to B$ à coefficient entiers de hauteur inférieure à $n \log \mathcal{D}$.*

1.1.3 La représentation de Kronecker

Dans le quotient B, les variables x_i peuvent s'exprimer comme des polynômes en l'élément primitif t. Nous choisissons une forme de représentation particulière, que nous appellerons *représentation de Kronecker*, puisque les travaux de ce mathématicien [127] en font déjà mention.

Définition 2 (Représentation de Kronecker) *Soient t un élément primitif de l'extension $k \to B$ et q son polynôme minimal. Nous appellerons* représentation de Kronecker *l'unique représentation des zéros de $\mathcal{I}$ de la forme*

$$q(t) = 0, \quad \left\{ \begin{array}{rcl} \frac{dq(t)}{dt} x_1 & = & v_1(t), \\ & \vdots & \\ \frac{dq(t)}{dt} x_n & = & v_n(t), \end{array} \right.$$

telle que les polynômes de $\mathbf{v} = (v_1, \ldots, v_n)$ soient de degré strictement inférieur à $\mathcal{D}$.

Cette définition a un sens, puisque notre hypothèse de séparabilité implique que la dérivée $dq(t)/dt$ est inversible modulo q. Dans le milieu du calcul formel, l'introduction de la dérivée du polynôme minimal comme dénominateur semble apparaître dans [9], dans le cadre de l'étude d'un idéal de dimension zéro. C'est sous cette forme que se présente une Représentation Univariée Rationnelle [185], dans le cas d'un idéal radical.

L'intérêt de se ramener à une situation univariée n'est plus à établir, que ce soit pour calculer effectivement dans le quotient B ou, dans le cas où $k = \mathbb{Q}$, obtenir des approximations numériques des racines ou détecter les solutions réelles [120, 179, 16, 185]. Un autre intérêt tient dans la *taille* des paramétrisations, une fois mises sous forme de Kronecker. Nous ne rentrons pas dans les détails ici, c'est le sujet des chapitres 8 et 13.

Une autre représentation. Nous utiliserons également une autre représentation des zéros de $\mathcal{I}$, connue sous le nom de "Shape Lemma", notamment en conjonction avec l'opérateur de Newton formel du chapitre 6. La différence entre cette dernière et la représentation de Kronecker tient dans le choix du terme de tête des paramétrisations.

Définition 3 (Représentation "Shape Lemma") *Soient t un élément primitif de l'extension $k \to B$ et q son polynôme minimal. Nous appellerons* représentation "Shape Lemma" *l'unique représentation univariée des zéros de $\mathcal{I}$ de la forme*

$$q(t) = 0, \quad \left\{ \begin{array}{rcl} x_1 & = & w_1(t), \\ & \vdots & \\ x_n & = & w_n(t), \end{array} \right.$$

telle que les polynômes de $\mathbf{w} = (w_1, \ldots, w_n)$ soient de degré strictement inférieur à $\mathcal{D}$.

La dénomination se réfère au "Shape Lemma" de Gianni et Mora [78] : si la variable x_n est élément primitif, une base de Gröbner pour l'ordre lexicographique $x_1 > \cdots > x_n$ produit cette représentation.

Notations Dans toute la suite, nous suivrons la convention ci-dessous :
- les paramétrisations d'une représentation de Kronecker seront notées $\mathbf{v}$;
- les paramétrisations d'une représentation "Shape Lemma" seront notées $\mathbf{w}$.

Ces résolutions seront notées $\mathcal{R}(t) = (t,q,\mathbf{v})$ ou $\mathcal{R}(t) = (t,q,\mathbf{w})$.

Les polynômes v_i et w_i sont liés par $q'w_i = v_i \mod q$. Autrement dit, passer d'une représentation de Kronecker à une représentation "Shape Lemma" se fait en inversant q' modulo q par un calcul de pgcd étendu. Dans les parties V et VI, ces changements de représentation seront effectués par des procédures notées `ShapeLemmaParametrisation` et `KroneckerParametrisation`. Pour donner leur complexité, nous notons $\mathcal{M}(d)$ le coût de la multiplication de deux polynômes de degré d, en nombre d'opérations du corps de base (voir le chapitre 5 pour une présentation de ce type de notations).

Proposition 3 *En nombre d'opérations de k Le passage à une représentation "Shape Lemma" s'effectue en temps $O\big(n\mathcal{M}(\mathcal{D}) + \mathcal{M}(\mathcal{D})\log\mathcal{D}\big)$, le passage à une représentation de Kronecker en temps $O\big(n\mathcal{M}(\mathcal{D})\big)$.*

Démonstration. Le calcul d'inverse dans un quotient univarié utilise l'algorithme d'Euclide Étendu. Il nécessite donc $O(\mathcal{M}(\mathcal{D})\log\mathcal{D})$ opérations du corps de base (voir [204, chapitre 11]) ; la multiplication avec reste se fait en en $O(\mathcal{M}(\mathcal{D}))$ opérations. □

Algorithme de calcul. Dans la suite de ce mémoire, nous noterons `Resolution` la "boîte noire" ci-dessous, qui renvoie une représentation de Kronecker de l'ensemble des zéros simples du système $(f_1,\ldots,f_n)$, en dehors de l'hypersurface $\mathcal{V}(g)$, s'il existe une forme linéaire à coefficients dans k qui sépare ces points.

FIG. 1.1 – *Résolution univariée des zéros simples*

procedure `Resolution`($\mathbf{f}$,g)

\# **Entree**
\# le système $\mathbf{f}$
\# l'équation g (optionnel)
\# **Sortie**
\# b,$\mathcal{R}$: b booléen. Si b est vrai, $\mathcal{R} = (t,q,\mathbf{v})$
\# est une représentation de Kronecker des zeros simples de $\mathbf{f}$
\# en-dehors de $\mathcal{V}(g)$
end;

Dans le cas particulier fréquent où k est un corps infini parfait, nous baserons ce calcul sur l'algorithme de résolution géométrique déjà mentionné. L'algorithme est incrémental en le nombre d'équations résolues, sa complexité dépend des degrés des *variétés intermédiaires* définies par le jeu d'équations $(f_1,\ldots,f_n)$. Le système $(f_1,\ldots,f_n,g)$ est donné par un calcul d'évaluation effectuant L opérations (voir la définition dans l'annexe B). Enfin, d désigne un majorant sur le degré de ces polynômes.

Théorème 1 [84, Theorem 1] *Notons $\mathcal{V}_i$ l'adhérence de la variété $\mathcal{V}(f_1,\ldots,f_i)$ privée de $\mathcal{V}(g)$ et δ le maximum des degrés des variétés $\mathcal{V}_1,\ldots,\mathcal{V}_{n-1}$. Il existe un algorithme probabiliste qui calcule une résolution géométrique de la variété $\mathcal{V}_n$ en $O(n(nL+n^{\Omega})\mathcal{M}(d\delta)^2)$ opérations de k. Le calcul est probabiliste, les choix des valeurs aléatoires qui mènent à l'échec sont inclus dans un fermé de Zariski.*

Voici un schéma simplifié du calcul :

- On applique un changement de variables aléatoire $\mathbf{x} \mapsto \mathbf{Mx} + \mathbf{L}$.
- À l'étape i, on calcule successivement les résolutions suivantes :

$$\text{Résolution de } \mathcal{V}_i \text{ spécialisée en } x_1, \ldots ,x_{n-i} = 0$$

$$\downarrow \ \textbf{Déspécialisation de } \mathbf{x_{n-i}}$$

$$\text{Résolution de } \mathcal{V}_i \text{ spécialisée en } x_1, \ldots ,x_{n-i-1} = 0$$

$$\downarrow \ \textbf{Intersection avec } f_{i+1}$$

$$\text{Résolution de } \mathcal{V}_i \cap \mathcal{V}(f_{i+1}) \text{ spécialisée en } x_1, \ldots ,x_{n-i-1} = 0$$

$$\downarrow \ \textbf{Nettoyage}$$

$$\text{Résolution de } \mathcal{V}_{i+1} \text{ spécialisée en } x_1, \ldots ,x_{n-i-1} = 0$$

- On revient dans les coordonnées d'origine.

La mise en coordonnées génériques assure que toutes les variétés intermédiaires $\mathcal{V}_i$ sont en position suffisamment générique (en "position de Noether projective") et que toutes les variables sont séparantes.

G. Lecerf a réalisé une première implantation de cet algorithme, nommée `kronecker` [140], dans le langage Magma. Les algorithmes développés dans ce manuscript se situent dans une même veine ; le code de `kronecker` est apparu comme une plateforme naturelle sur laquelle baser leur implantation.

1.2 Ensembles triangulaires

1.2.1 Historique

La notion d'*ensemble caractéristique*, proche de celle d'ensemble triangulaire, est communément attribuée à J.F. Ritt [180, 181], qui l'a introduite dans le contexte de l'algèbre différentielle. Depuis, de nombreux auteurs ont proposé des approches similaires, visant à décrire les zéros d'un système algébrique au moyen d'une famille finie d'*ensembles triangulaires* : Wu Wen-Tsün [209], D. Lazard [138, 137], M. Kalkbrener [111], D. Wang [205], ainsi que l'école de l'évaluation dynamique, notamment D. Duval, T. Gomez-Diàz et S. Dellière [62, 86, 57].

Les définitions diffèrent dans ces approches ; il faut attendre le travail d'Aubry, Lazard et Moreno Maza [12], ainsi que la thèse de S. Dellière [57] pour trouver des études comparatives de différents points de vue. Dans le cadre des systèmes de dimension zéro où l'on a choisi de se placer, différentes notions d'ensembles triangulaires vont coïncider. En dimension positive, ces notions se prolongent, et des différences s'installent : les décompositions "de Kalkbrener" [111, 11] décrivent un ouvert dense de la variété étudiée, alors que les décompositions "de Lazard" [137, 153] décrivent la variété dans son ensemble.

Le paradigme de l'*évaluation dynamique* [61, 86, 57] permet d'effectuer des calculs algébriques, dépendant éventuellement de paramètres, en gérant automatiquement les scindages ayant lieu lors des tests à zéro, en poursuivant le calcul dans différentes branches. Cette méthode trouve son origine dans les questions de traitement des nombres algébriques [61]. Pour des applications à la triangularisation de systèmes algébriques, on consultera [86, 57], et pour une présentation didactique [62].

1.2.2 Définition et propriétés

Définition 4 (Ensemble triangulaire) *Nous appellerons ensemble triangulaire un ensemble* $\mathbf{t} = (t_1, \ldots, t_n)$ *de polynômes de* $k[x_1, \ldots, x_n]$*, où, pour tout* k *de* $\{1, \ldots, n\}$ *:*

- *Le polynôme* t_k *ne dépend que des variables* $(x_k, \ldots, x_n)$*.*
- *Le polynôme* t_k *est unitaire de degré* d_k *en la variable* x_k *et de degré strictement inférieur à* d_j *pour* $j > k$*. Les* d_k *sont strictement positifs.*
- *La dérivée du polynôme* t_k *est inversible modulo les polynômes* $(t_k, \ldots, t_n)$*.*

Le quotient $k[x_1, \ldots, x_n]/(t_1, \ldots, t_n)$ *peut alors être muni d'une structure de produit de corps.*

Cette définition est restrictive ; au sens par exemple de [12], les ensembles définis ci-dessus sont des *ensembles triangulaires réguliers séparables.* Dans tout ce mémoire, nous n'utiliserons que des ensembles de ce type, et nous les appellerons tous ensembles triangulaires, pour ne pas alourdir les notations.

Notations Dans la suite de ce manuscript, nous noterons un ensemble triangulaire par la lette $\mathbf{t}$. Les polynômes que le composent seront notés $(t_1, \ldots, t_n)$ ou plus rarement $(T_1, \ldots, T_n)$. De manière plus imagée, on pourra se les représenter de la sorte :

$$\left\{\begin{array}{l} t_1(x_1, x_2, \ldots, x_n), \\ t_2(x_2, \ldots, x_n), \\ \vdots \\ t_n(x_n). \end{array}\right.$$

Dans le chapitre 18, nous aurons besoin du résultat suivant :

Proposition 4 *Soit* $\mathcal{I}$ *un idéal de* $k[x_1, \ldots, x_n]$*, radical et de dimension zéro. Supposons que l'extension* $k \to k[x_1, \ldots, x_n]/\mathcal{I}$ *est un produit d'extensions séparables. Alors il existe une description de l'ensemble des zéros de* $\mathcal{I}$ *par une famille finie d'ensembles triangulaires* $(\mathbf{t}^{(j)})$ *tels que, pour tout* k *entre* 1 *et* n*, si* $j_1 \neq j_2$*, alors*

- *soit* $(t_k^{(j_1)}, \ldots, t_n^{(j_1)}) = (t_k^{(j_2)}, \ldots, t_n^{(j_2)})$*,*
- *soit* $(t_k^{(j_1)}, \ldots, t_n^{(j_1)}) + (t_k^{(j_2)}, \ldots, t_n^{(j_2)}) = 1$*.*

Démonstration. Il suffit de suivre le déroulement de l'algorithme [138, Procedure 6.1] : deux branches distinctes de l'arbre de calcul sont identiques jusqu'au moment où la factorisation d'un polynôme minimal rend premiers entre eux les deux ensembles de générateurs. □

Soit $(f_1, \ldots, f_n)$ un système de polynômes de $k[x_1, \ldots, x_n]$. Nous appellerons `ResolutionTriangulaire` une procédure qui renvoie une description de l'ensemble des *zéros simples* du système $(f_1, \ldots, f_n)$ par un nombre fini d'ensembles triangulaires, suivant en outre les conditions de la proposition ci-dessus.

Lazard [138] propose deux algorithmes, notés `Triangular` et `LexTriangular`, où les ensembles sont calculés de manière incrémentale, à partir d'une base de Gröbner (respectivement base de Gröbner lexicographique) du système d'entrée ; les calculs reposent sur des factorisations explicites dans des extensions algébriques, ou des scindages, d'après l'algorithme “D5” de Della-Dora, Dicrescenzo et Duval [109].

La donnée d'une représentation de Kronecker permet également ce calcul ; la factorisation du polynôme éliminant univarié $q(t)$ permet notamment d'obtenir une décomposition en une famille d'ensembles engendrant des idéaux premiers.

FIG. 1.2 – *Résolution triangulaire des points simples du système* $\mathbf{f}$

```
procedure ResolutionTriangulaire(f)

# Entree
# le systàme f
# Sortie
# (t^(j)) famille finie d'ensembles triangulaires
# codant les zéros simples de f, satisfaisant les conditions
# de la proposition 4
end;
```

L'article [138] annonce pour l'algorithme `LexTriangular` une complexité polynomiale en fonction du degré de l'idéal considéré, sans cependant en apporter de preuve. Il serait intéressant de reprendre ces considérations, en prenant comme entrée une résolution géométrique, afin de donner une borne sur la complexité totale du processus.

Troisième partie

Rappels

Chapitre 2

Vue d'ensemble

La première partie a vocation de rappels. Ces chapitres sont une compilation de résultats, souvent déjà connus, qui viendront mesurer le déroulement des algorithmes des parties V, VI et VII. Il faut principalement de quoi mesurer les *entrées* et les *sorties* de nos algorithmes, puis les *objets géométriques* qu'elles déterminent, ainsi qu'un ensemble d'*outils élémentaires* pour les traiter.

Le chapitre 3 est formé de propositions de nature technique concernant le *degré* et la *hauteur* de *polynômes multivariés.* Nous insistons sur les diverses mesures liées à la hauteur d'un polynôme à coefficients entiers, et leur comportement lors d'opérations usuelles, puisque ces considérations sont apparues moins connues que leurs équivalents portant sur le degré. Ces résultats seront notamment utilisés dans les chapitres de la partie VI. Nous rappelons également le lemme de Zippel et Schwartz [211, 194], dans sa forme géométrique, ainsi qu'un pendant arithmétique. Ces résultats seront constamment utilisés dans les parties V, VI et VII, pour borner les probabilités d'erreur de calculs probabilistes.

On passe des concepts de degré et de hauteur d'un système de polynômes aux notions de *degré* et de *hauteur* d'une *variété algébrique.* Nous utiliserons la notion du degré géométrique d'une variété affine proposée par Heintz dans [98], et la définition de hauteur d'une variété affine proposée par Krick, Pardo et Sombra dans [124], elle-même inspirée par la définition donnée par Philippon [176] dans le cas projectif. Nous rappelons des formes du théorème de Bézout et du Nullstellensatz effectifs, dans le contexte géométrique classique, puis dans le contexte arithmétique. Tous ces résultats quantitatifs seront utilisés dans les parties V, VI et VII. Seul un résultat qui n'apparaît pas ailleurs dans la littérature, concernant la hauteur d'un cylindre construit sur une variété de dimension zéro, est démontré en détail.

Le troisième chapitre est un chapitre d'*algorithmique*, qui servira de référence pour les chapitres d'algorithmique 10, 15 et 19. Nous rappelons pour commencer les coûts des opérations élémentaires, pour l'arithmétique des entiers, puis des polynômes et des séries. J'ai trouvé peu de références concernant la reconstruction rationnelle d'une fraction rationnelle multivariée à partir de son développement de Taylor, je propose donc un algorithme probabiliste à cet effet. Les annexes C et D présentent respectivement les preuves de résultats annoncés dans ce chapitre, puis un aperçu des performances arithmétiques du logiciel Magma.

Chapitre 3

Degré et hauteur des polynômes

Résumé

Ce chapitre est constitué de rappels concernant les notions de degré et surtout de *hauteur* d'un système de polynômes à coefficients rationnels.

- Dans un premier temps, nous donnons une quantification de la croissance du degré et de la hauteur lors de diverses opérations arithmétiques. Les résultats énoncés sont en grande partie empruntés à l'article de Krick, Pardo et Sombra [124].
- Nous rappelons ensuite la définition des *mesures de Mahler* d'un polynôme à coefficients complexes et leur lien avec la norme de ses coefficients. La présentation est à nouveau tirée de [124].
- Pour finir, nous rappelons le lemme géométrique de Zippel et Schwartz [211, 194], ainsi qu'un analogue arithmétique.

Les outils présentés ici sont utilisés de manière intensive dans les preuves des résultats des chapitres 4, 13 et 14. Ils figurent ici sous une forme assez concise, à fins principalement de références.

Sommaire

Introduction

La notion de degré d'un polynôme multivarié est bien connue, et le comportement du degré lors des opérations usuelles se mesure sans difficulté. Par contre, la notion de hauteur d'un polynôme à coefficients rationnels mérite d'être précisée plus avant. Les définitions suivantes sont empruntées à [124] ; elles se basent sur les valeurs absolues définies sur $\mathbb{Q}$.
Les valeurs absolues définies sur $\mathbb{Q}$ sont notée v ci-dessous ; elles comprennent la valeur absolue v_∞ induite par la norme euclidienne, et les valeurs absolues p-adiques v_p, définies pour p premier. Celles-ci sont définies dans le livre de Lang [134].

Définition 5 (Hauteur) – *Soit Q un ensemble fini de nombres rationnels. Si v est une valeur absolue sur $\mathbb{Q}$, nous appellerons* v-valeur absolue *de Q et noterons $|Q|_v$ le maximum* $\max\{|q|_v, q \in Q\}$. *Nous appellerons ensuite* hauteur v-adique *de Q et noterons $h_v(Q)$ le maximum $\{0, \log |Q|_v\}$.*
- *Sous les mêmes hypothèses, nous appellerons* hauteur *(globale) de Q la somme $\sum h_v(Q)$, où v décrit l'ensemble des valeurs absolues de $\mathbb{Q}$.*
- *Si f est un polynôme à coefficients rationnels, nous appellerons* hauteur *de f et noterons ht f la hauteur globale de l'ensemble de ses coefficients.*

On trouvera d'autres références à ces questions dans les livres de Mignotte [159] et von zur Gathen et Gerhard [204], ainsi que dans les articles [173, 174, 175, 144, 79, 95].

Dans la suite, par souci de simplicité, nous ne manipulerons que des polynômes à coefficients entiers, pour lesquels les hauteurs v_p-adiques, pour p premier, sont nulles ; la hauteur d'un entier est égale au logarithme de sa valeur absolue. Si f est un polynôme à coefficients entiers, représenté par la liste de ses coefficients, son degré et sa hauteur conditionnent la place nécessaire à le stocker, ainsi que le coût binaire des opérations le mettant en jeu.

La hauteur de polynômes $(f_1, \ldots, f_n)$ se comporte d'une manière comparable au degré lors des opérations arithmétiques : la hauteur d'un produit $f_1 f_2$ est proche de la somme des hauteurs de f_1 et f_2, mais il faut apporter des termes correctifs. Dans la partie VI, nous manipulerons des polynômes à coefficients rationnels et entiers, pour lesquels des estimations précises seront nécessaires.

Ce chapitre a donc pour premier objet d'énoncer des bornes sur la croissance du degré et de la hauteur lors des opérations arithmétiques, puis du développement en série, de la factorisation et des opérations d'algèbre linéaire. Les résultats énoncés ici ne sont pas nouveaux, et les démonstrations sont absentes ou succintes. Ensuite, nous rappelons la définition des *mesures de Mahler* d'un polynôme f, et les relions à sa hauteur, lorsque f est à coefficients rationnels ; cette mesure intervient de façon cruciale dans la définition de la hauteur d'une variété donnée dans le chapitre suivant. Pour finir, nous rappelons deux résultats d'évitement fondamentaux : la version géométrique du lemme de Zippel et Schwartz (évitement d'une hypersurface), liée à la notion de degré, ainsi qu'un pendant arithmétique de non-divisibilité.

3.1 Hauteur et opérations usuelles

Dans cette première section, nous exposons des bornes sur le comportement de la hauteur lors des opérations arithmétiques et de la composition, puis sur la hauteur des coefficients d'un développement de Taylor et des facteurs d'un polynôme à coefficients entiers (borne de Mignotte).

3.1.1 Somme, produit et composée

Les deux propositions suivantes sont tirées de [124, Lemma 1.2]. Leur démonstration ne pose pas de difficulté.

Proposition 5 *Soient $(f_1,\ldots,f_n)$ des polynômes de $\mathbb{Z}[p_1,\ldots,p_m]$ de degré borné par d et de hauteur bornée par h. Alors les inégalités suivantes sont vérifiées :*

- $\mathrm{ht}\ \sum f_i \leq h + \log n$,
- $\mathrm{ht}\ \prod f_i \leq nh + (n-1)d\log(m+1)$,
- $\mathrm{ht}\ f_1 f_2 \leq \mathrm{ht}\ f_1 + \mathrm{ht}\ f_2 + \min\{\deg f_1, \deg f_2\}\log(m+1)$.

Proposition 6 *Soient $(f_1,\ldots,f_n)$ des polynômes de $\mathbb{Z}[p_1,\ldots,p_m]$ de degré borné par d_f et de hauteur bornée par h_f, et g un polynôme de $\mathbb{Z}[x_1,\ldots,x_n]$ de degré borné par d_g et de hauteur bornée par h_g. Alors l'inégalité suivante est vérifiée :*

$$\mathrm{ht}\ g(f_1,\ldots,f_n) \leq h_g + d_g\big(h_f + \log(n+1) + d_f\log(m+1)\big).$$

En particulier, si $(x_1,\ldots,x_n)$ est un point de $\mathbb{Z}^n$ de hauteur h, alors $g(x_1,\ldots,x_n)$ a une hauteur inférieure à $h_g + d_g(h + \log(n+1))$.

3.1.2 Développement de Taylor

Nous proposerons dans le chapitre 5 un algorithme de reconstruction d'une fraction rationnelle multivariée à partir de son développement de Taylor à l'origine. Dans le cas d'une fraction à coefficients rationnels, il faudra mesurer la taille des nombres intervenant dans le calcul ; c'est l'objet de la proposition suivante.

Proposition 7 *Soit f/g une fraction rationnelle, où f et g sont des polynômes de $\mathbb{Z}[p_1,\ldots,p_m]$, de degré inférieur à d et de hauteur inférieure à h, avec $g(0) \neq 0$. Soit d' un entier ; le développement de Taylor de f/g à l'origine, à l'ordre $d'+1$ peut s'écrire*

$$\frac{s}{g(0)^{d'+1}},$$

où s est un polynôme de $\mathbb{Z}[p_1,\ldots,p_m]$ de degré d' et de hauteur inférieure à

$$(d'+1)(h + d\log(m+1) + 1).$$

Démonstration. Le développement de f/g à l'origine s'écrit

$$\begin{aligned} & \frac{1}{g(0)}\sum_{i=0}^{d'} f\ (-1)^i \frac{g^i}{g(0)^i} \mod (p_1,\ldots,p_m)^{d'+1} \\ = & \frac{1}{g(0)^{d'+1}}\sum_{i=0}^{d'} f(-1)^i g(0)^{d'-i} g^i \mod (p_1,\ldots,p_m)^{d'+1} \end{aligned}$$

Les termes apparaissant dans la dernière somme sont tous à coefficients entiers. La borne sur le degré est évidente, et la borne en hauteur découle des bornes données ci-dessus sur la somme et le produit, après quelques majorations. □

3.1.3 Hauteur des facteurs

Contrairement au degré, la hauteur peut augmenter lors d'une factorisation. Le résultat suivant, dû à Mignotte [159, Page 172], vient majorer la croissance possible des coefficients.

Proposition 8 *Soient f un polynôme de $\mathbb{Z}[p_1,\ldots,p_m]$ de degré d et de hauteur h, puis g un facteur de f dans $\mathbb{Z}[p_1,\ldots,p_m]$. Alors, si l'on note $\mathcal{T}(d,m)$ le nombre de monômes de degré d en m variables, on a l'inégalité :*

$$\begin{aligned} \text{ht }(g) &\leq h + md\log 2 + \log(\mathcal{T}(d,m)) \\ &\leq h + m(d + \log(d+m)). \end{aligned}$$

3.1.4 Algèbre linéaire

On déduit des résultats précédents une borne de hauteur sur le déterminant d'une matrice de polynômes. Il en découle des bornes pour des cas particuliers : déterminant jacobien et discriminant.

Proposition 9 *Soit $\mathbf{M}$ une matrice $n \times n$ de polynômes de $\mathbb{Z}[p_1,\ldots,p_m]$ de degré borné par d et de hauteur bornée par h. Soit D le déterminant de $\mathbf{M}$. Alors on a les inégalités :*

- $\deg D \leq nd$,
- $\text{ht } D \leq nh + n\log n + (n-1)d\log(m+1)$.

Démonstration. On développe le déterminant comme somme de $n!$ termes. La borne sur le degré est immédiate, la borne sur la hauteur est obtenue en utilisant les inégalités sur la somme et le produit. □

Corollaire 1 *Soient $(f_1,\ldots,f_m)$ des polynômes de $\mathbb{Z}[p_1,\ldots,p_m]$ de degré inférieur à d et de hauteur inférieure à h , et soit* **jac** *le déterminant jacobien du système. Alors on a les inégalités :*

- *Le degré de $\frac{\partial f_i}{\partial p_j}$ est inférieur à d et sa hauteur à $h + \log d$.*
- *Le degré de* **jac** *est inférieur à md et sa hauteur à $m(h+\log d)+m\log m+(2m-1)d\log(m+1)$.*

Démonstration. Les valeurs absolues des coefficients de toutes les dérivées partielles de f_i sont inférieures à d fois celles des coefficients correspondants de f_i, ceci montre la première assertion. Le deuxième point est une conséquence de la proposition 9. □

Corollaire 2 *Soient f un polynôme de $\mathbb{Z}[p_1,\ldots,p_m][t]$ de degré d, dont les coefficients sont de degré inférieur à $\mathcal{D}$ et de hauteur inférieure à h, puis R le résultant de f et f'. Alors on a les inégalités :*

- *le degré de R est inférieur à $2d\mathcal{D}$.*
- *la hauteur de R est inférieure à $2d\big(h + \log(2d) + \mathcal{D}\log(m+1)\big)$.*

Démonstration. C'est une conséquence de la proposition 9 et de la borne sur la hauteur des dérivées donnée plus haut. □

3.2 Hauteur et mesures de Mahler

Nous introduisons ici les mesures de Mahler d'un polynôme multivarié à coefficients rationnels ; ces notions interviendront dans la définition de la hauteur d'une variété définie sur $\mathbb{Q}$ donnée dans le chapitre suivant. Elles ont été introduites par Lehmer [142], puis dans leur version multi-indicée par Philippon [174].

Définition 6 (Mesures de Mahler) *On définit deux notions de mesures de Mahler, la seconde étant une extension de la première.*

- *Soit f un polynôme de $\mathbb{Q}[x_1,\dots,x_n]$, on appelle mesure de Mahler de f et on note $m(f)$ l'intégrale :*
$$\int_0^1 \cdots \int_0^1 \log|f(e^{2i\pi t_1},\dots,e^{2i\pi t_n})|dt_1\dots dt_n.$$
- *Soient $\mathbf{V}$ un ensemble de s groupes de p variables et f un polynôme appartenant à $\mathbb{Q}[\mathbf{V}]$. Soit μ_p la mesure sur la sphère S_p invariante par le groupe unitaire $U(p)$. On note S_p^s le produit $S_p \times \cdots \times S_p$, muni de la mesure produit. La S_p^s-mesure de Mahler de f est définie comme*
$$m(f,S_p^s) = \int_{S_p^s} \log|f|\mu_p^s.$$

On peut relier les mesures de Mahler d'un polynôme f à coefficients entiers à la taille de ses coefficients. Les inégalités suivantes sont issues de [144, 173], et on les trouve également dans [124], sous une forme légèrement plus précise. Elles seront à la base de diverses majorations dans le chapitre suivant.

Proposition 10 *Soit f un polynôme de $\mathbb{Q}[\mathbf{V}]$. Supposons que le degré de f en chacun des s groupes de p variables est borné par d. Alors on a les inégalités :*
$$0 \le m(f) - m(f,S_p^s) \le sd\sum_{i=1}^{p-1}\frac{1}{2i} \quad \textit{et}$$
$$-sd\log(p+1) \le m(f) - \log|f|_\infty \le sd\log(p+1).$$

3.3 Tests à zéro probabilistes

Les algorithmes des parties V, VI et VII sont probabilistes, et se basent sur des *réductions* : spécialisation de variables et réduction d'entiers modulo un nombre premier, leur succès est conditionné par la non-annulation de certaines quantités lors de la réduction. Les résultats ci-dessous viennent mesurer la probabilité qu'une réduction s'annule, d'abord une spécialisation des variables d'un polynôme, ensuite la réduction d'un entier modulo un nombre premier.

3.3.1 Le "lemme" de Zippel-Schwartz

Le résultat suivant a été énoncé, sous des formes très semblables, par Zippel et Schwartz dans respectivement [211] et [194].

Théorème 2 *Soient k un corps, et f un polynôme de $k[p_1,\dots,p_m]$ de degré total majoré par d. Soit Ω un sous-ensemble de k ; alors le nombre total de zéros de f dans Ω^m est borné par*
$$d|\Omega|^{m-1}.$$

Nous appliquerons systématiquement le théorème ci-dessus pour quantifier probabilité qu'un point annule un polynôme de degré borné, via par exemple le corollaire suivant :

Corollaire 3 *Soient k un corps et f un polynôme de $k[p_1,\dots,p_m]$ de degré total majoré par d. Soit Ω un sous-ensemble de k. La probabilité qu'un point tiré de manière uniforme dans Ω^m annule f est inférieure à*

$$\frac{d}{|\Omega|}.$$

3.3.2 Une version arithmétique de Zippel-Schwartz

Dans le même ordre d'idées, nous cherchons à quantifier la probabilité qu'un entier non nul se réduise à zéro modulo un nombre premier p. Les résultats suivants sont tirés du chapitre 18 de [204], on trouve une variante proche dans [94].

Théorème 3 *Soit B un entier positif. Le cardinal de l'ensemble des nombres premiers compris entre B et $2B$ est supérieur à*

$$\frac{B}{2\log B}.$$

Corollaire 4 *Soient B un entier positif, et M et k des entiers positifs tels que $\log M \le \frac{B}{k}$. Si p est un nombre premier compris entre B et $2B$, alors p ne divise pas M avec probabilité au moins*

$$1-\frac{2}{k}.$$

Démonstration. Le nombre de diviseurs premiers de M compris entre B et $2B$ est inférieur à $\frac{\log M}{\log B}$, donc à $\frac{B}{\log Bk}$. D'après l'estimation ci-dessus, il y a au moins $\frac{B}{2\log B}$ nombres premiers entre B et $2B$, donc si l'on en choisit un de manière uniforme, la probabilité qu'il divise M est inférieure à $\frac{2}{k}$. □

Dans la suite, nous négligerons la phase de recherche d'un nombre premier dans un intervalle donné, pour ne pas surcharger l'exposition. Au sens strict, nos algorithmes ne seront pas uniformes en temps, puisqu'ils demandent un tel précalcul : trouver un nombre premier de 200 chiffres décimaux n'est pas immédiat. Pour suivre dans cet ordre d'idées, il est possible de ne pas utiliser des nombres premiers, mais des nombres premiers avec une forte probabilité, voir [204, Theorem 18.10].

Chapitre 4

Degré et hauteur des variétés algébriques

Résumé

Dans ce chapitre, nous présentons rapidement certains outils d'algèbre commutative effective.

- Nous rappelons la notion de *degré* d'une variété algébrique affine, due à Heintz [98], qui est utilisée dans toute la suite, et nous donnons ses principales propriétés. Nous rappelons également la définition d'une *forme de Chow* d'une variété équidimensionnelle, ainsi qu'un lemme de *division* effectif dû à Krick, Sabia et Solernó [125].
- Ensuite, nous rappelons la définition de *hauteur* d'une variété algébrique affine, due à Philippon [176] dans le cas projectif, puis reprise par Krick, Pardo et Sombra dans le cas affine [124]. À nouveau, nous donnons ses principales propriétés, qui s'articulent autour d'un théorème de Bézout et d'un Nullstellensatz effectif arithmétiques.

Nous nous référerons régulièrement aux rappels donnés dans ce chapitre, pour quantifier les résultats de nature géométrique des parties V, VI et VII. Ce chapitre fait office de référence, et peut être passé rapidement en première lecture. Les démonstrations déjà connues sont omises ; seul un résultat concernant la hauteur d'un cylindre est démontré en intégralité, faute d'en avoir trouvé une preuve ailleurs.

Sommaire

4.1 Autour de la notion de degré

4.1.1 Degré d'une variété

Tout au long de ce mémoire, nous utilisons la notion de degré d'une variété algébrique affine proposée par Heintz dans [98] : dans les "bons cas", les degrés des polynômes que nous manipulerons dans les parties V, VI et VII seront des degrés d'objets géométriques.

Définition 7 (Degré de Heintz) *Soit k un corps algébriquement clos.*

- *Soient $\mathcal{U}$ une variété équidimensionnelle de dimension r de $\mathbb{A}^n(k)$ et $\mathcal{H}^{(r)}$ l'ensemble des r-uplets $(\mathcal{H}_1,\dots,\mathcal{H}_r)$ d'hyperplans de $\mathbb{A}^n(k)$ tels que l'intersection $\mathcal{H}_1 \cap \cdots \cap \mathcal{H}_r \cap \mathcal{U}$ est finie. Nous appellerons* degré *de $\mathcal{U}$ et noterons $\deg \mathcal{U}$ le maximum des cardinaux des intersections de $\mathcal{U}$ avec les éléments de $\mathcal{H}^{(r)}$. Il existe un ouvert dense de $\mathcal{H}^{(r)}$ dans lequel le cardinal de l'intersection avec $\mathcal{U}$ est exactement $\deg \mathcal{U}$.*
- *Soit $\mathcal{V}$ une variété de $\mathbb{A}^n(k)$, nous appellerons* degré *de $\mathcal{V}$ et noterons $\deg \mathcal{V}$ la somme des degrés de ses composantes équidimensionnelles.*

Si $\mathcal{V}$ est une variété définie par des polynômes $(f_1,\dots,f_s)$ à coefficient dans un corps k non algébriquement clos, son degré est celui de la variété définie par $(f_1,\dots,f_n)$ dans une clôture algébrique de k.

Pour résumer ses points forts, cette notion de degré est *géométrique* : c'est une notion affine, qui ne tient pas compte des multiplicités. Les principales propriétés dont on fera usage sont rappelées ci-dessous ; elles sont démontrées dans [98].

Proposition 11 *Le degré affine vérifie les propriétés suivantes :*

- *Si $\mathcal{V}$ est de dimension zéro, son degré est égal à son nombre de points.*
- *Si $\mathcal{V}$ une hypersurface, son degré est le degré d'un polynôme sans carré la définissant.*
- *Si $\mathcal{V}$ et $\mathcal{W}$ sont deux variétés algébriques, le degré de l'intersection $\mathcal{V} \cap \mathcal{W}$ est inférieur au produit des degrés $\deg \mathcal{V} \deg \mathcal{W}$. C'est l'"inégalité de Bézout" de Heintz.*
- *Si $\mathcal{V}$ est une variété de $\mathbb{A}^n(k)$ et si φ est une application linéaire $\mathbb{A}^n(k) \to \mathbb{A}^m(k)$, alors le degré de $\overline{\varphi(\mathcal{V})}$ est inférieur ou égal au degré de $\mathcal{V}$.*

En corollaire, si $(f_1,\dots,f_n)$ sont des polynômes à coefficients dans k, avec $\deg f_i = d_i$, alors le degré de la variété définie par les f_i est inférieur au produit $\prod d_i$.

4.1.2 Forme de Chow

La *forme de Chow* d'une variété équidimensionnelle interviendra dans la section suivante, dans la définition de la hauteur d'une variété définie sur $\mathbb{Q}$.

Soient k un corps et $\mathcal{U} \subset \mathbb{A}^n(\overline{k})$ une variété équidimensionnelle de dimension r définie sur k. Pour $0 \le i \le r$, notons $\mathbf{U}_i$ un groupe de $n+1$ variables, et $N = (r+1)(n+1)$ le nombre total de ces nouvelles variables. Ces variables permettent de construire une *forme de Chow* de $\mathcal{U}$, en paramétrant l'espace des hyperplans de $\mathbb{A}^n(\overline{k})$: au $(r+1)$-uplet $\mathbf{u}_i$ on associe l'hyperplan de $\mathbb{A}^n(\overline{k})$ d'équation $u_{i,0} + \sum u_{i,j} x_j = 0$. Pour $\mathbf{x} = (x_1,\dots,x_n)$ dans $\mathbb{A}^n(\overline{k})$, on écrira abusivement $\mathbf{u}_i(\mathbf{x})$ l'évaluation de cette forme affine en $(\mathbf{x})$.

Posons

$$\mathcal{W} := \big\{(\mathbf{u}_0,\dots,\mathbf{u}_r,\mathbf{x}), \text{ t.q. } \mathbf{x} \in \mathcal{U} \text{ et } \mathbf{u}_i(\mathbf{x}) = 0 \text{ pour } i = 0,\dots,r\big\}.$$

La variété $\mathcal{W}$ vit dans $\mathbb{A}^N(\overline{k}) \times \mathbb{A}^n(\overline{k})$ et l'adhérence de se projection sur le premier facteur $\mathbb{A}^N(\overline{k})$ est une hypersurface.

Définition 8 (Forme de Chow) *Nous appellerons forme de Chow de $\mathcal{U}$ et noterons $f_{\mathcal{U}}$ une équation définissant la projection de $\mathcal{W}$ sur $\mathbb{A}^N(\overline{k})$.*

Le degré de $f_{\mathcal{U}}$ en chacun des groupes de variables $\mathbf{U}_i$ est le degré de la variété $\mathcal{U}$; ce polynôme caractérise les ensembles de $(r+1)$ hyperplans qui se recontrent en un point de $\mathcal{U}$ — avec le vocabulaire du chapitre 21, c'est un "lieu de consistance".

4.1.3 Résultat de division

La proposition suivante est due à Krick, Sabia et Solernó, elle est tirée de [125, Lemma 5] et se base sur une formule de trace dans les algèbres Gorenstein, comme les résultats analogues de [79, 124]. Ces résultats sont à la base de Nullstellensätze effectifs, qui produisent des coefficients de Bézout de degrés bornés par des quantités intrinsèques. Dans le chapitre 18, nous utiliserons la forme suivante :

Proposition 12 (Division modulo une suite régulière réduite) *Soit k un corps ; donnons-nous une suite régulière $(f_1, \dots, f_n)$ de $k[x_1, \dots, x_n]$. Soient d une borne sur le degré des polynômes f_i et δ le maximum des degrés des variétés $\mathcal{V}(f_1, \dots, f_i)$, pour $i = 1, \dots, n-1$.*
Pour tout i compris entre 0 et $n-1$, notons B_j le quotient

$$k[x_1, \dots, x_n]/(f_1, \dots, f_{n-i})$$

et supposons que

- *$k[x_1, \dots, x_i] \to B_j$ est une extension entière d'anneau,*
- *le déterminant jacobien de $(f_1, \dots, f_{n-i})$ par rapport aux variables $(x_{i+1}, \dots, x_n)$ n'est pas diviseur de zéro dans B_j.*

Dans ce cas, si h est un polynôme de l'idéal $(f_1, \dots, f_n)$, alors il existe des polynômes $(q_1, \dots, q_n)$ de $k[x_1, \dots, x_n]$ tels que

$$h = q_1 f_1 + \cdots + q_n f_n,$$

avec pour tout i, $\deg q_i f_i \leq 2n^2 d\delta + \delta \max\{\deg h, d\}$.

4.2 Hauteur d'une variété

Soit $\mathcal{U} \subset A^n(\overline{\mathbb{Q}})$ une variété définie sur $\mathbb{Q}$, équidimensionnelle de dimension r et de degré $\mathcal{D}$. On peut définir la *hauteur* de $\mathcal{U}$ à partir d'une de ses formes de Chow. Cette notion de hauteur a un comportement proche du degré d'une variété algébrique ; à elles deux, ces quantités contrôleront la taille des polynômes qui interviendront notamment dans la partie VI.

4.2.1 Définition

La définition suivante est une version simplifiée de [175], où est définie la hauteur d'une variété définie sur un corps de nombres. La définition se base sur la mesure de Mahler et sur les hauteurs p-adiques d'une forme de Chow de $\mathcal{U}$ introduites dans le chapitre précédent. Pour d'autres références sur la notion de hauteur, on pourra consulter [93, 33, 79, 95, 124].

Définition 9 (Hauteur d'une variété) *Soient $\mathcal{U}$ une variété équidimensionnelle de dimension r incluse dans $\mathbb{A}^n(\overline{\mathbb{Q}})$, de degré $\mathcal{D}$, définie sur $\mathbb{Q}$, et $f_{\mathcal{U}}$ une de ses formes de Chow. Nous appellerons* *hauteur de $\mathcal{U}$ et noterons $h(\mathcal{U})$ le nombre*

$$m(f_{\mathcal{U}},S_{n+1}^{r+1}) + \sum_{p \text{ premier}} \log(|f_{\mathcal{U}}|_p) + (r+1)\mathcal{D}\sum_{i=1}^{n}\frac{1}{2i},$$

qui est indépendant du choix de $f_{\mathcal{U}}$. Si $\mathcal{V} \subset A^n(\overline{\mathbb{Q}})$ est une variété définie sur $\mathbb{Q}$, nous définissons sa hauteur comme la somme des hauteurs de ses composantes équidimensionnelles.

4.2.2 Principaux outils

On retrouve dans ce cadre arithmétique des énoncés parallèles à ceux donnés dans la première section, notamment des théorèmes de Bézout et des Nullstellensätze arithmétiques. Voici une série d'outils, qui seront principalement utilisés dans le chapitre 13. Pour commencer, on lie la hauteur d'une hypersurface à celle d'un polynôme sans carré la définissant.

Proposition 13 (Hauteur d'une hypersurface) *Soit $\mathcal{V} \subset \mathbb{A}^n(\overline{\mathbb{Q}})$ une hypersurface définie sur $\mathbb{Q}$, de degré $\mathcal{D}$ et de hauteur $\mathcal{H}$. Alors il existe un polynôme f de $\mathbb{Z}[x_1,\ldots,x_n]$, tel que $\mathcal{V} = \mathcal{V}(f)$, de degré $\mathcal{D}$, et dont la hauteur est bornée par $\mathcal{H} + 2\mathcal{D}$.*

Démonstration. Soit f un polynôme de $\mathbb{Q}[x_1,\ldots,x_n]$ à coefficients entiers, de contenu 1, définissant $\mathcal{V}$; notons f^h le polynôme f homogénéisé par une variable x_0. La hauteur de $\mathcal{V}$ est celle de sa clôture projective, puisqu'elle est définie à partir d'une forme de Chow. Alors d'après [176, Page 347] (voir également [124, Section 1.2.3]), la hauteur de $\mathcal{V}$ et la mesure de Mahler de f^h sont reliés par

$$h(\mathcal{V}) = m(f^h,S_{n+1}) + \deg \mathcal{V}\sum_{i=1}^{n}\sum_{j=1}^{i}\frac{1}{2j}.$$

Les inégalités données dans la proposition 10 du chapitre 3 permettent de majorer $\log|f|_\infty$ à partir de la mesure de Mahler $m(f^h,S_{n+1})$: en prenant pour (s,p) le couple $(1,n+1)$, nous obtenons

$$\begin{aligned}\log|f|_\infty &\leq m(f^h,S_{n+1}) + \mathcal{D}\big(\log(n+2) + \sum_{i=1}^{n}\frac{1}{2i}\big)\\ &\leq h(\mathcal{V}) + \mathcal{D}\big(\log(n+2) + \sum_{i=1}^{n}\frac{1}{2i} - \sum_{i=1}^{n}\sum_{j=1}^{i}\frac{1}{2j}\big).\end{aligned}$$

On montre pour conclure que le terme entre les parenthèses est inférieur à 2. □

Pour borner la hauteur d'une variété à partir d'équations la définissant, on dispose de la borne de Bézout suivante, issue de [124, Corollary 2.10].

Proposition 14 (Borne de Bézout arithmétique) *Soit $(f_1,\ldots,f_s)$ une famille de polynômes de $\mathbb{Z}[x_1,\ldots,x_n]$ définissant une variété $\mathcal{U}$ de dimension r dans $\mathbb{A}^n(\overline{\mathbb{Q}})$. Appelons d_i le degré du polynôme f_i et h une borne sur la hauteur des polynômes f_i. Alors on a l'inégalité :*

$$h(\mathcal{U}) \leq \big(sh + (n+s)\log(n+1)\big)\prod_{i=1}^{s} d_i.$$

Voici une forme de Nullstellensatz arithmétique effectif due à Krick, Pardo et Sombra [124]. C'est un corollaire de [124, Lemma 3.3].

Théorème 4 (Nullstellensatz arithmétique) *Soit* $(f_1,\dots,f_{n+1})$ *une famille de polynômes de* $\mathbb{Z}[x_1,\dots,x_n]$ *sans zéro commun dans* $\mathbb{C}^n$. *On suppose que*

- *les polynômes* $(f_1,\dots,f_n)$ *forment une suite régulière,*
- *pour tout* $i = 1,\dots,n$, *l'idéal* $(f_1,\dots,f_i)$ *est radical,*
- *pour tout* $i = 1,\dots,n$, *la variété* $\mathcal{V}_i = \mathcal{V}(f_1,\dots,f_i)$ *est en position de Noether, avec pour variables libres* $(x_1,\dots,x_{n-i})$,
- *notons* π_i *la projection canonique* $\mathbb{C}^n \to \mathbb{C}^{n-i}$ *; alors il existe un point de* C^{n-i} *admettant* $\deg \mathcal{V}_i$ *antécédents par* π_i *sur* $\mathcal{V}_i$.

Soit d *une borne sur les degrés des polynômes* $(f_1,\dots,f_{n+1})$ *et* h *une borne sur leur hauteur. Alors il existe des polynômes* $(p_1,\dots,p_{n+1})$ *de* $\mathbb{Z}[x_1,\dots,x_n]$ *et un entier* a *tels que*

$$a = p_1 f_1 + \cdots + p_{n+1} f_{n+1},$$

où a *est de hauteur inférieure à*

$$2nd \sum_{j=1}^{n} h(\mathcal{V}_j) + \big((n+1)h + 2n(2n+5)\log(n+1)d\big)\big(1 + \sum_{j=1}^{n} \deg \mathcal{V}_i\big).$$

Enfin, nous utiliserons le résultat suivant sur la hauteur d'un cylindre construit sur une variété de dimension zéro. Ce résultat n'est peut-être pas nouveau, mais n'ayant pas de référence à son sujet, j'en donne une démonstration. Dans [176], on trouve un résultat proche, une borne sur la hauteur du produit de deux variétés *projectives*, dans un plongement de Segre.

Proposition 15 (Hauteur d'un cylindre) *Soit* $\mathcal{V} \subset \mathbb{A}^n(\overline{\mathbb{Q}})$ *une variété définie sur* $\mathbb{Q}$, *de dimension zéro. On appelle* $\widetilde{\mathcal{V}}$ *le cylindre construit sur* $\mathcal{V}$ *dans* $\mathbb{A}^{n+p}(\overline{\mathbb{Q}})$ *:*

$$\widetilde{\mathcal{V}} = \mathcal{V} \times \mathbb{A}^p(\overline{\mathbb{Q}}).$$

Alors la hauteur de $\widetilde{\mathcal{V}}$ *est inférieure à*

$$h(\mathcal{V}) + \deg \mathcal{V}\big(1 + 2\log(n+2) + (p+1)\log(p(n+p)) + (p+2)\log(n+1)\big).$$

Démonstration. L'idée est de déterminer une forme de Chow de $\widetilde{\mathcal{V}}$ à partir d'une forme de Chow de $\mathcal{V}$. La forme de Chow de $\widetilde{\mathcal{V}}$ caractérise la consistance d'un système surdéterminé comprenant plusieurs équations linéaires et portant sur l'ensemble des variables de $\mathbb{A}^{n+p}(\overline{\mathbb{Q}})$. Par combinaisons linéaires, nous allons faire porter ces conditions de consistance uniquement sur les premières variables, ce qui nous ramènera à la forme de Chow de $\mathcal{V}$.

Notons $\mathbf{x} = (x_1,\dots,x_n)$ les coordonnées de $\mathbb{A}^n(\overline{\mathbb{Q}})$ et $\tilde{\mathbf{x}} = (x_1,\dots,x_n,x_{n+1},\dots,x_{n+p})$ les coordonnées dans le nouvel espace ambiant $\mathbb{A}^{n+p}(\overline{\mathbb{Q}})$. Ensuite, pour $i = 0$, notons $\mathbf{u}^0$ le vecteur ligne $(u_0^0,\dots,u_n^0)$, puis pour i entre 0 et p, notons $\tilde{\mathbf{u}}^i$ le vecteur ligne $(u_0^i,\dots,u_n^i,u_{n+1}^i,\dots,u_{n+p}^i)$. La forme de Chow de $\widetilde{\mathcal{V}}$ décrit le lieu de consistance du système surdéterminé noté abusivement

$$\mathcal{E}_1 : \{\tilde{\mathbf{x}} \in \widetilde{\mathcal{V}},\ \text{ et } \tilde{\mathbf{u}}^i(\tilde{\mathbf{x}}) = 0 \ \text{ pour } \ i = 0,\dots,p\}.$$

Le premier jeu d'équations du système $\mathcal{E}_1$ est équivalent à la condition $\mathbf{x} \in \mathcal{V}$ puisque $\widetilde{\mathcal{V}}$ a été construite comme un cylindre. Intéressons-nous maintenant à éliminer les nouvelles variables $(x_{n+1},\dots,x_{n+p})$ des $p+1$ équations linéaires de $\mathcal{E}_1$.

Soit $\mathbf{M}$ la matrice

$$\begin{bmatrix} u_{n+1}^1 & \dots & u_{n+p}^1 \\ \vdots & & \vdots \\ u_{n+1}^p & \dots & u_{n+p}^p \end{bmatrix}.$$

Elle permet d'écrire les p dernières équations linéaires de $\mathcal{E}_1$ sous la forme

$$\mathbf{M} \begin{bmatrix} x_{n+1} \\ \vdots \\ x_{n+p} \end{bmatrix} = \begin{bmatrix} \ell_1 \\ \vdots \\ \ell_p \end{bmatrix},$$

où les ℓ_i sont des formes bilinéaires en les u_i^j et $(x_1, \dots, x_n)$. La forme de Chow de $\widetilde{\mathcal{V}}$ n'est pas un multiple du déterminant de la matrice $\mathbf{M}$, et, dans l'ouvert où cette matrice est inversible, nous pouvons multiplier toutes les équations de $\mathcal{E}_1$ par le déterminant $\det(\mathbf{M})$ sans changer les conditions de consistance.
La première équation linéaire du système $\mathcal{E}_1$, $\tilde{\mathbf{u}}^0(\tilde{\mathbf{x}}) = 0$, est ainsi équivalente à $\det(\mathbf{M})\tilde{\mathbf{u}}^0(\tilde{\mathbf{x}}) = 0$ et le système $\mathcal{E}_1$ est équivalent à

$$\{\mathbf{x} \in \mathcal{V},\ \det(\mathbf{M})\tilde{\mathbf{u}}^0(\tilde{\mathbf{x}}) = 0\},$$

avec la condition supplémentaire

$$\det(\mathbf{M}) \begin{bmatrix} x_{n+1} \\ \vdots \\ x_{n+p} \end{bmatrix} = \operatorname{adj}(\mathbf{M}) \begin{bmatrix} \ell_1 \\ \vdots \\ \ell_p \end{bmatrix}.$$

Cette dernière égalité donne les variables $(x_{n+1}, \dots, x_{n+p})$ en fonction des variables $(x_1, \dots, x_n)$, et permet donc de se débarrasser des variables $(x_{n+1}, \dots, x_{n+p})$, dans la première équation linéaire. L'équation

$$\det(\mathbf{M})\tilde{\mathbf{u}}^0(\tilde{\mathbf{x}}) = 0$$

devient une équation $\mathcal{L}$ linéaire en les variables $(x_1, \dots, x_n)$, et les variables $(x_{n+1}, \dots, x_{n+p})$ ont disparu. Les coefficients de $\mathcal{L}$ sont des polynômes à coefficients entiers, en les variables $\tilde{\mathbf{u}}$. Quelques majorations utilisant la proposition 9 du chapitre 4 montrent que leur degré est inférieur à $p + 1$ et que leur hauteur est égale à 1, puisque toutes les opérations effectuées jusqu'ici mettent en jeu des variables distinctes.

Le système $\mathcal{E}_1$ est consistant si et seulement si les coefficients de la forme linéaire $\mathcal{L}$ annulent une forme de Chow de la variété $\mathcal{V}$. Autrement dit, il suffit d'évaluer une forme de Chow de $\mathcal{V}$ sur les coefficients de $\mathcal{L}$ pour obtenir une forme de Chow de $\widetilde{\mathcal{V}}$.

Choisissons une forme de Chow de $\mathcal{V}$ notée $f_{\mathcal{V}}$, à coefficients entiers et de contenu 1. Alors on peut écrire :

$$h(\mathcal{V}) = m(f_{\mathcal{V}}, S_{n+1}^1) + 0 + \sum_{i=1}^{n} \frac{1}{2i} \deg \mathcal{V}.$$

Par les majorations de la proposition 10, on en déduit que la hauteur de $f_{\mathcal{V}}$ est inférieure à

$$h(\mathcal{V}) + \deg \mathcal{V} \log(n+1).$$

Ensuite, par la majoration de la proposition 6, on en déduit qu'il existe une forme de Chow de $\widetilde{\mathcal{V}}$ à coefficients entiers de hauteur au plus

$$h(\mathcal{V}) + \deg \mathcal{V}\big(1 + 2\log(n+2) + (p+1)\log(p(n+p))\big).$$

Les majorations de la proposition 10 permettent de majorer la mesure de Mahler de cette forme de Chow, puis la hauteur de $\widetilde{\mathcal{V}}$ par

$$h(\mathcal{V}) + \deg \mathcal{V}\big(1 + 2\log(n+2) + (p+1)\log(p(n+p)) + (p+2)\log(n+1)\big).$$

$\square$

Chapitre 5

Algorithmique

Résumé

Ce chapitre est dédié à une présentation de résultats de complexité pour les opérations usuelles d'arithmétique, en grande partie déjà présents dans la littérature, à l'exception d'un algorithme probabiliste de reconstruction rationnelle d'une fraction rationnelle multivariée, à ma connaissance nouveau.

Les résultats de ce chapitre sont à la base des estimations de complexité des algorithmes des parties suivantes ; à nouveau, il faut voir ce chapitre comme une source de références. Les preuves des résultats nouveaux et un aperçu des performances arithmétiques du logiciel Magma sont reportés aux annexes C et D.

Sommaire

Introduction

Les algorithmes proposés dans la suite opèrent dans des anneaux de polynômes et de séries univariés ou multivariés, sur un anneau de base qui sera $\mathbb{Z}$ ou un quotient $\mathbb{Z}/n\mathbb{Z}$. Les éléments de ces anneaux sont mesurés par le nombre de chiffres pour les entiers, par le degré pour les polynômes.

Notations Nous adopterons les *conventions* suivantes.

- Si $\mathcal{P}$ est un algorithme traitant des suites de bits, nous appellerons complexité *binaire* de $\mathcal{P}$ le nombre d'opérations binaires effectuées par $\mathcal{P}$.
- Si $\mathcal{P}$ est un algorithme effectuant des opérations arithmétiques sur un corps k ou un anneau $\mathcal{A}$, la complexité *arithmétique* de $\mathcal{P}$ désignera le nombre d'opérations de k ou $\mathcal{A}$ effectuées lors du calcul.

Des définitions plus précises demanderaient l'introduction, dans le premier cas, d'une machine de Turing ou d'un modèle équivalent (voir [7] pour une présentation), d'un modèle de complexité algébrique dans le second (voir par exemple [37]). Malgré tout, leur signification paraît claire.
Outre les articles cités dans les sections qui suivent, citons comme références les livres d'Aho, Hopcroft et Ullman [7], Zippel [213], Bini et Pan [28], Knuth [119], Bürgisser, Clausen et Shokrollahi [37] et von zur Gathen et Gerhard [204]. Nous adoptons ici une présentation inspirée de [204].

Définition 10 *Nous adopterons les notations de complexité suivantes :*

Opération	*Notation*
$(+,\times)$ *sur les entiers à* d *chiffres*	$\mathcal{M}_{\mathbb{N}}(d)$
$(+,\times)$ *sur les polynômes univariés de degré* d	$\mathcal{M}(d)$
$(+,\times)$ *sur les polynômes en* m *variables de degré* d	$\mathcal{M}_m(m,d)$
$(+,\times)$ *sur les séries en* m *variables à précision* d	$\mathcal{M}_s(m,d)$
$(+,\times)$ *modulo un ensemble triangulaire de degrés* $(d_1,\dots,d_n)$	$\mathcal{M}(d_1,\dots,d_n)$
$(+,\times,\div)$ *sur les matrices* $n\times n$	n^{Ω}
Nombre de monômes de degré inférieur à d *en* m *variables*	$\mathcal{T}(d,m)=\binom{d+m}{m}$

Les coûts sont comptés respectivement en nombre d'opérations binaires dans le premier cas, en nombre d'opérations arithmétiques sinon.

Enfin, la notation $Y \in O_{log}(X)$ *est une abbréviation de* $Y \in X(\log_2(3+X))^{O(1)}$. *Nous dirons alors que* Y *est linéaire en* X *à des facteurs logarithmiques près.*

Ce chapitre donne le coût de diverses opérations arithmétiques sur les entiers, les polynômes et les séries, et les questions d'algèbre linéaire. Nous commencerons par donner le coût de la multiplication, car les coûts des autres opérations s'écrivent généralement en fonction de ce dernier ; les complexités des algorithmes présentés dans ce mémoire s'énonceront en ces termes.

La plupart des résultats donnés ici sont déjà connus, excepté semble-t-il le traitement de la reconstruction rationnelle multivariée, dont le détail est donné dans l'annexe C. L'annexe D fournit des données expérimentales concernant l'arithmétique du langage Magma, qui a été retenu pour toutes les implantations.

5.1 Arithmétique des entiers

Voici les estimations de complexité de l'arithmétique entière qui seront utilisées ultérieurement :

Multiplication. : Calculer le produit de deux entiers de d chiffres.

Reste. : Soient p et q des entiers de respectivement d et d' chiffres, avec $d' \geq d$. Calculer le reste de la division euclidienne de q par p.

Arithmétique dans un quotient. Soit p un entier de d chiffres. Donner le coût des opérations $(+,\times)$ et $\div$ modulo p lorsqu'elles sont possibles.

Reconstruction Rationnelle. Soient $m \geq 0$ un entier de d chiffres, et p et q deux entiers bornés en valeur absolue par $\sqrt{m/2}$. Reconstruire p et q à partir de la réduction de $r = p/q \mod m$.

Le tableau suivant résume des résultats déjà connus sur la complexité de ces différentes tâches. Le coût des algorithmes d'arithmétique entière est mesuré en fonction du nombre de bits nécessaires pour coder les entrées, c'est-à-dire, à une constante près, leur hauteur.

Opération	Coût
Multiplication	$\mathcal{M}_{\mathbb{N}}(d) = O(d \log d \log \log d)$
Reste	$O(\mathcal{M}_{\mathbb{N}}(d'))$
$(+,\times)$ dans un quotient	$O(\mathcal{M}_{\mathbb{N}}(d))$
Division dans un quotient	$O(\mathcal{M}_{\mathbb{N}}(d) \log d)$
Reconstruction rationnelle	$O(\mathcal{M}_{\mathbb{N}}(d) \log d)$

Références et commentaires

- Le calcul rapide du produit est basé sur l'algorithme de Schönhage et Strassen [192], qui utilise la transformée de Fourier rapide. À titre d'exemple, le logiciel Magma v. 2.4 [2] utilise l'algorithme de Karatsuba [117] de complexité $O(d^{\log 3/\log 2}) \subset O(d^{1.59})$ pour les entiers jusqu'à 2^{15} bits.
- L'addition et le produit dans un quotient reposent sur des calculs rapides d'inverse, basés sur l'utilisation d'un opérateur de Newton traitant des nombres flottants ; on verra à ce sujet l'article de Cook [50], ainsi que les livres d'Aho, Hopcroft et Ullman [7] et Knuth [119].
- L'inversion modulaire et la reconstruction rationnelle (quand elles sont possibles) sont basées sur l'algorithme d'Eulide étendu rapide, que l'on trouve dans Lehmer [143]. Dans le cas particulier présenté ici, la reconstruction rationnelle est traitée par Thue [200], puis Dixon [60]. L'article de Kaltofen et Rolletschek [115] répond à la question de manière plus générale, on verra aussi [204, chapter 5.10].

5.2 Arithmétique des polynômes et des séries univariés

Les questions concernant l'arithmétique des polynômes et des séries univariés et les réponses qui y sont apportées sont proches du cas des entiers et souvent plus simples, puisqu'on ne gère pas de retenues. Voici les questions qui nous intéresseront ; suivant les cas, elles sont posées sur un anneau $\mathcal{A}$ ou un corps k.

Multiplication. Calculer le produit de deux polynômes de $\mathcal{A}[t]$ de degrés d.

Reste. : Soient p et q des polynômes de degrés d et $d' \geq d$. Calculer les reste de la division euclidienne de q par p.

Arithmétique dans un quotient. Soit p un polynôme de $\mathcal{A}[t]$ unitaire de degré d. Donner le coût des opérations $(+,\times)$. Si p est à coefficients dans un corps k, donner le coût de l'inversion modulo p, quand elle est possible.

Reconstruction rationnelle. Soient p et q des polynômes de $k[t]$ de degrés inférieurs à $d/2$, avec $q(0) \neq 0$. Soit r le développement de Taylor de p/q à l'origine, à précision $2d$. Calculer p et q à partir de r.

Le tableau suivant résume les réponses connues à ces questions ; les complexités sont données en nombre d'opérations de l'anneau ou du corps de base.

Opération	Coût
Multiplication	$\mathcal{M}(d) = O(d \log d \log \log d)$
Reste	$O(\mathcal{M}(d'))$
$(+,\times)$ dans un quotient	$O(\mathcal{M}(d))$
Division dans un quotient	$O(\mathcal{M}(d) \log d)$
Reconstruction rationnelle	$O(\mathcal{M}(d) \log d)$

Références et commentaires

- La multiplication rapide de polynômes s'effectue par l'algorithme de Schönhage et Strassen. Si l'anneau de base contient des racines 2^kèmes de l'unité pour tout k, le coût descend à $O(d \log d)$ en utilisant transformée de Fourier Rapide (FFT) de Cooley et Tukey [51]. Le logiciel Magma version 2.4 utilise l'algorithme de Karatsuba pour les polynômes de degré inférieur à 128.
- Le calcul de la somme et du produit dans un quotient univarié se base sur le calcul rapide de l'inverse d'un polynôme modulo t^d, en utilisant un processus d'approximation de Newton, et se trouve chez Sieveking [195] et Kung [131].
- La division s'effectue en utilisant l'algorithme d'Euclide étendu rapide.
- La reconstruction rationnelle et le problème analogue du calcul d'approximants de Padé d'une série r de $k[[t]]$ se basent sur l'algorithme d'Euclide étendu appliqué à r et t^{d+1}. Le premier calcul d'approximants de Padé univariés semble dû à Kronecker [126], on les retrouve ensuite

chez Padé [168], puis plus récemment Brendt, Gustavson et Yun [178], et beaucoup d'autres références. Leur construction et des applications sont détaillées dans [28, 204].

5.3 Arithmétique des polynômes et des séries multivariés

Les questions abordées ici sont dans la prolongation des situations précédentes. À nouveau, on les pose sur un anneau $\mathcal{A}$ ou un corps k.

Multiplication de polynômes multivariés. Soient f et g deux polynômes de $\mathcal{A}[p_1, \ldots, p_m]$ de degrés totaux bornés par d. Calculer fg.

Multiplication de séries multivariées. Soient f et g deux séries de $\mathcal{A}[[p_1, \ldots, p_m]]$ de précision d. Calculer fg à précision d.

Arithmétique dans un quotient triangulaire. Soient $\mathbf{t} = (t_1, \ldots, t_n)$ des polynômes de $\mathcal{A}[x_1, \ldots, x_n]$, où pour tout k, t_k ne dépend que des variables $(x_k, \ldots, x_n)$, est unitaire de degré d_k en x_k, et de degré strictement inférieur à d_i pour $i > k$. Donner le coût des opérations $(+, \times)$ et $\div$ modulo $\mathbf{t}$, lorsqu'elles sont possibles

Translation. Soit f un polynôme de $\mathcal{A}[p_1, \ldots, p_m]$ de degré d. Soit $\mathbf{p}_1 = (p_{1,1}, \ldots, p_{1,m})$ un point de $\mathcal{A}^m$, calculer $f(p_1 + p_{1,1}, \ldots, p_m + p_{1,m})$.

Reconstruction rationnelle. Soient p et q des polynômes de $k[p_1, \ldots, p_m]$ de degrés inférieurs à d, avec $q(0) \neq 0$. Soit r le développement de Taylor de p/q à l'origine, à précision $d' \geq 2d$. Calculer p et q à partir de r.

Cette fois, les réponses sont plus difficiles à apporter. Le tableau suivant résume quelques résultats à ce sujet ; à nouveau, le coût est exprimé en nombre d'opérations de l'anneau de base.

Opération	Coût
Multiplication de polynômes	$\mathcal{M}_m(d,m) = O(\mathcal{M}((2d+1)^m))$
Multiplication de séries	$\mathcal{M}_s(d,m) = \mathcal{M}_m(d,m)$
$(+, \times)$ modulo $\mathbf{t}$	$\mathcal{M}(d_1, \ldots, d_n) = O(C^n \mathcal{M}(d_1) \ldots \mathcal{M}(d_n))$ C constante universelle
$\div$ modulo $\mathbf{t}$	$O_{\log}(\mathcal{M}(d_1, \ldots, d_n))$
Translation	$O(m\mathcal{M}_s(d,m))$
Reconstruction rationnelle	$O_{\log}(m\mathcal{M}_s(d',m))$ probabiliste

Les estimations sur le coût de la translation et le coût de la reconstruction rationnelle sont valables pour des séries de degré inférieur au cardinal de $\mathcal{A}$ ou k ; ces conditions seront toujours rencontrées dans la suite.

L'algorithme de reconstruction rationnelle prend en entrée une série r de $k[[p_1,\dots,p_m]]$ et un paramètre $\gamma = (\gamma_2,\dots,\gamma_m)$ de k^{m-1}, et calcule une fraction p/q dont r est le développement, si elle existe. Cette opération est notée `ReconstructionRationnelle`(r,γ), l'algorithme et sa preuve sont donnés dans l'annexe C. Le caractère probabiliste réside dans le choix du paramètre γ ; il est précisé dans le théorème suivant :

Théorème 5 *Soient p et q deux polynômes de $k[p_1,\dots,p_m]$ de degrés au plus d, avec $q(0) \neq 0$, et r le développement de Taylor de p/q à l'origine à précision $d' \geq 2d$. Il existe un polynôme P non nul de $k[\gamma_2,\dots,\gamma_m]$ de degré au plus*

$$(d'+1)d'^2$$

tel que pour tout γ n'annulant pas P, l'algorithme `ReconstructionRationnelle`(r,γ) *calcule p/q en $O_{log}(m\mathcal{M}_s(d',m))$ opérations de k.*
Supposons en outre que $k = \mathbb{Q}$ et que p et q sont à coefficients entiers de hauteur bornée par h. Soit ensuite γ un point de $\mathbb{N}^{m-1}$ à coordonnées bornées par M n'annulant pas P.
Alors il existe un entier non nul e tel que pour tout nombre premier a ne divisant pas $q(0)e$, l'algorithme précédent effectué modulo a est la trace du calcul sur les rationnels. On peut prendre e de hauteur inférieure à

$$d'(d'+1)^2(h + \log M + (d+2)\log(m+1) + 2).$$

Références et commentaires

- Une construction récursive du produit multivarié mène à une complexité d'ordre $c^m d^m$, pour une certaine constante c. On se ramène de préférence à la multiplication de polynômes univariés de degré $(2d+1)^m$, par la *substitution de Kronecker* [28, 204]. La multiplication des séries se fait de la même manière.
- Si on travaille en caractéristique zéro, Canny, Kaltofen et Lakshman [41] proposent un algorithme de multiplication de polynômes basé sur une évaluation, une interpolation et de l'algèbre linéaire rapides, avec une meilleure complexité. Ce résultat peut se transporter en caractéristique p, avec des restriction sur le nombre premier p considéré ; l'article [212] donne davantage de détails. G. Lecerf [141] propose un produit rapide de séries basé sur un principe analogue. Ces méthodes sont décrites dans l'annexe C.
- La complexité des opérations modulo un ensemble triangulaire est décrite en détail par Langemyr [136]. Le résultat donné ici s'obtient par un argument de récurrence, en empilant des quotients univariés. Le résultat de complexité de la division est valable en l'abscence de scindages.
- La preuve de la borne sur la complexité de la translation est donnée dans l'annexe C.
- On trouve peu de références sur la question de reconstruction rationnelle abordées sous l'angle du calcul formel. Les articles [47, 148, 54] et l'article de synthèse de Guillaume et Huard [92] abordent la question de l'approximation rationnelle de fonctions méromorphes, et la problématique est celle de la convergence des approximants ; dans [71], la question est traitée par un calcul de base de Gröbner.

5.4 Algèbre linéaire

Pour conclure, si $\mathcal{A}$ est anneau commutatif unitaire nous noterons n^Ω une borne sur le coût arithmétique des opérations d'addition, de multiplication et d'inversion sur les matrices de taille $n \times n$ à coefficients dans $\mathcal{A}$. Dans toute la suite, *nous pourrons prendre Ω=4.*

Références et commentaires

- Dans le cas où $\mathcal{A}$ est un corps, l'exposant de l'algèbre linéaire est ramené à une valeur de l'ordre de 2.39, suite aux résultats de Coppersmith et Winograd [52]. Dans ce cadre, le premier algorithme de produit et d'inversion de matrices $n \times n$ offrant une complexité inférieure à n^3 est dû à Strassen [198].
- Dans le cas plus général d'un anneau permettant la division par les entiers $(1,\ldots,n)$, l'algorithme de Leverrier [147] permet d'inverser une matrice en $O(n^4)$ opérations. Abdeljaoued [5] donne une description complète de ces méthodes, ainsi que des méthodes bien parallélisables à la Berkowitz [24].

Quatrième partie

Opérateur de Newton-Hensel

Chapitre 6

Opérateur de Newton-Hensel

Résumé

Ce chapitre présente un *opérateur de Newton-Hensel formel* qui calcule des approximations d'un ensemble triangulaire à coefficients dans un anneau $\mathcal{A}$ à partir de la donnée de sa réduction modulo un idéal de $\mathcal{A}$.

Notre version de cet opérateur est dans la lignée de celles présentées dans les articles du groupe TERA [81, 79, 100, 84] et repose toujours sur des techniques d'évaluation. La formulation matricielle est simplifiée, et la portée est étendue du cas d'une résolution univariée à une résolution sous forme triangulaire.

Les conditions d'application de ce processus sont précisément celles des parties V, VI et VII, où nous l'utiliserons comme outil principal.

Sommaire

Introduction

Considérons dans l'anneau $\mathbb{Z}[x_1,x_2]$ les polynômes

$$\begin{aligned}
f_1 &= x_1^2x_2^3 - 1025x_1^2x_2^2 - 1680x_1^2x_2 - 1644x_1^2 - 1276x_1x_2^5 - 1309211x_1x_2^4 - 3489092x_1x_2^3 - \\
&\quad 5978149x_1x_2^2 - 4905444x_1x_2 - 2691228x_1 - 1220x_2^5 - 1251934x_2^4 - 3520476x_2^3 - \\
&\quad 5466450x_2^2 - 4081176x_2 - 1686744, \\
f_2 &= -12x_1^2x_2^3 - 12300x_1^2x_2^2 - 20160x_1^2x_2 - 19727x_1^2 - 15312x_1x_2^5 - 15710532x_1x_2^4 - \\
&\quad 41869104x_1x_2^3 - 71736512x_1x_2^2 - 58864017x_1x_2 - 32293099x_1 - 14640x_2^5 - 15023208x_2^4 - \\
&\quad 42245712x_2^3 - 65596180x_2^2 - 48972678x_2 - 20239901.
\end{aligned}$$

Soit $\mathcal{I}$ l'idéal $(101) \subset \mathbb{Z}$. On se donne une résolution $\mathbf{t}_0$ de ce système modulo $\mathcal{I}$; c'est la base de Gröbner calculée dans $(\mathbb{Z}/101\mathbb{Z})[x_1,x_2]$ pour l'ordre lexicographique $x_1 > x_2$:

$$\begin{aligned}
t_{1,0} &= x_1^2 + 64x_1x_2^2 + 99x_1x_2 + 21x_1 + 8x_2^2 + 20x_2 + 17, \\
t_{2,0} &= x_2^3 + 15x_2^2 + 64x_2 + 28.
\end{aligned}$$

Les résultats de ce chapitre fournissent un moyen de calculer des descriptions des solutions du système (f_1,f_2) modulo les puissances de 101, telle la résolution modulo 101^2 ci-dessous :

$$\begin{aligned}
t_{1,1} &= x_1^2 + (1637 + 1311x_2 + 1276x_2^2)x_1 + 1220x_2^2 + 1434x_2 + 1027, \\
t_{2,1} &= x_2^3 + 1025x_2^2 + 1680x_2 + 1644.
\end{aligned}$$

Ces descriptions permettront *in fine* de décrire les solutions du système sur $\mathbb{Q}$.

Nous travaillerons sous les hypothèses suivantes :

Hypothèse 1 *Soient $\mathcal{A}$ un anneau commutatif unitaire, $\mathcal{I}$ un idéal de $\mathcal{A}$ et $\mathbf{f} = (f_1,\ldots,f_n)$ des polynômes de $\mathcal{A}[x_1,\ldots,x_n]$. Nous considérons un ensemble triangulaire $\mathbf{t} = (t_1,\ldots,t_n)$ dans $\mathcal{A}[x_1,\ldots,x_n]$ tel que :*

- *pour tout k, le polynôme t_k ne dépend que des variables $x_k,\ldots,x_n$;*
- *pour tout k, le polynôme t_k est unitaire en x_k et réduit par rapport aux polynômes $t_{k+1},\ldots,t_n$;*
- *il existe une matrice $\mathbf{A}$ de polynômes de $\mathcal{A}[x_1,\ldots,x_n]$ telle que $\mathbf{f} = \mathbf{A}\mathbf{t}$;*
- *le déterminant jacobien du système $\mathbf{f}$ est inversible dans $(\mathcal{A}/\mathcal{I})[x_1,\ldots,x_n]/(t_1,\ldots,t_n)$.*

La réduction de l'ensemble triangulaire $\mathbf{t}$ modulo $\mathcal{I}$ joue le rôle d'un point initial pour un opérateur de Newton qui calcule les réductions de $\mathbf{t}$ modulo les puissances successives de $\mathcal{I}$, à une précision qui croît quadratiquement.

Nous présentons une écriture matricielle simple de cet opérateur, qui généralise les formules données dans [84]. Pour en donner la complexité, le modèle naturel est une représentation des polynômes d'entrée $(f_1,\ldots,f_n)$ par des calculs d'évaluation (voir la définition dans l'annexe B). La sortie est développée sur la base monomiale.

La section 6.6 montre quelques exemples de calculs, dans les cadres d'application des parties suivantes : remontée des séries formelles et remontée des entiers. Nous décrivons ensuite quelques points liés à la mise en pratique, en particulier diverses possibilités d'accélération des calculs. La dernière section décrit l'implantation dans le langage Magma ; dans le cas particulier d'une représentation univariée, nous utilisons des fonctions issues du code `kronecker` [140] écrit par G. Lecerf.

Les algorithmes des parties V, VI et VII reposent en grande partie sur l'utilisation de l'opérateur présenté ici, appliqué à la remontée de variables libres.

6.1 Historique

L'opérateur de Newton se présente classiquement dans le cadre de l'approximation numérique d'un zéro d'un système d'équations $\mathbf{f}$ disons analytiques. Soit $\mathbf{x} \in \mathbb{R}^n$ et supposons que la jacobienne de $\mathbf{f}$ est inversible en $\mathbf{x}$. Basé sur le développement $\mathbf{f}(\mathbf{x}+\delta\mathbf{x}) = \mathbf{f}(\mathbf{x}) + \mathbf{Jac}(\mathbf{f})\delta\mathbf{x} + o(\|\delta\mathbf{x}\|)$, l'opérateur de Newton s'écrit :

$$N_{\mathbf{f}} : \mathbf{x} \mapsto \mathbf{x} - \mathbf{Jac}(\mathbf{f})^{-1}\mathbf{f}(\mathbf{x}).$$

Si $\mathbf{r}$ est un zéro simple du système $\mathbf{f}$, alors il existe un ouvert $\mathcal{U}$ contenant $\mathbf{r}$ tel que pour tout $\mathbf{x}_0 \in \mathcal{U}$ la suite définie par $\mathbf{x}_{n+1} = N_{\mathbf{f}}(\mathbf{x}_n)$ converge quadratiquement vers $\mathbf{r}$.

L'opérateur de Newton se retrouve dans le cadre d'approximations $\mathcal{I}$-adiques, pour un idéal $\mathcal{I}$ d'un anneau $\mathcal{A}$, tel l'anneau $\mathbb{Z}$ des entiers ; dans ce cas, on parle de méthode de Hensel. La méthode a été introduite dans le milieu du calcul formel par Zassenhaus [210] pour le calcul de pgcd.

Dans le domaine de la résolution des systèmes polynomiaux, l'utilisation de la méthode de Newton-Hensel semble remonter à l'article de Trinks [202], qui propose le calcul d'une base de Gröbner lexicographique de type "Shape Lemma" sur les rationnels à partir d'une base calculée modulo un nombre premier "chanceux".
À la suite de ce travail, Winkler [208] propose un algorithme calculant des approximations p-adiques d'une base de Gröbner, sous l'hypothèse que les générateurs du module des relations se spécialisent modulo p. Les algorithmes proposés dans ces deux articles font appel à des résolutions de systèmes linéaires à coefficients dans un anneau de polynômes. Gianni [76], Kalkbrener [110, 112], Pauer [172], Gräbe [89], Assi [10], puis Gianni, Fortuna et Trager [77] poursuivent l'étude de la question des réductions d'une base de Gröbner, en reliant en particulier cette question à la platitude du morphisme de réduction correspondant. Nauheim [167] aborde la question d'une remontée de Newton effectuée à partir d'une "mauvaise" réduction.

Les travaux du groupe TERA [81, 79, 100] réintroduisent l'usage de l'opérateur de Newton formel, lors de la conception de l'algorithme de résolution géométrique. L'algorithme de résolution traite des variétés de dimension positive, mises en position de Noether projective et représentées par des résolutions univariées, codées par des calculs d'évaluation. L'opérateur de Newton est utilisé pour faire chuter la taille de ces calculs d'évaluation : une résolution représentée par un "gros" calcul d'évaluation est évaluée en un point, l'opérateur de Newton permet de reconstituer une résolution équivalente mais de taille inférieure.
L'article [84] propose une simplification de l'algorithme de résolution géométrique, d'où les calculs d'évaluation intermédiaires ont disparu. L'algorithme de Newton est toujours utilisé pour "déspécialiser" des variables libres d'une résolution univariée mais aussi calculer des approximations p-adiques, pour un nombre premier p ; la formulation est considérablement simplifiée. L'opérateur prend toujours en entrée des polynômes codés par des calculs d'évaluation, sa complexité est donc directement fonction du coût d'évaluation du système.

Notre version de cet opérateur calcule désormais des approximations d'un *ensemble triangulaire*, en prenant toujours comme entrée un système donné par un calcul d'évaluation. La présentation des calculs n'utilise plus que des produits matriciels ; dans le cas particulier d'une résolution sous forme résolution univariée, les calculs effectués sont les mêmes que dans [84] mais présentés sous une forme plus synthétique.

6.2 Préliminaire : approximations d'inverse

Nous commençons par donner un résultat sur les calculs d'approximations successives de l'inverse d'un élément dans une algèbre, qui servira dans la section suivante. L'approximation d'inverse est à la base de l'algorithme de calcul du reste euclidien mentionné dans le chapitre 5 ; elle est décrite dans [195, 131, 169, 213, 28]

Lemme 1 *Soient $\mathcal{A}$ un anneau commutatif, $\mathcal{I}$ un idéal de $\mathcal{A}$, $\mathcal{M}$ une $\mathcal{A}$-algèbre et m un élément de $\mathcal{M}$. Si m est inversible d'inverse n_0 dans $\mathcal{M}/\mathcal{I}\mathcal{M}$, alors m est inversible d'inverse $n_1 = 2n_0 - n_0 m n_0$ dans $\mathcal{M}/\mathcal{I}^2\mathcal{M}$.*

Démonstration. L'hypothèse nous dit qu'il existe i dans $\mathcal{I}\mathcal{M}$ tel que $mn_0 = 1 + i$. L'élément n_1 est l'image de $2n_0 - n_0 m n_0$ calculé dans $\mathcal{M}$, pris modulo $\mathcal{I}^2$, et est indépendant du choix du représentant de n_0 dans $\mathcal{M}$. Alors $mn_1 = 2(1+i) - (1+i)^2 = 1 \mod \mathcal{I}^2$. Ce calcul est une version de l'opérateur de Newton appliqué à la fonction $y \mapsto y - 1/x$. □

Corollaire 1 *Sous les mêmes hypothèses, m est inversible modulo toutes les puissances $\mathcal{I}^{2^i}$. Les inverses successifs sont donnés par l'itération du schéma ci-dessus.*

Nous appliquerons notamment ce processus dans le cas où $\mathcal{M}$ est l'algèbre de matrices $\mathcal{M}_n(\mathcal{A})$, où l'on désire les inverses modulo les puissances successives $\mathcal{I}^{2^i}$ d'une matrice donnée. Le calcul d'inverse dans $\mathcal{M}_n(\mathcal{A}/\mathcal{I}^{2^i})$ se fait en $O(n^\Omega)$ opérations de $\mathcal{I}^{2^i}$, et dans le cas général, Ω est pris égal à 4 (voir le chapitre 5). Le procédé décrit ici offre une meilleure complexité, puisque la seule inversion est effectuée dans $\mathcal{M}_n(\mathcal{A}/\mathcal{I})$, les mises à jour se faisant par produits matriciels, en $O(n^3)$ opérations.

6.3 Approximation d'un système triangulaire

Notations Nous travaillons sous les hypothèses données dans l'introduction, et nous adoptons les notations suivantes. Pour tout $i \geq 0$, nous appellerons :

- $\mathcal{I}_i$ l'idéal $\mathcal{I}^{2^i}$.
- $\mathcal{A}_i$ le quotient $\mathcal{A}/\mathcal{I}_i$ (avec en particulier $\mathcal{A}_0 = \mathcal{A}/\mathcal{I}$).
- $\mathbf{t}_i$ le vecteur $\mathbf{t}$ modulo $\mathcal{I}_i$ et $\mathbf{Jac(t)}_i$ sa jacobienne.
- $\mathbf{f}_i$ le vecteur $\mathbf{f}$ modulo $\mathcal{I}_i$ et $\mathbf{Jac(f)}_i$ sa jacobienne.
- $\mathbf{A}_i$ la matrice $\mathbf{A}$ modulo $\mathcal{I}_i$.

Lemme 2 *Pour tout $i \geq 0$, la jacobienne de $\mathbf{f}$ est inversible dans $\mathcal{A}_i[x_1, \ldots, x_n]/(t_{1,i}, \ldots, t_{n,i})$. Nous noterons $\mathbf{Jac(f)}_i^{-1}$ son inverse dans $\mathcal{A}_i[x_1, \ldots, x_n]/(t_{1,i}, \ldots, t_{n,i})$.*

Démonstration. On applique le lemme 1 à la matrice jacobienne de $\mathbf{f}$ dans l'algèbre des matrices sur $\mathcal{A}[x_1, \ldots, x_n]/(t_1, \ldots, t_n)$. □

D'après l'égalité $\mathbf{Jac(f)} = \mathbf{A}\mathbf{Jac(t)} \mod \mathbf{t}$, on déduit du lemme précédent que pour tout i, $\mathbf{A}_i$ est inversible modulo $\mathbf{t}_i$.

Décrivons maintenant comment calculer le vecteur $\mathbf{t}_{i+1}$ à partir de $\mathbf{t}_i$. Soient

$$\mathbf{T}_i = (T_{1,i}, \ldots, T_{n,i})$$

des polynômes unitaires de $\mathcal{A}_{i+1}[x_1,\ldots,x_n]$ qui coïncident avec $(t_{1,i},\ldots,t_{n,i})$ modulo $\mathcal{I}_i$, et qui ont même support. Les calculs suivants sont effectués dans le quotient

$$Q_i = \mathcal{A}_{i+1}[x_1,\ldots,x_n]/(T_{1,i},\ldots,T_{n,i})$$

muni de sa base canonique. La matrice $\mathbf{Jac}(\mathbf{t})_i\mathbf{Jac}(\mathbf{f})_i^{-1}$ a ses entrées dans l'anneau quotient $\mathcal{A}_i[x_1,\ldots,x_n]/(t_{1,i},\ldots,t_{n,i})$. Notons $\mathbf{N}_i$ une préimage quelconque de cette matrice dans l'anneau des matrices sur $\mathcal{A}[x_1,\ldots,x_n]$, puis, abusivement, son image dans l'anneau des matrices sur le quotient $Q_i[x_1,\ldots,x_n]$. Cette image n'est pas uniquement déterminée, mais sa réduction modulo $\mathcal{I}_i$ l'est.

Proposition 16 (Schéma de calcul de l'itération de Newton-Hensel) *Soit i un entier positif. Appelons δ_i le vecteur produit $\mathbf{N}_i\mathbf{f}_{i+1}$ calculé dans Q_i. Dans $\mathcal{A}_{i+1}[x_1,\ldots,x_n]$, on a l'égalité :*

$$\mathbf{t}_{i+1} = \mathbf{T}_i + \delta_i.$$

Démonstration. Dans $\mathcal{A}_{i+1}$ l'égalité $\mathbf{f}_{i+1} = \mathbf{A}_{i+1}\mathbf{t}_{i+1}$ est vérifiée. Dans l'anneau des matrices à entrées dans le quotient Q_i, les membres gauches et droits de cette égalité sont nuls modulo $\mathcal{I}_i$; autrement dit, ils ont une valuation au moins égale à 2^i. On en déduit que dans cet anneau $\mathbf{t}_{i+1}$ est donné par l'égalité

$$\mathbf{t}_{i+1} = \mathbf{A}_{i+1}^{-1}\mathbf{f}_{i+1}$$

où il suffit de connaître $\mathbf{A}_{i+1}^{-1}$ modulo $\mathcal{I}_i$.

Ensuite, $\mathbf{Jac}(\mathbf{f}) = \mathbf{A}\mathbf{Jac}(\mathbf{t})$ modulo $(t_1,\ldots,t_n)$, on a donc l'égalité $\mathbf{Jac}(\mathbf{f})_i = \mathbf{A}_i\mathbf{Jac}(\mathbf{t})_i$ dans l'anneau des matrices sur le quotient $\mathcal{A}_i[x_1,\ldots,x_n]/(t_{1,i},\ldots,t_{n,i})$. On en déduit que $\mathbf{A}_i^{-1}$ y est égal à $\mathbf{Jac}(\mathbf{t})_i\mathbf{Jac}(\mathbf{f})_i^{-1}$, et cette égalité reste vraie modulo $\mathcal{I}_i$ dans Q_i. Ainsi, dans le quotient Q_i, on a l'égalité

$$\mathbf{t}_{i+1} = \mathbf{N}_i\mathbf{f}_{i+1}.$$

La structure d'ensemble triangulaire fournit une base canonique du quotient Q_i, formée de la réunion des ensembles de monômes $\{1,x_i,\ldots,x_i^{d_i-1}\}$. Exprimée dans cette base, l'image de $\mathbf{t}_{i+1}$ dans Q_i est formée des polynômes $t_{j,i+1} - T_{j,i}$, d'après les conditions que l'on s'est imposées sur les $T_{j,i}$. Ceci prouve notre proposition. □

De manière abusive, le schéma de calcul peut se résumer par la formule

$$\mathbf{t}_{i+1} = \mathbf{t}_i + \mathbf{Jac}(\mathbf{t})_i\mathbf{Jac}(\mathbf{f})_i^{-1}\mathbf{f}_{i+1}.$$

La formulation est proche de celle de l'opérateur utilisé dans l'approximation numérique.

Exemple Reprenons les étapes de la preuve sur l'exemple donné dans l'introduction. Les matrices $\mathbf{Jac}(\mathbf{t})_0$ et $\mathbf{Jac}(\mathbf{f})_0^{-1}$ sont calculées sur l'anneau $(\mathbb{Z}/101\mathbb{Z})[x_1,x_2]/(t_{1,0},t_{2,0})$, ce qui donne une précision suffisante d'après la preuve de la proposition 16.

$$\mathbf{Jac}(\mathbf{t})_0 = \begin{bmatrix} 2x_1 + 64x_2^2 + 99x_2 + 21 & (27x_2+99)x_1 + 16x_2 + 20 \\ 0 & 3x_2^2 + 30x_2 + 64 \end{bmatrix},$$

$$\mathbf{Jac}(\mathbf{f})_0^{-1} =$$

$$\begin{bmatrix} (60x_2^2 + 53x_2 + 14)x_1 + 75x_2^2 + 10x_2 + 40 & (58x_2^2 + 54x_2 + 10)x_1 + 98x_2^2 + 23x_2 + 82 \\ 28x_2^2 + 37 + 54x_2 & 65x_2^2 + 39x_2 + 46 \end{bmatrix}.$$

Pour simplifier la présentation, nous notons toujours $(t_{1,0},t_{2,0})$ les polynômes $(t_{1,0},t_{2,0})$ vus dans $(\mathbb{Z}/(101)^2\mathbb{Z})[x_1,x_2]$. On calcule dans $(\mathbb{Z}/(101)^2\mathbb{Z})[x_1,x_2]$ quotienté par $(t_{1,0},t_{2,0})$:

$$\mathbf{f}_1 \quad = \quad \left[\begin{array}{c} 1010x_2^2 + 1616x_2 + 1616 \\ (1212x_2^2 + 1212x_2 + 1616)x_1 + 3131x_2^2 + 404x_2 \end{array} \right].$$

Finalement, la grandeur $\delta\mathbf{t}_0 = \mathbf{Jac}(\mathbf{t})_0\mathbf{Jac}(\mathbf{f})_0^{-1}\mathbf{f}_1$ vaut

$$\left[\begin{array}{c} (1212x_2^2 + 1212x_2 + 1616)x_1 + 1212x_2^2 + 1414x_2 + 1010 \\ 1010x_2^2 + 1616x_2 + 1616 \end{array} \right].$$

Ainsi, la nouvelle résolution $\mathbf{t}_1 = \mathbf{t}_0 + \delta\mathbf{t}_0$ calculée modulo $(101)^2\mathbb{Z}$ vaut :

$$\begin{aligned} t_{1,1} &= x_1^2 + (1637 + 1311x_2 + 1276x_2^2)x_1 + 1220x_2^2 + 1434x_2 + 1027, \\ t_{2,1} &= x_2^3 + 1025x_2^2 + 1680x_2 + 1644. \end{aligned}$$

Ceci constitue la solution du système modulo $(101)^2\mathbb{Z}$; on peut vérifier qu'il s'agit de la solution sur $\mathbb{Q}$.

Sur la base de la proposition précédente, l'algorithme ci-dessous calcule les approximations $\mathcal{I}_i$-adiques successives $\mathbf{t}_i$ de $\mathbf{t}$ à partir de l'approximation $\mathbf{t}_0$. Par souci de simplicité, l'algorithme donné ici calcule à chaque étape l'inverse de la jacobienne de $\mathbf{f}$. Le calcul itératif des approximations d'inverse donné dans la section 1 offre une meilleure complexité, et est utilisé en pratique. Son écriture modifierait légèrement l'algorithme, puisque l'inverse de la jacobienne devrait être passée en argument, mise à jour et renvoyée après calculs.

FIG. 6.1 – *Opérateur de Newton triangulaire*

procedure `Newton`$(\mathbf{f},\mathcal{I},\mathbf{t}_{i-1})$

\# **Entree**
\# $\mathbf{f}$ système à coefficients dans $\mathcal{A}$
\# $\mathcal{I}$ idéal de $\mathcal{A}$
\# $\mathbf{t}_i = (t_{1,i-1},\ldots,t_{n,i-1})$
\# **Sortie**
\# $\mathbf{t}_{i+1}$
\# Les calculs sont faits dans $\mathcal{A}/\mathcal{I}_{i-1}[x_1,\ldots,x_n]/(t_{1,i-1},\ldots,t_{n,i-1})$
$\mathbf{Jac}(\mathbf{f})_{i-1} \leftarrow$ `Jacobienne`$(\mathbf{f},\mathbf{x})$;
$\mathbf{Jac}(\mathbf{f})_{i-1}^{-1} \leftarrow$ `Inverse`$(\mathbf{Jac}(\mathbf{f})_{i-1})$;
$\mathbf{Jac}(\mathbf{t})_{i-1} \leftarrow$ `Jacobienne`$(\mathbf{t}_{i-1})$;
\# Les calculs sont faits dans $\mathcal{A}/\mathcal{I}_i[x_1,\ldots,x_n]/(t_{1,i-1},\ldots,t_{n,i-1})$
$\delta\mathbf{t} \leftarrow \mathbf{Jac}(\mathbf{t})_{i-1}\mathbf{Jac}(\mathbf{f})_{i-1}^{-1}\mathbf{f}$;
\# Le dernier calcul est fait dans $\mathcal{A}/\mathcal{I}_i$
return $\mathbf{t}_{i-1} + \delta\mathbf{t}$;
end;

6.4 Cas particulier : représentation univariée “Shape Lemma”

Dans les parties V et VI, nous nous intéresserons à la remontée d'une représentation univariée, donnée par un élément primitif $t = \sum u_i x_i$, son polynôme minimal q, et des paramétrisations des variables en fonction de t. Ce cas particulier est traité dès [202, 79, 100, 84] ; la formulation la plus proche de la nôtre est celle de [84].

On peut appliquer le processus de remontée sous des hypothèses plus faibles que dans la section précédente : puisqu'une forme linéaire séparante pour la réduction modulo $\mathcal{I}$ reste séparante pour les puissances de $\mathcal{I}$, il n'est plus nécessaire de supposer assurée l'existence des ensembles $\mathbf{t}_i$ successifs. Dans le cas général, les polynômes $\mathbf{t}_i$ sont supposés unitaires ; dans le cas d'une représentation univariée, il faut donc mettre les paramétrisations sous la forme “Shape Lemma”.

Hypothèse 2 *Nous supposons désormais que :*
- *il existe une combinaison linéaire $t = \sum u_j x_j$ à coefficients dans $\mathcal{A}$ et des polynômes q et $(w_j)_{1\le j\le n}$ de $\mathcal{A}/\mathcal{I}[t]$ tels que $\mathbf{f}(\mathbf{w}) = 0$ et $\sum u_j w_j = t$ dans $\mathcal{A}/\mathcal{I}[t]/(q)$.*
- *la jacobienne du système $\mathbf{f}$ évaluée en les paramétrisations w_i est inversible dans $\mathcal{A}_0[t]/(q)$.*

Nous avons alors un résultat d'unicité de la solution modulo les puissances de $\mathcal{I}$, tiré de [84, Section 4.3].

Proposition 17 *Sous les hypothèses précédentes, pour tout $i \ge 0$ il existe d'uniques polynômes q_i et $(w_{1,i},\dots,w_{n,i})$ de $\mathcal{A}/\mathcal{I}_i[t]$ tels que $\mathbf{f}(\mathbf{w}_i) = 0$ et $\sum u_j w_{j,i} = t$ dans $\mathcal{A}/\mathcal{I}_i[t]/(q_i)$.*

Dans ce cas, on peut appliquer l'opérateur de Newton précédemment défini au système

$$\mathbf{F} = [f_1,\dots,f_n,t - \sum u_j x_j]$$

dans $k[x_1,\dots,x_n,t]$ pour obtenir les approximations successives $\mathcal{R}_i = (t,q_i(t),\mathbf{w}_i(t))$. Le système triangulaire associé est

$$\mathbf{t}_i = (x_1 - w_{1,i}(t),\dots,x_n - w_{n,i}(t),q_i(t)).$$

Le schéma de calcul ainsi obtenu est le même que dans [84, Section 4.3], même si ce dernier n'emploie pas notre formulation matricielle.

Nous pouvons encapsuler tout ceci dans une procédure de remontée de Newton pour représentation univariée. Comme précédemment, il est possible d'utiliser une version itérative des calculs d'inverse de la jacobienne, auquel cas cette inverse doit figurer parmi les entrées et les sorties de l'algorithme.

6.5 Complexité d'une étape de remontée

Notations Les notations suivantes sont introduites dans le chapitre 5.
- Si $\mathcal{A}$ est un anneau commutatif, $\mathbf{t} \subset \mathcal{A}[x_1,\dots,x_n]$ un ensemble triangulaire dont tous les polynômes sont unitaires, et $(d_1,\dots,d_n)$ les degrés des éléments de $\mathbf{t}$ en leur variable principale, $\mathcal{M}(d_1,\dots,d_n)$ désigne le coût de la multiplication dans le quotient $\mathcal{A}[x_1,\dots,x_n]/\mathbf{t}$, donné en nombre d'opérations sur $\mathcal{A}$. Ce coût admet notamment en facteur un terme du type C^n, dû à l'empilement de n quotients univariés.
- On note n^Ω le coût de l'inversion d'une matrice à coefficients dans un anneau commutatif. Une version fine met à jour les inverses successives de la jacobienne de $\mathbf{f}$, et n'opère qu'une seule inversion, à la première étape.

FIG. 6.2 – *Opérateur de Newton univarié*

procedure `NewtonUnivarie`($\mathbf{f}$,$\mathcal{I}$,$\mathcal{R}_{i-1}$)

\# **Entree**
\# $\mathbf{f} = (f_1, \ldots, f_n)$ système à coefficients dans $\mathcal{A}$
\# $\mathcal{I}$ idéal de $\mathcal{A}$
\# $\mathcal{R}_{i-1} = (t, q_{i-1}, \mathbf{w}_{i-1})$ résolution à coefficients dans $\mathcal{I}_{i-1}$.
\# **Sortie**
\# $\mathcal{R}_{i-1} = (t, q_{i-1}, \mathbf{w}_{i-1})$ résolution à coefficients dans $\mathcal{I}_i$.

tmp $\leftarrow$ `Newton`($\mathbf{F}$,$\mathcal{I}$,$\mathcal{R}_{i-1}$);
$q \leftarrow$ tmp$[n+1]$;
$\mathbf{w} \leftarrow [-(\text{tmp}[i] - x_i), i = 1, \ldots, n]$;
return q,$\mathbf{w}$;
end;

Enfin, et surtout, on suppose que le système $\mathbf{f}$ est donné par un calcul d'évaluation de taille L. Cette hypothèse est naturelle: le calcul demande d'évaluer le système $\mathbf{f}$ et sa jacobienne modulo l'ensemble triangulaire $\mathbf{t}_i$, la complexité s'écrit donc naturellement en termes de coût d'évaluation.

Proposition 18 *Soit $a(i)$ le coût de l'arithmétique dans le quotient $\mathcal{A}_i$. Alors le coût de la ième étape de remontée de Newton est*

$$O\big(((n^\Omega + nL)a(i-1) + (n^2 + L)a(i))\mathcal{M}(d_1, \ldots, d_n)\big).$$

Démonstration. Il suffit de détailler le coût de chaque calcul.

- Le calcul de la jacobienne $\mathbf{Jac}(\mathbf{f})_{i-1}$ de $\mathbf{f}$ dans $\mathcal{A}/\mathcal{I}_{i-1}[x_1, \ldots, x_n]/(t_{1,i-1}, \ldots, t_{n,i-1})$ se fait en évaluant cette jacobienne dans le quotient $\mathcal{A}/\mathcal{I}_{i-1}[x_1, \ldots, x_n]/(t_{1,i-1}, \ldots, t_{n,i-1})$. D'après le théorème de Baur et Strassen [20], ce calcul se fait en $O(nL)$ opérations de l'anneau, soit
$$O\big(nL\mathcal{M}(d_1, \ldots, d_n)\big)$$
opérations de $\mathcal{A}/\mathcal{I}_{i-1}$.
- Le calcul dans $\mathcal{A}/\mathcal{I}_{i-1}[x_1, \ldots, x_n]/(t_{1,i-1}, \ldots, t_{n,i-1})$ de l'inverse $\mathbf{Jac}(\mathbf{f})_{i-1}^{-1}$ se fait en $O(n^\Omega)$ opérations , soit
$$O\big(n^\Omega\mathcal{M}(d_1, \ldots, d_n)\big)$$
opérations de $\mathcal{A}/\mathcal{I}_{i-1}$. Cette partie du calcul peut être amélioré par une remise à jour itérative des inverses de la jacobienne.
- Le calcul de la jacobienne de $\mathbf{t}_{i-1}$ se fait en $O(nd_1 \ldots d_n)$ opérations de $\mathcal{A}/\mathcal{I}_{i-1}$.
- L'évaluation du vecteur de polynômes $\mathbf{f}$ dans le quotient $\mathcal{A}/\mathcal{I}_i[x_1, \ldots, x_n]/(t_{1,i-1}, \ldots, t_{n,i-1})$ se fait en $O(L\mathcal{M}(d_1, \ldots, d_n))$ opérations de $\mathcal{A}/\mathcal{I}_i$.
- Le produit matriciel $\mathbf{Jac}(\mathbf{t})_{i-1}\mathbf{Jac}(\mathbf{f})_{i-1}^{-1}\mathbf{f}$ se fait en $O(n^2\mathcal{M}(d_1, \ldots, d_n))$ opérations de $\mathcal{A}/\mathcal{I}_i$.
$\square$

Dans le cas particulier d'une représentation univariée, on obtient un léger raffinement de la proposition précédente qui mérite une proposition à part. Le résultat se trouve sous une forme quasi-identique dans [84]. Nous utilisons les notations de la section précédente, et la notation $\mathcal{M}(d)$, qui désigne le coût du produit de deux polynômes univariés de degré d.

Proposition 19 *La complexité de la ième étape de remontée appliquée à une représentation univariée est :*

$$O\big(((n^\Omega + nL)a(i-1) + (n^2 + L)a(i))\mathcal{M}(d)\big)$$

opérations.

Démonstration. Le coût de l'arithmétique dans le quotient par une résolution de degré d sous la forme "Shape Lemma"est $O(\mathcal{M}(d))$: on gagne un facteur de type C^n par rapport à l'application directe de la proposition précédente, puisqu'on évite l'empilement de n extensions univariées. □

6.6 Cadres d'utilisation

Nous donnons ici un avant-goût des utilisations à venir, qui concerneront l'approximation de résolutions à coefficients dans les séries formelles ou les entiers p-adiques. Pour plus de clarté, dans les exemples, les polynômes d'entrée sont écrits sur la base monomiale.

6.6.1 Remontée des séries formelles

Ici, $\mathcal{A}$ est un anneau de séries formelles $k[[p_1,\ldots,p_m]]$ sur un corps k et $\mathcal{I}$ l'idéal $(p_1,\ldots,p_m)$ de $\mathcal{A}$. Traduits dans ce contexte, nos résultats mènent à la proposition suivante :

Proposition 20 *Soit* $\mathbf{f} = (f_1,\ldots,f_n)$ *une famille de polynômes de* $k[p_1,\ldots,p_m][x_1,\ldots,x_n]$ *et* $\mathbf{p}_1 = (p_{1,1},\ldots,p_{1,m})$ *un point de* k^m*. Soit* $\mathbf{t} = (t_1,\ldots,t_n)$ *un ensemble triangulaire à coefficients dans* $k(p_1,\ldots,p_m)$*, et supposons qu'il existe une matrice* $\mathbf{A}$ *de polynômes à coefficients dans* $k(p_1,\ldots,p_m)$ *telle que* $\mathbf{f} = \mathbf{At}$*.*
Si tous les dénominateurs sont non nuls en $\mathbf{p}_1$ *et si la jacobienne du système* $\mathbf{f}$ *y est inversible modulo* $\mathbf{t}$*, alors à partir de la spécialisation de* $\mathbf{f}$ *en* $\mathbf{p}_1$*, l'opérateur de Newton calcule des ensembles triangulaires dont les coefficients sont les développements de Taylor de ceux de* $\mathbf{t}$ *en* $\mathbf{p}_1$*.*

Démonstration. Après le changement de coordonnées $p'_k = p_i - p_{1,k}$, on effectue la remontée des séries modulo les puissances de l'idéal $(p'_1,\ldots,p'_m)$ de $k[[p'_1,\ldots,p'_m]]$. □

Exemple On considère dans $\mathbb{F}_{9001}[[p_1,p_2]][x_1,x_2]$ les polynômes

$$\begin{aligned} f_1 &= (p_1-3)^2x_1 + (p_2-4)x_2 + x_1x_2 - 1, \\ f_2 &= (p_1-3)(p_2-4)x_2 - x_1^2 + 2. \end{aligned}$$

Soit $\mathcal{I}$ l'idéal (p_1,p_2) de $\mathbb{F}_{9001}[[p_1,p_2]]$. La réduction modulo $\mathcal{I}$ correspond à la spécialisation en zéro ; l'exemple est prévu pour que cette spécialisation soit chanceuse. Modulo $\mathcal{I}$, la solution du système est donnée par :

$$\begin{aligned} t_{1,0} &= x_1 + 8487x_2^2 + 7975x_2 + 5401 \\ t_{2,0} &= x_2^3 + 1517x_2^2 + 6084x_2 + 5264 \end{aligned}$$

On calcule alors la jacobienne de $\mathbf{t}$ et l'inverse de la jacobienne de $\mathbf{f}$ modulo $(t_{1,0},t_{2,0})$.

$$\begin{aligned}
\mathbf{Jac(t)}_0 &= \begin{bmatrix} 1 & 7973x_2+7975 \\ 0 & 3x_2^2+3034x_2+6084 \end{bmatrix}, \\
\mathbf{Jac(f)}_0^{-1} &= \begin{bmatrix} 2162x_2^2+6164x_2+6662 & 5409x_2^2+671x_2+638 \\ 5625x_2^2+8768x_2+165 & 4853x_2^2+5666x_2+204 \end{bmatrix}.
\end{aligned}$$

Comme dans l'exemple précédent, notons abusivement $(t_{1,0},t_{2,0})$ les polynômes $(t_{1,0},t_{2,0})$, vus cette fois dans $\mathbb{F}_{9001}[[p_1,p_2]]/(p_1,p_2)^2$. On calcule dans le quotient $\mathbb{F}_{9001}[[p_1,p_2]]/(p_1,p_2)^2[x_1,x_2]/(t_{1,0},t_{2,0})$. L'évaluation de (f_1,f_2) est

$$\mathbf{f}_1 = \begin{bmatrix} 5917p_1x_2^2+2845p_1x_2+5403p_1+p_2x_2 \\ 8997p_1x_2+8998p_2x_2 \end{bmatrix}.$$

Dans ce quotient, le produit $\mathbf{Jac(t)}_0\mathbf{Jac(f)}_0^{-1}\mathbf{f}_1$ est

$$\begin{bmatrix} 3505p_1x_2^2+6237p_1x_2+1903p_1+4754p_2x_2^2+1793p_2x_2+103p_2 \\ 3488p_1x_2^2+1891p_1x_2+4737p_1+5626p_2x_2^2+2251p_2x_2+1316p_2 \end{bmatrix}.$$

On rajoute $(t_{1,0},t_{2,0})$ pour obtenir l'ensemble recherché modulo $((p_1,p_2)^2)$:

$$f_{1,1} = x_1+(8487+3505p_1+4754p_2)x_2^2+(797+6237p_1+1793p_2)x_2+5401+1903p_1+103p_2,$$

$$f_{2,1} = x_2^3+(1517+3488p_1+5626p_2)x_2^2+(6084+1891p_1+2251p_2)x_2+5264+4737p_1+1316p_2.$$

On peut vérifier que les coefficients sont les développements de Taylor à l'ordre deux à l'origine des fractions rationnelles qui apparaissent dans la base de Gröbner du système dans l'anneau $\mathbb{F}_{9001}(p_1,p_2)[x_1,x_2]$. On peut également vérifier que le calcul dans

$$\mathbb{F}_{9001}[[p_1,p_2]]/(p_1,p_2)^2[x_1,x_2]/(T_{1,0},T_{2,0}),$$

où $(T_{1,0},T_{2,0})$ sont des polynômes arbitraires coïncidant avec $(t_{1,0},t_{2,0})$ modulo (p_1,p_2) et ayant même support, donne le même résultat.

6.6.2 Remontée des entiers par les nombres p-adiques

Si $\mathcal{A}$ est l'anneau des entiers, et l'idéal $\mathcal{I}$ est engendré par un nombre premier p, nous obtenons alors un développement p-adique des coefficients rationnels d'un ensemble triangulaire, à partir de sa réduction modulo p. La preuve de la proposition ci-dessous est calquée sur celle de la section précédente, en remplaçant les séries formelles autour de $\mathbf{p}$ par les nombres p-adiques.

Proposition 21 *Soient* $\mathbf{f} = (f_1,\ldots,f_n)$ *des polynômes de* $\mathbb{Q}[x_1,\ldots,x_n]$, *$p$ un nombre premier, et* $\mathbf{t}$ *un ensemble triangulaire à coefficients dans* $\mathbb{Q}$ *; supposons qu'il existe une matrice* $\mathbf{A}$ *à entrées dans* $\mathbb{Q}[x_1,\ldots,x_n]$ *telle que* $\mathbf{f} = \mathbf{At}$.
Supposons que modulo p, tous les dénominateurs sont non nuls et que la jacobienne du système $\mathbf{f}$ *est inversible modulo* $\mathbf{t}$. *Alors à partir de la spécialisation de* $\mathbf{f}$ *modulo p l'opérateur de Newton calcule des ensembles triangulaires dont les coefficients sont les développements p-adiques de ceux de* $\mathbf{t}$.

Dans un autre registre, la proposition suivante interviendra dans le chapitre 13.

Proposition 22 *Soient* $\mathbf{f} = (f_1,\ldots,f_n)$ *des polynômes de* $\mathbb{Q}[x_1,\ldots,x_n]$, $\mathcal{U}$ *l'ensemble des zéros simples de* $\mathbf{f}$ *et* $\mathcal{R}(t) = (t,q,\mathbf{v})$ *une résolution de Kronecker des points de* $\mathcal{U}$ *dans* $\mathbb{C}^n$.
Soit p *un nombre premier tel que, modulo* p*, la jacobienne du système* $\mathbf{f}$ *est inversible en tous les points décrits par* $\mathcal{R}(t)$ *et telle que les dénominateurs intervenant dans* $\mathbf{f}$ *et* $\mathcal{R}(t)$*, ainsi que le discriminant de* q *sont non nuls. Alors* tous *les zéros simples du système* $\mathbf{f}$ *pris modulo* p *sont décrits par la réduction de* $\mathcal{R}(t)$ *modulo* p.

Démonstration. Notons $\mathcal{D}_p$ l'ensemble des zéros simples définis par le système $\mathbf{f}$ dans $\overline{\mathbb{F}_p}^n$. La spécialisation de $\mathcal{R}(t)$ modulo p décrit une partie des zéros simples de $\mathbf{f}$ modulo p. Il suffit de montrer qu'on les obtient tous ainsi. Notant $\mathcal{D}$ le degré de la variété $\mathcal{U}$ dans $\mathbb{C}^n$, cela revient à montrer que $\mathcal{D}_p \leq \mathcal{D}$.
Soit $\mathcal{R}(t)_p$ une résolution décrivant l'ensemble des $\mathcal{D}_p$ zéros simples de $\mathbf{f}$ dans une clôture algébrique de $\mathbb{F}_p$. D'après la proposition 17, on peut remonter cette résolution modulo toutes les puissances de p. En passant à la limite, on obtient l'existence de $\mathcal{D}_p$ solutions simples au système $\mathbf{f}$ dans une clôture algébrique du corps des nombres p-adiques $\mathbb{Q}_p$. Voyons $\mathbb{Q}$ comme un sous-corps de $\mathbb{Q}_p$. Les $\mathcal{D}_p$ points limites sont algébriques sur $\mathbb{Q}$, leur nombre ne peut donc pas dépasser $\mathcal{D}$. □

6.6.3 Enchaînement des deux remontées

Proposition 23 *Soit* $\mathbf{f} = (f_1,\ldots,f_n)$ *une famille de polynômes de* $\mathbb{Z}[p_1,\ldots,p_m][x_1,\ldots,x_n]$, $\mathbf{p}_1 = (p_{1,1},\ldots,p_{1,m})$ *un point de* $\mathbb{Z}^m$ *et* p *un nombre premier. Soient ensuite* $\mathbf{t}$ *un ensemble triangulaire à coefficients dans* $\mathbb{Q}(p_1,\ldots,p_m)$*, tel qu'il existe une matrice* $\mathbf{A}$ *à coefficients dans* $\mathbb{Q}(p_1,\ldots,p_m)$ *pour laquelle on a l'égalité* $\mathbf{f} = \mathbf{At}$.
Supposons que tous les dénominateurs sont inversibles modulo $(p_1 - p_{1,1},\ldots,p_m - p_{1,m}) + (p)$ *et que la jacobienne du système* $\mathbf{f}$ *est inversible sur les points décrits par* $\mathbf{t}$ *pris modulo* $(p_1 - p_{1,1},\ldots,p_m - p_{1,m}) + (p)$.
Sous ces hypothèses, on peut calculer le développement de Taylor de la résolution $\mathbf{t}$ *à précision arbitraire en* $(p_1 - p_{1,1},\ldots,p_m - p_{1,m})$ *et* (p) *par l'opérateur de Newton.*

Démonstration. Après le changement de coordonnées $p'_k = p_i - p_{1,k}$, on effectue la remontée des séries modulo les puissances de l'idéal $(p'_1,\ldots,p'_m)$ de $\mathbb{Z}/p\mathbb{Z}[[p'_1,\ldots,p'_m]]$, puis la remontée des entiers modulo les puissances de l'idéal (p) de l'anneau des séries formelles (tronquées) $\mathbb{Z}[[p'_1,\ldots,p'_m]$. □

Exemple Considérons dans $\mathbb{Q}[p][x]$ l'équation

$$\mathbf{f} : f_1 = (1+2p)x^2 + (3+4p)x + p.$$

Nous cherchons à développer l'équation

$$x^2 + \frac{3+4p}{1+2p}x + \frac{p}{1+2p}$$

modulo les puissances de l'idéal (5) autour du point $p = 0$. La résolution de cette équation modulo $(5,p)$ est

$$x^2 + 3x.$$

Commençons par remonter les fractions à précision $O(p^4)$, pour obtenir

$$t_4 = x^2 + (3 + 3p + 4p^2 + 2p^3)x + p + 3p^2 + 4p^3.$$

Ensuite, nous remontons les entiers sur ces séries, à précision $O(5^8)$, en obtenant successivement

$$\begin{aligned}
t_4^2 &= x^2 + (3 + 23p + 4p^2 + 17p^3)x + p + 23p^2 + 4p^3 \\
t_4^4 &= x^2 + (3 + 623p + 4p^2 + 617p^3) + p + 623p^2 + 4p^3 \\
t_4^8 &= x^2 + (3 + 390623p + 4p^2 + 390617p^3)x + p + 390623p^2 + 4p^3,
\end{aligned}$$

dont on peut vérifier qu'il s'agit du développement de Taylor à l'ordre $O(p^4)$ de f_1, modulo 5^8.

Dans l'autre sens, nous commençons par calculer la résolution modulo 5^8. Nous obtenons

$$t^8 = x^2 + 3x.$$

Remontons maintenant les séries formelles, pour avoir :

$$\begin{aligned}
t_2^8 &= x^2 + (3 + 390623p)x + p \\
t_4^8 &= x^2 + (3 + 390623p + 4p^2 + 390617p^3)x + p + 390623p^2 + 4p^3,
\end{aligned}$$

qui coïncide avec le développement précédent.

6.6.4 Précautions d'emploi

Les hypothèses employées pour prouver l'algorithme sont fortes : il faut savoir que l'on part de la réduction d'un ensemble triangulaire. Vue "d'en-bas", la situation modulo $\mathcal{I}$ ne permet pas de s'assurer que ces hypothèses sont satisfaites : l'existence d'une solution représentable par un ensemble triangulaire non trivial (où la dernière variable n'est pas séparante) est une condition non génériquement vraie, donc instable :

Exemple Posons dans $\mathbb{Q}[t][x,y]$:

$$\begin{aligned}
f_1 &= x^2 - 2 + yt \\
f_2 &= y^2 - 1 + xt.
\end{aligned}$$

Spécialisons en $t = 0$, pour obtenir la solution à précision $O(t)$. Le système est déjà triangulaire :

$$\begin{aligned}
t_{1,0} &= x^2 - 2 \\
t_{2,0} &= y^2 - 1.
\end{aligned}$$

Cependant, on ne peut pas écrire la solution à précision $O(t^2)$ sous la forme $(t_{1,1}(x,y),t_{2,1}(y))$, avec $t_{2,1}$ de degré 2. La base de Gröbner de l'idéal (f_1,f_2) dans $\mathbb{Q}(t)[x,y]$ muni de l'ordre lexicographique $x > y$ est :

$$\begin{gathered}
x + \frac{1}{t}y^2 - \frac{1}{t}, \\
y^4 - 2y^2 + t^3y - 2t^2 + 1.
\end{gathered}$$

Cette base ne se spécialise pas en $t = 0$, ce qui explique nos ennuis. Nous donnons dans le chapitre 18 un ensemble de conditions assurant qu'une réduction est chanceuse, qui permettent de quantifier les dégénérescences.

6.7 Mise en pratique

Cette section décrit quelques points pratiques liés à l'utilisation de l'opérateur de Newton.

6.7.1 Factorisation

Hypothèse 3 *Nous supposons ici que le polynôme $t_{n,0}$ se factorise dans l'anneau $\mathcal{A}/\mathcal{I}[x_n]$ en produit de deux facteurs : $t_{n,0} = t_{n,0}^{(1)} t_{n,0}^{(2)}$. Les polynômes $t_{n,0}^{(j)}$ sont des polynômes unitaires de $\mathcal{A}/I[x_n]$, et engendrent l'anneau $\mathcal{A}_i[x_n]$ en entier.*

Sous ces nouvelles conditions, l'idéal engendré par l'ensemble triangulaire $\mathbf{t}_0$ est engendré par les deux ensembles $\mathbf{t}_0^{(j)}$, avec $j = 1{,}2$:

$$\mathbf{t}_0^{(j)} = \begin{cases} t_{1,0}^{(j)}(x_1,x_2,\ldots,x_n) \\ t_{2,0}^{(j)}(x_2,\ldots,x_n) \\ \vdots \\ t_{n,0}^{(j)}(x_n), \end{cases}$$

où pour $k < n$, le polynôme $t_{k,0}^{(j)}$ est la réduction de $t_{k,0}$ par $t_{n,0}^{(j)}$.

Proposition 24 *Pour tout $i \geq 0$, le polynôme $t_{n,i}$ se factorise en deux facteurs $t_{n,i}^{(1)}, t_{n,i}^{(2)}$ unitaires, dans $\mathcal{A}_i[x_n]$, tels que $t_{n,i+1}^{(j)} = t_{n,i}^{(j)}$ modulo $\mathcal{I}_i$ pour $j = 1{,}2$.*

Démonstration. C'est le lemme de Hensel dans une forme plus classique, voir par exemple [65, Chapter 7]. □

En conséquence, pour tout $i \geq 0$, il existe des ensembles triangulaires $\mathbf{t}_i^{(j)}$, avec $j = 1{,}2$, tels que $\mathbf{t}_i^{(j)}$ est égal à $\mathbf{t}_i$ réduit par $t_{n,i}^{(j)}$. Montrons que la remontée de Newton peut se faire séparément sur chacun des facteurs.

Lemme 3 *Il existe une suite de matrices $\mathbf{B}_i$ de $\mathcal{A}_i[x_1,\ldots,x_n]$ telle que pour tout $i \geq 0$, on a :*
- $\mathbf{B}_{i+1} = \mathbf{B}_i$ *modulo* $\mathcal{I}_i\mathcal{A}_{i+1}$,
- $\mathbf{t}_i = \mathbf{B}_i \mathbf{t}_i^{(1)}$.

Démonstration. Construisons les différentes lignes des matrices $\mathbf{B}_i$:
- Pour $k = n$, $t_{n,i} = t_{n,i}^{(1)} t_{n,i}^{(2)}$. On met donc $t_{n,i}^{(2)}$ à la dernière colonne, et zéro ailleurs.
- Pour $k < n$, il faut un peu plus d'efforts. Considérons le polynôme $t_{k,i}$ comme un polynôme multivarié à variables $(x_1,\ldots,x_{n-1})$ et à coefficients dans $\mathcal{A}_i[x_n]$. Le polynôme $t_{k,i}^{(1)}$ est obtenu par la réduction de chacun de ces coefficients par $t_{n,i}^{(1)}$. Isolons un de ces coefficients c_i, et sa réduction $c_i^{(1)}$. Alors $t_{n,i}^{(1)}$ divise $c_i - c_i^{(1)}$. On applique le lemme de Hensel, pour obtenir l'existence d'une suite de polynômes q_i de $\mathcal{A}_i[x_n]$ telle que $q_{i+1} = q_i$ modulo $\mathcal{I}_i$ et
$$c_i = q_i t_{n,i}^{(1)} + c_i^{(1)}.$$
Ceci permet de construire la $k$ème ligne de la matrice $\mathbf{B}_i$.

□

Proposition 25 *L'application de l'algorithme de Newton donné dans la section 6.3 à partir du facteur $\mathbf{t}_0^{(1)}$ calcule les approximations successives $\mathbf{t}_i^{(1)}$.*

Démonstration. Nous avions déjà la suite d'égalités $\mathbf{f}_i = \mathbf{A}_i\mathbf{t}_i$; à l'aide du lemme précédent, nous obtenons la suite d'égalités

$$\mathbf{f}_i = \mathbf{A}_i\mathbf{B}_i\mathbf{t}_1^{(1)}.$$

On applique alors le schéma de démonstration de la proposition 16. □

On obtient ainsi les $\mathbf{t}_i^{(1)}$ successifs et les $\mathbf{t}_i^{(2)}$, de manière séparée. Il est possible de les combiner pour obtenir les $\mathbf{t}_i$ successifs, par le théorème des restes chinois.

Corollaire 5 *Plaçons-nous sous les hypothèses précédentes. Notons $d_n^{(1)}$ et $d_n^{(2)}$ les degrés des polynômes $t_{n,0}^1$ et $t_{n,0}^2$ en la variable x_n. Alors dans l'estimation de complexité, le terme*

$$\mathcal{M}(d_1,\dots,d_n)$$

est remplacé par

$$\mathcal{M}(d_1,\dots,d_{n-1},d_n^{(1)}) + \mathcal{M}(d_1,\dots,d_{n-1},d_n^{(2)}).$$

La section 6.8 montre les conséquences pratiques de ce corollaire.

Pour finir, nous annonçons sans démonstration la propriété suivante, qui étend le résultat démontré ci-dessus. La démonstration est la même, mais plus fastidieuse, car elle nécessite de manipuler davantage d'indices.

Proposition 26 *Supposons plus généralement qu'un polynôme $t_{j_0,0}$ se factorise modulo les polynômes $(t_{j_0+1},\dots,t_n)$ en deux facteurs premiers entre eux. Scindons l'ensemble $\mathbf{t}_0$ en conséquence, en deux ensembles $\mathbf{t}_0^{(1)}$ et $\mathbf{t}_0^{(2)}$. Alors la remontée peut s'effectuer sur chacun des deux facteurs par le même schéma de calcul que dans le cas général.*

6.7.2 Accélération du calcul de l'inverse de la jacobienne

La pratique de calculs de remontée de Newton montre que le calcul des inverses de la jacobienne du système $\mathbf{f}$ dans les quotients successifs est très coûteux, aussi toute possibilité d'accélérer ces calculs est-elle la bienvenue.

- Nous avons déjà mentionné plusieurs fois la mise à jour itérative des inverses par le calcul donné en section 6.2 ; cette solution remplace un calcul d'inverse par deux produits matriciels.
- Citons également les calculs d'algèbre linéaire "creuse" : si $\mathbf{A}$ est une matrice de $\mathcal{M}_n(\mathcal{A})$ inversible, si on note a le coût de la multiplication de $\mathbf{A}$ par un vecteur alors on peut calculer le produit $\mathbf{A}^{-1}\mathbf{x}$ en $O(an)$ opérations de $\mathcal{A}$, incluant des divisions, par l'algorithme de Lanczos [133] ou de Wiedemann [207].
- Finalement, les algorithmes d'algèbre linéaire structurée [178, 28, 163, 66] peuvent tout aussi bien s'appliquer, pour des jacobiennes de type Toeplitz ou quasi-Toeplitz.

6.7.3 Procédure de remontée

Nous encapsulons l'opérateur dans une procédure de remontée notée `RemonteeNewton`. Nous l'utiliserons dans divers cadres, aussi en donne-t-on une présentation générique. Pour un anneau commutatif unitaire intègre $\mathcal{A}$, de corps des fractions $\mathrm{fr}(\mathcal{A})$, cette fonction prend en arguments :

- un système $\mathbf{f}$ à coefficients dans l'anneau $\mathrm{fr}(\mathcal{A})[x_1,\dots,x_n]$,
- un idéal $\mathcal{I}$ de $\mathcal{A}$,

– la réduction modulo $\mathcal{I}$ de la résolution $\mathcal{R}$ (triangulaire ou univariée) d'une composante de l'ensemble des zéros de $\mathbf{f}$. Nous supposerons que les conditions d'application de l'opérateur de Newton sont remplies. Le calcul fournit alors des résolutions $\mathcal{R}_i$ qui sont solutions de $\mathbf{f}$ modulo les puissances $\mathcal{I}^{2^i}$.
– une procédure `Arret`, qui effectue un test d'arrêt et une éventuelle reconstruction sur chaque résolution $\mathcal{R}$. Si ce test répond "oui", nous arrêtons les calculs.
– une borne *a priori* sur le nombre d'étapes, pour arrêter les calculs.

FIG. 6.3 – *Remontée par Newton-Hensel*

procedure `RemonteeNewton`($\mathbf{f}$,$\mathcal{I}$,$\mathcal{R}$,`Arret`,PrecisionMax)

\# **Entree**
\# $\mathbf{f}$: système à coefficients dans fr($\mathcal{A}$)[$x_1,\dots,x_n$]
\# $\mathcal{I}$: idéal d'un anneau $\mathcal{A}$
\# $\mathcal{R}$: résolution de $\mathbf{f}$ modulo $\mathcal{I}$
\# `Arret` : fonction de test d'arrêt et reconstruction de la résolution
\# PrecisionMax : borne sur le nombre d'étapes
\# **Sortie**
\# $\mathcal{R}'$ une résolution satisfaisant `Arret`, ou "echec"

$b \leftarrow$ **false** ;
precision $\leftarrow$ 1;
while (precision $\leq$ PrecisionMax) **do**
  $\mathcal{R} \leftarrow$ `Newton`($\mathbf{f}$,$\mathcal{I}$,$\mathcal{R}$,precision);
  b,$\mathcal{R}' \leftarrow$ `Arret`($\mathcal{R}$);
  if b **then return** $\mathcal{R}'$ **fi** ;
  precision $\leftarrow$ 2 precision;
od ;
return "echec";
end;

6.8 Implantation en Magma

Pour finir, voici un aperçu des performances d'une implantation en Magma de la procédure de remontée de Newton formelle. Dans le cas particulier d'une représentation univariée, le prototype `kronecker` fournit les procédures nécessaires, et effectue les calculs décrits dans [84]. Pour le cas général, j'ai écrit un code spécifique.

Prenons comme exemple le système Hawes4, qui est donné dans l'annexe E.9. Dans le tableau ci-dessous, Hawes4 (développé) désigne le système tel qu'il est disponible sur le serveur PolyData [15], sous forme développée. Avec G. Lecerf, nous l'avons réécrit sous la forme plus compacte donnée dans l'annexe E.9.
Nous nous donnons une résolution de ce système spécialisé en la valeur des paramètres (123,456), modulo le nombre premier $p = 100000007$. D'un côté, nous relevons les entiers de cette résolution ;

FIG. 6.4 – *Mesure du système-test*

	Hawes4	Hawes4 (développé)
Variables	6	6
Parametres	2	2
Multiplications	55	92

de l'autre, les séries formelles en les paramètres. Pour travailler avec des objets de tailles comparables dans les deux cas, on relève les séries le long d'une droite de l'espace des paramètres, en remplaçant les paramètres par $(123+s,456+s)$. Nous allons notamment mesurer l'influence de la factorisation et de la complexité d'évaluation du système sur le coût de la remontée.

- Dans un premier temps, nous avons trouvé une variable séparante et utilisé l'opérateur de Newton univarié ; le degré générique est 120.
- Le système admet également une résolution par un ensemble triangulaire, qui n'est pas sous forme "Shape Lemma", la suite des degrés étant (1,1,1,1,2,60). On y applique l'opérateur de Newton général.
- On effectue une remontée à précision 64 : dans le cas des entiers, on obtient une résolution à coefficients modulo p^{64} ; dans le cas des séries, les coefficients sont des séries univariées à précision 64.

FIG. 6.5 – *Remontée à précision 64 par l'opérateur de Newton univarié (*`kronecker`*)*

Temps	Hawes4	Hawes4 (développé)
Entiers	35 s.	58 s.
Entiers, avec factorisation	57 s.	141 s.
Séries	677 s.	1246 s.
Séries, avec factorisation	307 s.	507 s.

Conclusion L'influence de la complexité d'évaluation du système est claire : le système Hawes4 retravaillé s'évalue deux fois mieux que sa version développée, le temps de calcul devient deux fois plus court. La dépendance linéaire en le paramètre L de la complexité se traduit bien en pratique.

La remontée des entiers se distingue de la remontée des séries. Dans le premier cas, la solution la plus rapide est le calcul sans factorisation : l'arithmétique entière de Magma est bonne, et le calcul

FIG. 6.6 – *Remontée à précision 64 par l'opérateur de Newton triangulaire*

Temps	Hawes4	Hawes4 (développé)
Entiers	65 s.	112 s.
Entiers, avec factorisation	61 s.	92 s.
Séries	955 s.	2155 s.
Séries, avec factorisation	492 s.	1073 s.

factorisé peut provoquer une perte de temps : cela indique que le temps passé dans le contrôle, les coercions des objets,... n'est plus négligeable. Dans le cas des séries, la factorisation apporte un gain de temps appréciable. Enfin, le calcul par des ensembles triangulaires est plus lent que le calcul d'une résolution univariée, ce qui est dû à l'empilement des structures. L'annexe D donne une évaluation des performances de l'arithmétique de Magma sur différents domaines, qui explique les différences constatées ici.

Cinquième partie

Calcul d'une résolution de Kronecker générique

Chapitre 7

Vue d'ensemble

7.1 Hypothèses et notations

Dans cette partie, nous considérons un corps k et un système de polynômes $\mathbf{f} = (f_1, \ldots, f_n)$ de $k[p_1, \ldots, p_m, x_1, \ldots, x_n]$. Les variables $\mathbf{p} = (p_1, \ldots, p_m)$ sont considérées comme des paramètres, les variables $\mathbf{x} = (x_1, \ldots, x_n)$ comme de véritables inconnues. Nous allons décrire les inconnues $\mathbf{x}$ en fonction de $\mathbf{p}$ par l'intermédiaire d'une résolution de Kronecker à coefficients dans $k(p_1, \ldots, p_m)$. Nous travaillerons toujours avec les notations ci-dessous.

Notations 1 *Nous noterons $\mathcal{U}$ la réunion des composantes de $\mathcal{V}(f_1, \ldots, f_n) \subset \mathbb{A}^{m+n}(\overline{k})$ où le déterminant jacobien de $\mathbf{f}$ par rapport aux variables $\mathbf{x}$ n'est pas identiquement nul.*

L'entrée est mesurée par des grandeurs syntaxiques :

- *le degré partiel en $(x_1, \ldots, x_n)$ des polynômes $(f_1, \ldots, f_n)$ est borné par $d_{\mathbf{x}}$;*
- *le degré total des polynômes $(f_1, \ldots, f_n)$ est borné par $d_{\mathbf{p},\mathbf{x}}$, supposé au moins égal à 2 ;*
- *le système $(f_1, \ldots, f_n)$ est donné par un calcul d'évaluation de taille L.*

Les objets sous-jacents sont mesurés par des degrés géométriques :

- *$\mathcal{D}_{\mathbf{x}}$ est le nombre générique de solutions simples du système $(f_1, \ldots, f_n)$. Par le théorème de Bézout, ce nombre est inférieur au produit des degrés partiels en $\mathbf{x}$ des polynômes f_i, donc à $d_{\mathbf{x}}^n$.*
- *$\mathcal{D}_{\mathbf{p},\mathbf{x}}$ désigne le degré de la variété $\mathcal{U}$. Par le théorème de Bézout, ce degré est inférieur au produit des degrés des polynômes f_i, donc à $d_{\mathbf{p},\mathbf{x}}^n$.*

Le but de cette partie est de *calculer une résolution de Kronecker* des zéros de l'idéal $\mathcal{I}(\mathcal{U})$ étendu dans l'anneau de polynômes $k(p_1, \ldots, p_m)[x_1, \ldots, x_n]$. Nous appellerons ces zéros les *zéros simples génériques* du système $(f_1, \ldots, f_n)$ et une résolution à coefficients dans $k(p_1, \ldots, p_m)$ une *résolution générique.*

S'il existe une composante $\mathcal{U}'$ de l'ensemble des zéros simples génériques que l'on sait *particulariser*, nos résultats s'y appliquent de façon immédiate. En particulier, si l'on se donne une équation supplémentaire f_{n+1}, nos méthodes permettent de calculer une résolution de l'ensemble des points simples génériques qui annulent ou n'annulent pas f_{n+1}.

Les notations de complexité utilisées dans cette partie ont été définies dans le chapitre 5 :

- la notation $\mathcal{M}(d)$ désigne le coût de la multiplication des polynômes univariés de degré d sur un anneau $\mathcal{A}$, en nombre d'opérations de $\mathcal{A}$ — dans cette partie, $\mathcal{A}$ sera un anneau de séries formelles ;

– la notation $\mathcal{M}_s(d,m)$ désigne le coût de la multiplication des séries formelles en m variables à précision d sur un anneau $\mathcal{A}$, en nombre d'opérations de $\mathcal{A}$;
– la notation n^Ω indique le coût de l'algèbre linéaire pour des matrices de taille $n \times n$ sur un anneau $\mathcal{A}$.

Enfin, nous utiliserons les dénominations suivantes :
– Nous appellerons *élément génériquement séparant* une forme linéaire $\sum u_i x_i$ à coefficients dans k qui sépare génériquement les points de $\mathcal{U}$.
– Si t est un élément génériquement séparant, nous noterons $\mathcal{R}(t)$ la résolution de Kronecker générique avec t pour élément séparant, et $\mathcal{D}(t)$ le maximum des degrés en $(p_1,\ldots,p_m)$ des numérateurs et dénominateurs de ses coefficients.
– Nous dirons qu'une valeur des paramètres $\mathbf{p}$ *satisfait toutes les hypothèses de généricité pour* t si la résolution générique $\mathcal{R}(t)$ se spécialise en une résolution des zéros simples du système $\mathbf{f}(\mathbf{p},.)$.

7.2 Résultats et plan des chapitres suivants

Le chapitre 8 est une étude géométrique. Nous y démontrons essentiellement des majorations de degré sur une résolution générique, puis sur une hypersurface discriminante qui contient les valeurs des paramètres ne satisfaisant pas les hypothèses de généricité. Les techniques utilisées remontent au moins à Macaulay [150] pour l'utilisation d'un élément primitif générique, et ont été depuis réutilisées à plusieurs reprises [179, 9] ; les majorations de degré s'inspirent de l'article de Sabia et Solernó [190].

Théorème 6 *Soit t un élément génériquement séparant, de la forme $\sum u_i x_i$, où les u_i sont dans k. La résolution $\mathcal{R}(t)$*

$$q(t) = 0, \quad \left\{ \begin{array}{rcl} \frac{dq(t)}{dt} x_1 & = & v_1(t), \\ & \vdots & \\ \frac{dq(t)}{dt} x_n & = & v_n(t), \end{array} \right.$$

satisfait les propriétés suivantes : les polynômes q et v_i sont dans $k(p_1,\ldots,p_m)[t]$, le polynôme q est unitaire de degré $\mathcal{D}_{\mathbf{x}}$, les polynômes v_i sont de degré strictement inférieur à $\mathcal{D}_{\mathbf{x}}$ et les numérateurs et le ppcm des dénominateurs des coefficients de ces polynômes sont de degré en $(p_1,\ldots,p_m)$ inférieur à $\mathcal{D}_{\mathbf{p},\mathbf{x}}$.
Le lieu des points $\mathbf{p}$ où cette résolution ne se spécialise pas en une résolution des points simples de $\mathbf{f}(\mathbf{p},.)$ est inclus dans une hypersurface de $\mathbb{A}^m(\overline{k})$ de degré inférieur à

$$d^n_{\mathbf{p},\mathbf{x}}(2d^n_{\mathbf{x}} + n d_{\mathbf{p},\mathbf{x}}).$$

Ces quantifications permettront en particulier de contrôler de la probabilité d'erreur de l'algorithme du chapitre 10.

Remarque Dans le théorème ci-dessus, et dans toute la suite, nous avons décidé de majorer des degrés géométriques par des termes en $d^n_{\mathbf{x}}$ et $d^n_{\mathbf{p},\mathbf{x}}$. Cette majoration n'est pas la plus fine possible : nous pourrions la remplacer par le *produit des degrés* (resp. partiels ou totaux). Cette amélioration se ferait au détriment de la lisibilité des résultats, aussi nous gardons la forme ci-dessus. Dans le chapitre 11, nous chiffrons explicitement des probabilités d'erreur et nous employons la forme plus fine.

Le chapitre 9 introduit des fonctions auxiliaires sur lesquelles nous baserons nos calculs. Ces fonctions concernent :

- la reconstruction de fractions rationnelles à partir de leur développement en série ;
- un test probabiliste pour décider si une résolution à coefficients dans $k(p_1,\ldots,p_m)$ est bien la résolution recherchée.

Nous donnons des estimations sur la complexité et la probabilité d'erreur de ces opérations, basées sur les rappels du chapitre 5 et sur les estimations de degré du chapitre 8.

Le chapitre 10 contient l'algorithme principal de cette partie. Le calcul se base sur l'opérateur de Newton formel présenté dans le chapitre 6 : on procède par approximations, à partir d'une résolution des zéros simples du système spécialisé $\mathbf{f}(\mathbf{p},.)$, pour une valeur de $\mathbf{p}$ satisfaisant les conditions de généricité. Le résultat principal est le théorème de complexité ci-dessous, où l'on note $\mathcal{C}_{\text{Zéro}}(\mathbf{f},\mathbf{p},k)$ le coût du calcul spécialisé en le paramètre $\mathbf{p}$.

Théorème 7 (Complexité du calcul de la résolution générique) *On peut calculer par un algorithme probabiliste une résolution univariée des zéros simples génériques du système* $\mathbf{f}$ *en*

$$\mathcal{C}_{Z\acute{e}ro}(\mathbf{f},\mathbf{p},k) + O_{log}\big((n^{\Omega} + nL + nm)\mathcal{M}(\mathcal{D}_{\mathbf{x}})\mathcal{M}_s(4\mathcal{D}(t),m)\big)$$

opérations de k*, où* $\mathbf{p}$ *est une valeur aléatoire des paramètres et* t *est un élément séparant génériquement les points de* $\mathcal{U}$*. Les choix des valeurs aléatoires intervenant dans l'algorithme qui mènent à l'échec sont enfermés dans une hypersurface algébrique de* $\mathbb{A}^{3m-1}(\overline{k})$ *de degré au plus* $80nd_{\mathbf{x}}^{n}d_{\mathbf{p},\mathbf{x}}^{3n}$.

Le terme $\mathcal{M}(\mathcal{D}_{\mathbf{x}})\mathcal{M}_s(4\mathcal{D}(t),m)$ est le coût de la multiplication modulo la dernière approximation calculée, le coût du calcul est donc proche de l'optimal. En d'autres termes, la taille des données intermédiaires n'explose pas. L'algorithme nécessite de tirer $3m-1$ valeurs aléatoires dans k, ce qui explique que l'hypersurface à éviter est incluse dans $\mathbb{A}^{3m-1}(\overline{k})$.

Le dernier chapitre décrit l'implantation de cet algorithme dans le langage Magma. Dans cette partie, nous ne considérons que des résolutions sous forme univariée, aussi nous réemployons l'opérateur de Newton univarié écrit par G. Lecerf pour le prototype `kronecker`. Nous avons compté les opérations dans le corps de base comme de coût constant, nous n'effectuons donc que des calculs sur des corps finis, où cette hypothèse est vérifiée avec un bon degré de vraisemblance ; la partie suivante décrira le cas du calcul sur les rationnels. Nous comparons nos résultats à des calculs de bases de Gröbner effectués en Magma. Nos essais montrent que la solution que nous proposons se comporte bien, et fait aboutir des calculs autrement difficiles, voire impossibles, à mener à bien par cette autre méthode

Les points suivants résument en partie l'expérience de ce modèle de calcul.

- Le déroulement de l'algorithme est linéaire : il n'y pas de branchement ni de test, excepté pour détecter la fin des calculs. Ce comportement itératif est à l'opposé du comportement peu “prévisible” d'un calcul par exemple de base de Gröbner : au vu des premières étapes de calcul, on peut rapidement donner une estimation du temps que prendra la résolution.
- Autre conséquence, le comportement pratique est fortement déterminé par la qualité de l'arithmétique sous-jacente, qui est d'autant plus difficile à estimer que les objets manipulés vivent dans des structures non triviales : ce sont des quotients d'anneaux de polynômes univariés sur des anneaux de séries formelles.
- La complexité d'évaluation devient déterminante pour le coût d'un calcul, au même titre que le nombre de variables ou le degré. Il apparaît que la majorité des applications que nous avons traitées peuvent se mettre sous une forme qui *s'évalue bien*, par rapport au nombre de monômes une fois développés.

– Notre algorithme est probabiliste, mais sa probabilité est faible dans la plage des problèmes étudiés ici.

Un algorithme "modulaire". Notre méthode est basée sur une réduction suivie d'une remontée utilisant l'opérateur de Newton-Hensel, ce qui permet d'obtenir une complexité polynomiale (quasiment linéaire) en la taille de la sortie. Une grande partie du travail algorithmique a été présentée dans le chapitre 6, il reste à mettre le tout en forme, en particulier à contrôler les probabilités d'erreur de divers choix qui seront faits au cours du calcul.

Comme d'autres algorithmes basés sur un processus d'approximation de Newton-Hensel, cette méthode trouve un équivalent dans une utilisation du théorème des restes chinois. Dans cette optique, on calculera la résolution au-dessus de suffisamment de valeurs des paramètres ; les coefficients qui apparaissent dans ces résolutions sont les valeurs de fractions rationnelles qu'il faudra interpoler.

Autres solutions. L'article [100] adopte une démarche similaire à la nôtre, en se plaçant sous des hypothèses plus fortes de position générique, difficiles à assurer en pratique. La sortie est représentée par un calcul d'évaluation, ce qui offre une meilleure complexité mais ne permet pas une mise en pratique immédiate.

Les techniques d'évaluation dynamique se prêtent bien au calcul dépendant de paramètres [61, 62, 86, 57, 58], et fournissent une description exhaustive des possibilités de spécialisation. La thèse de S. Dellière [57] aborde notamment le sujet de la triangularisation des systèmes algébriques dans ce contexte. Les calculs de "*comprehensive Gröbner bases*" de Weispfenning se placent dans une démarche analogue. Récemment, Grigoriev et Vorobjov proposent un algorithme de calcul d'une base de Gröbner paramétrée, dont la complexité est, avec nos notations, $d^{O(mn^2)}$. Nous avons fait le choix d'une représentation valable génériquement, ce qui permet un calcul plus facile, dont le coût est fonction directe de la taille de la sortie.

Citons également les algorithmes d'élimination plus classiques, basés sur des calculs de bases de Gröbner [35, 69, 70] ou d'ensembles triangulaires [209, 137, 111, 205, 153, 11]. Dans ce cadre, leur application directe ne permet pas de distinguer les paramètres des variables, donc provoque la résolution des situations dégénérées, que l'on a cherché à éviter.

Enfin, outre toutes ces solutions, les algorithmes de résolution en dimension zéro [84, 185, 66], effectués sur le corps des paramètres, résolvent notre problème. Les complexités de toutes ces approches, en nombre d'opérations de k, ne sont pas parfaitement connues.

Chapitre 8

Estimations de degré pour les systèmes à paramètres

Résumé

Nous nous plaçons sous les hypothèses du chapitre 7. Nous donnons dans ce chapitre des bornes d'origine géométrique sur le degré en $(p_1, \ldots, p_m)$ d'objets éliminants liés aux zéros simples génériques du système $\mathbf{f}$, polynôme éliminant puis représentation de Kronecker des zéros. Nous en déduisons une borne sur le degré d'une hypersurface discriminante de l'espace des paramètres, contenant le lieu où cette représentation ne se spécialise pas.

Ces résultats sont dans la lignée de l'article de Sabia et Solernó [190], où l'on trouve une borne sur le degré d'un polynôme éliminant, sous l'hypothèse plus forte d'une variété en position de Noether par rapport à l'espace des paramètres.

À notre connaissance, l'étude pour cette situation plus générale est nouvelle. Les bornes obtenues ici permettront en particulier de contrôler la probabilité d'erreur des algorithmes de deux chapitres suivants.

Sommaire

Hypothèses et résultats

Dans ce chapitre, nous étudions les conséquences des hypothèses du chapitre 7, dont voici l'essentiel :

Notations 2 *Nous notons $\mathcal{U}$ la réunion des composantes $\mathcal{V}(f_1,\ldots,f_n)$ sur laquelle le déterminant jacobien de $\mathbf{f}$ par rapport aux variables $\mathbf{x}$ n'est pas identiquement nul et $\mathcal{V}_{sing}$ la réunion des autres composantes.*

Remarque Pour cette étude géométrique, nous n'avons pas besoin de supposer que $\mathcal{U}$ est l'*ensemble* des composantes où le déterminant jacobien n'est pas identiquement nul. Le raisonnement et les résultats sont les mêmes si l'on travaille avec une, plusieurs ou toutes les composantes définies sur k qui satisfont au critère jacobien, et la lecture de ce chapitre peut se faire dans cette optique plus générale.

Sous ces hypothèses, nous définissons les polynômes *éliminant* et *minimal* dans $\mathcal{U}$ d'une expression polynomiale en les variables $(x_1,\ldots,x_n)$. Ces polynômes ont des coefficients dans respectivement $k[p_1,\ldots,p_m]$ et $k(p_1,\ldots,p_m)$, et nous bornons leur degré en les paramètres $(p_1,\ldots,p_m)$. Comme nous l'avons annoncé dans le chapitre précédent, nous prenons le parti de la simplicité : toutes les bornes de Bézout qui suivent peuvent s'écrire en fonction d'un produit de degrés, plutôt qu'en fonction d'une puissance d'un majorant de ces degrés. Nous estimons que le gain en clarté des formules l'emporte sur la perte d'efficacité.

Selon des techniques remontant à Macaulay [150], également utilisées dans [179, 9], nous déduisons de ces résultats une majoration sur les degrés en $(p_1,\ldots,p_m)$ pouvant apparaître dans les coefficients d'une résolution de Kronecker générique. Il en découle une majoration du degré d'une hypersurface de l'espace des paramètres contenant le lieu dégénéré où cette résolution ne se spécialise pas. Ces objets sont mesurés par des degrés géométriques, eux-mêmes dominés par des bornes de Bézout.

L'article de Sabia et Solernó [190] est à la base ce cette étude. Nous avons repris ces idées, sous des hypothèses plus faibles : la différence fondamentale est que $\mathcal{U}$ n'est plus supposée être en position de Noether, mais n'est qu'un revêtement "génériquement fini" de l'espace des paramètres. Ces hypothèses plus faibles provoquent en particulier l'apparition d'un terme de tête appartenant à $k(p_1,\ldots,p_m)$ dans les polynômes éliminants et les résolutions que l'on construit. Si l'on rend ces polynômes éliminants unitaires, il faut désormais travailler dans $k(p_1,\ldots,p_m)$; à ceci près, nos résultats sont du même ordre.

Notations Nous noterons π la projection canonique :

$$\begin{array}{rccc}\pi : & \mathcal{V}(f_1,\ldots,f_n) & \to & \mathbb{A}^m(\overline{k}) \\ & (p_1,\ldots,p_m,x_1,\ldots,x_n) & \mapsto & (p_1,\ldots,p_m).\end{array}$$

8.1 Préliminaires

8.1.1 La variété privilégiée $\mathcal{U}$

Écrivons une décomposition primaire irredondante dans $\overline{k}[p_1,\ldots,p_m,x_1,\ldots,x_n]$ de la forme

$$(f_1,\ldots,f_n) = \cap Q_i.$$

Nous notons $(Q_1,\ldots,Q_k)$ les composantes sur lesquelles le déterminant jacobien de $\mathbf{f}$ par rapport aux variables $\mathbf{x}$ est non identiquement nul et B le quotient

$$B := \overline{k}(p_1,\ldots,p_m)[x_1,\ldots,x_n]/(Q_1 \cap \cdots \cap Q_k).$$

La réunion des lieux des zéros des Q_i forme le lieu $\mathcal{U}$ défini plus tôt. L'intersection $Q_1 \cap \cdots \cap Q_k$ est définie sur k, et si l'on a décidé que $\mathcal{U}$ n'était qu'une partie de l'ensemble des zéros simples, il suffit de n'en retenir que la partie correspondante, qui est définie sur k.
Chaque Q_i est premier, son image par π est dense dans $\mathbb{A}^m(\overline{k})$, et la restriction de π est y génériquement finie (c'est par exemple une conséquence du lemme de Lazard, donné dans [34, 161]).
Les composantes Q_i restent premières dans $\overline{k}(p_1,\ldots,p_m)[x_1,\ldots,x_n]$; chaque quotient

$$\overline{k}(p_1,\ldots,p_m)[x_1,\ldots,x_n]/Q_i$$

est intègre et de dimension finie comme $\overline{k}(p_1,\ldots,p_m)$-espace vectoriel, c'est donc un corps, et par le théorème des restes chinois, B est le produit de ces corps. En outre, le déterminant jacobien de $(f_1,\ldots,f_n)$ par rapport aux variables $(x_1,\ldots,x_n)$ est inversible dans chacun de ces corps, donc toutes ces extensions sont *séparables*, d'après [65, Corollary 16.16].
Enfin, d'après [98, Proposition 1], il existe un ouvert de $\mathbb{A}^m(\overline{k})$ au-dessus duquel le nombre de points dans la fibre de π restreinte à $\mathcal{U}$ est fini et égal à la somme des degrés de ces extensions sur $\overline{k}(p_1,\ldots,p_m)$. Pour les autres valeurs des paramètres, le nombre de points est infini ou inférieur.

Nous résumons une partie de ces conclusions: *Il* existe un fermé algébrique de $\mathbb{A}^m(\overline{k})$, tel que pour toute valeur $\mathbf{p}$ des paramètres en-dehors de ce fermé, le système $\mathbf{f}(\mathbf{p},.)$ admet un nombre fini non nul de solutions appartenant à la variété $\mathcal{U}$, ces solutions étant des points simples du système $(f_1,\ldots,f_n)$.

8.1.2 Lemmes complémentaires

Lemme 4 (Évitement des autres composantes) *Il existe une hypersurface de $\mathbb{A}^m(\overline{k})$ de degré inférieur à $nd_{\mathbf{p},\mathbf{x}}\mathcal{D}_{\mathbf{p},\mathbf{x}}$, telle qu'au-dessus de tout point $\mathbf{p}$ extérieur à cette hypersurface, $\mathcal{U}$ ne recontre pas la variété $\mathcal{V}_{sing}$ définie dans la précédente.*

Démonstration. Le déterminant jacobien $\mathbf{jac}(\mathbf{f})$ de $\mathbf{f}$ par rapport aux variables $\mathbf{x}$ est un polynôme de degré inférieur à $nd_{\mathbf{p},\mathbf{x}}$. L'intersection $\mathcal{U} \cap \mathcal{V}_{\mathrm{sing}}$ est incluse dans $\mathcal{U} \cap \mathcal{V}(\mathbf{jac}(\mathbf{f}))$, qui est de degré inférieur à $nd_{\mathbf{p},\mathbf{x}}\mathcal{D}_{\mathbf{p},\mathbf{x}}$. Son image par π n'est pas dense dans $\mathbb{A}^m(\overline{k})$ et a un degré plus petit. □

Lemme 5 (Image des fibres non finies) *Soient $\mathcal{P}$ une variété irréductible de dimension r incluse dans $\mathbb{A}^n(\overline{k})$ et φ la projection de $\mathcal{P}$ sur les r premières coordonnées, que l'on suppose dominante. Alors l'image des fibres non finies de φ est de codimension au moins 2.*

Démonstration. (d'après [164, §3B]) Soit $\mathcal{W}$ l'adhérence dans $\mathcal{P}$ de l'ensemble des fibres infinies. C'est une sous-variété de dimension au plus $r-1$ de $\mathcal{P}$. Les fibres de φ restreinte à $\mathcal{W}$ ont dimension au moins 1, donc l'image de $\mathcal{W}$ par φ a pour dimension au plus $m-2$. □

On en déduit que l'image des fibres non finies de π restreinte à $\mathcal{U}$ est également de codimension au moins 2.

8.2 Degré d'un polynôme éliminant

Dans cette section, nous introduisons le polynôme éliminant d'une expression en les variables $(x_1,\ldots,x_n)$. Nous insistons sur ses propriétés de degré, en particulier dans le cas d'une combinaison linéaire des variables à coefficients génériques. Ces résultats sont à la base des majorations de la section suivante.

8.2.1 Polynômes éliminant et minimal

Nous donnons pour commencer une borne sur le degré d'un *polynôme éliminant* d'un polynôme en les variables de la fibre. C'est une continuation de [190, Proposition 1], qui traite du cas où la variété $\mathcal{U}$ est en position de Noether.

Proposition 27 (Polynôme éliminant et polynôme minimal) *Soit g un polynôme l'anneau $k[x_1,\dots,x_n]$. Alors il existe un polynôme $M_g \in k[p_1,\dots,p_m][t]$ de degré total inférieur à $\mathcal{D}_{\mathbf{p},\mathbf{x}} \deg g$, de degré en t inférieur à $\mathcal{D}_{\mathbf{x}}$, de contenu 1, premier avec sa dérivée par rapport à t, et tel que $M_g(g)$ s'annule sur $\mathcal{U}$. On peut choisir le polynôme M_g de sorte que rendu unitaire en t, il coïncide avec le polynôme minimal m_g de la multiplication par g dans B.*

Démonstration. La preuve est une adaptation de celle de [190]. Considérons le morphisme ϕ :

$$\begin{array}{rccl} \phi: & \mathcal{U} & \to & \mathbb{A}^{m+1}(\overline{k}) \\ & (p_1,\dots,p_m,x_1,\dots,x_n) & \mapsto & (p_1,\dots,p_m,g(x_1,\dots,x_n)). \end{array}$$

Notons $\mathcal{T}$ l'image de ϕ, dont l'adhérence est une hypersurface de $\mathbb{A}^{m+1}(\overline{k})$. Nous commençons par borner le degré de $\mathcal{T}$; nous montrons ensuite qu'un polynôme sans carré définissant $\mathcal{T}$ est séparable, et qu'il coïncide avec le numérateur du polynôme minimal de la multiplication par g dans B — ce qui montrera que ses coefficients sont dans k, et non dans $\overline{k}$.

Soit $\mathcal{H}$ une droite générique de $\mathbb{A}^{m+1}(\overline{k})$ qui rencontre $\mathcal{T}$ en un nombre fini de points. Cette droite est l'intersection de m hyperplans $h_1(\mathbf{p},t) = \cdots = h_m(\mathbf{p},t) = 0$ de $\mathbb{A}^{m+1}(\overline{k})$ et peut donc se redéfinir par combinaisons linéaires sous la forme $h'_1(\mathbf{p}) = \cdots = h'_{m-1}(\mathbf{p}) = h'_m(\mathbf{p},t) = 0$. Sa projection sur $\mathbb{A}^m(\overline{k})$ définie par $h'_1(\mathbf{p}) = \cdots = h'_{m-1}(\mathbf{p}) = 0$ ne rencontre pas la projection par π des fibres infinies de $\mathcal{U}$, puisque celles-ci sont enfermées dans un espace de codimension au moins 2. Les antécédents par ϕ des points de $\mathcal{H} \cap \mathcal{T}$ sont les points de $\mathcal{U}$ vérifiant $h'_1(\mathbf{p}) = \cdots = h'_{m-1}(\mathbf{p}) = h'_m(\mathbf{p},g(\mathbf{x})) = 0$ et sont ainsi en nombre fini, inférieur à $\mathcal{D}_{\mathbf{p},\mathbf{x}} \deg g$ d'après l'inégalité de Bézout. Ainsi, $\deg \mathcal{T} \leq \mathcal{D}_{\mathbf{p},\mathbf{x}} \deg g$.

Soit $M_g \in \overline{k}[p_1,\dots,p_m,t]$ un polynôme sans carré définissant $\mathcal{T}$. Si l'on voit M_g comme un polynôme dans $\overline{k}[p_1,\dots,p_m][t]$, alors celui-ci a contenu 1, d'après le lemme 5. Chaque composante irréductible des zéros de M_g est l'adhérence de l'image par ϕ d'au moins une composante irréductible de $\mathcal{U}$. Comme l'extension

$$\overline{k}(p_1,\dots,p_m) \to B$$

est un produit d'extensions séparables, l'extension

$$\overline{k}(p_1,\dots,p_m) \to \overline{k}(p_1,\dots,p_m)[t]/M_g$$

l'est alors également. D'après [98, Proposition 1], le degré en t de M_g est donc égal au nombre générique de valeurs que prend la fonction g dans les fibres de π, donc inférieur à $\mathcal{D}_{\mathbf{x}}$.

Enfin, considérons le polynôme minimal m_g de la multiplication par la classe de g dans B.
Le polynôme $M_g(g)$ est nul sur $\mathcal{U}$; puisque les idéaux Q_i définis précédemment sont premiers, $M_g(g)$ appartient à leur intersection. En conséquence, le polynôme m_g divise M_g dans $\overline{k}(p_1,\dots,p_m)[t]$. Réciproquement, le numérateur du polynôme minimal m_g évalué en g est nul sur les points de $\mathcal{U}$, donc m_g est un multiple de M_g dans $\overline{k}[p_1,\dots,p_m,t]$. Les polynômes m_g et M_g ont ainsi même degré en t, et M_g est égal au numérateur de m_g, à un facteur constant près. L'idéal définissant le quotient B est engendré par des polynômes définis sur k, donc m_g a ses coefficients dans $k(p_1,\dots,p_m)$, et c'est également le cas de M_g. □

Définition 11 (Polynôme éliminant et polynôme minimal) *Soit g dans $k[x_1,\ldots,x_n]$.*

- *Nous appellerons polynôme éliminant de g et noterons M_g le polynôme de $k[p_1,\ldots,p_m][t]$ défini ci-dessus et déjà noté M_g.*
- *Nous appellerons polynôme minimal de g et noterons m_g le polynôme unitaire de $k(p_1,\ldots,p_m)[t]$ déjà noté m_g ci-dessus.*

Remarque En particulier, si $\sum u_i x_i$ est une combinaison k-linéaire séparant génériquement les points de $\mathcal{U}$, son polynôme minimal est égal à son polynôme caractéristique dans B ; il est de degré $\mathcal{D}_{\mathbf{x}}$, et a ses racines distinctes. L'élément $\sum u_i x_i$ est un élément primitif pour l'extension $\overline{k}(p_1,\ldots,p_m) \to B$.

8.2.2 Application à une forme linéaire générique

Dans la suite, nous aurons besoin de manipuler le polynôme éliminant d'une forme linéaire générique. Ce polynôme codera la description des variables $\mathbf{x}$ en fonction des paramètres $\mathbf{p}$.

Définition 12 (Forme linéaire générique) *Étendons k par n indéterminées $(t_1,\ldots,t_n)$; nous appellerons forme linéaire générique t_{gen} la forme linéaire $\sum t_i x_i$.*

Nous appellerons alors polynômes éliminant et minimal *génériques* les polynômes définis ci-dessus, pour l'élément t_{gen}. Le polynôme éliminant générique n'est pour l'instant défini qu'à une constante de $k(t_1,\ldots,t_n)$ près. La proposition précédente s'applique sur le corps $k(t_1,\ldots,t_n)$; les degrés des composantes de $\mathcal{V}(f_1,\ldots,f_n)$ ne croissent pas par cette extension du corps de base. Ainsi, le polynôme éliminant générique est dans

$$k(t_1,\ldots,t_n)[p_1,\ldots,p_m][t];$$

il a un degré total en $(p_1,\ldots,p_m)$ au plus $\mathcal{D}_{\mathbf{p},\mathbf{x}}$, son degré en t est $\mathcal{D}_{\mathbf{x}}$, et le pgcd de ce polynôme avec sa dérivée par rapport à t est 1.

Lemme 6 *Il existe un multiple du polynôme éliminant générique qui est un polynôme de*

$$k[t_1,\ldots,t_n,p_1,\ldots,p_m][t],$$

homogène de degré $\mathcal{D}_{\mathbf{x}}$ en $(t,t_1,\ldots,t_n)$.

Démonstration. L'endomorphisme $\mathcal{M}(t)$ de multiplication par t dans l'extention de B sur le corps $\overline{k}(t_1,\ldots,t_n,p_1,\ldots,p_m)$ est la combinaison linéaire $\sum t_i \mathcal{M}(x_i)$. La structure du quotient B sur le corps $\overline{k}(t_1,\ldots,t_n,p_1,\ldots,p_m)$ est la même que sur le corps $\overline{k}(p_1,\ldots,p_m)$, aussi, dans la base des x_i, les matrices de multiplication par les variables x_i ont pour seuls dénominateurs des polynômes en $(p_1,\ldots,p_m)$. En conséquence, le polynôme caractéristique de $\mathcal{M}(t)$ a pour dénominateur une fraction en $(p_1,\ldots,p_m)$, et est homogène de degré $\mathcal{D}_{\mathbf{x}}$ en $(t,t_1,\ldots,t_n)$. En prenant le numérateur de ce polynôme caractéristique, on obtient le polynôme éliminant recherché. □

Remarque La construction du polynôme éliminant d'une forme linéaire générique n'est pas une idée nouvelle. On en trouve trace chez Macaulay [150], qui le baptise u-résultant. Dans le milieu du calcul formel, des constructions explicites du u-résultant sont proposées dans [104, 39, 41, 9, 66], entre autres références.

8.2.3 Un autre point de vue sur le polynôme éliminant générique

Mettre les lettres t_i dans le corps de base a permis de borner facilement le degré en $(p_1,\dots,p_m)$ du polynôme éliminant générique. Il est aussi utile de considérer les t_i comme éléments d'un anneau de polynômes : l'intérêt se révèlera notamment quand des considérations de hauteur rentreront en jeu, dans la partie suivante.

Notons $\widetilde{\mathcal{U}}$ la sous-variété de $\mathbb{A}^{n+1}(\overline{k}) \times \mathbb{A}^{n+m}(\overline{k})$:

$$\widetilde{\mathcal{U}} = \{(t,t_1,\dots,t_n,p_1,\dots,p_m,x_1,\dots,x_n) \text{ t.q. } (p_1,\dots,p_m,x_1,\dots,x_n) \in \mathcal{U}, \text{ et } t = \sum t_i x_i\}.$$

Soit ensuite φ l'application

$$\begin{array}{cccc} \varphi : & \widetilde{\mathcal{U}} & \to & \mathbb{A}^{n+m+1}(\overline{k}) \\ & (t,t_1,\dots,t_n,p_1,\dots,p_m,x_1,\dots,x_n) & \mapsto & (t,t_1,\dots,t_n,p_1,\dots,p_m). \end{array}$$

Les composantes irréductibles de l'image de φ sont les images des composantes irréductibles de $\widetilde{\mathcal{U}}$; celles-ci ont dimension $m+n$, et les fibres de φ ont toutes génériquement dimension zéro. L'image de φ est donc une hypersurface de $\mathbb{A}^{n+m+1}(\overline{k})$. Notons M_{Gen} un polynôme sans carré définissant de cette hypersurface. Dans l'anneau $k(t_1,\dots,t_n)[p_1,\dots,p_m,t]$, M_{Gen} est un multiple du polynôme éliminant générique défini dans le lemme ci-dessus par une constante, c'est-à-dire un polynôme de $k[t_1,\dots,t_n]$. Comme le polynôme M_{Gen} ne peut avoir de facteur dans $k[t_1,\dots,t_n]$, il est homogène de degré $\mathcal{D}_{\mathbf{x}}$ en $(t,t_1,\dots,t_n)$. Enfin, on a vu dans la proposition précédente que le degré en $(p_1,\dots,p_m)$ d'un polynôme éliminant d'une forme linéaire générique est au plus $\mathcal{D}_{\mathbf{p},\mathbf{x}}$.

Résumons toutes ces propriétés :
Polynôme éliminant générique Désormais, nous choisirons comme polynôme éliminant d'une forme linéaire générique le polynôme M_{Gen} défini ci-dessus. Il est homogène de degré $\mathcal{D}_{\mathbf{x}}$ en les variables $(t,t_1,\dots,t_n)$, et a degré au plus $\mathcal{D}_{\mathbf{p},\mathbf{x}}$ en les variables $(p_1,\dots,p_m)$; enfin, il est premier avec sa dérivée par rapport à t.

Spécialisation du polynôme éliminant générique. Soit $t = u_i x_i$ une combinaison des variables à coefficients dans k qui sépare génériquement les points de $\mathcal{U}$. Effectuons la spécialisation des t_i en u_i dans le polynôme éliminant générique ci-dessus. On obtient alors le polynôme éliminant de t, *éventuellement multiplié par un facteur dans* $k(p_1,\dots,p_m)$.
Le polynôme minimal de t coïncide avec son polynôme caractéristique dans le quotient B. Celui-ci est bien la spécialisation du polynôme caractéristique de la forme générique, puisque le calcul du polynôme caractéristique commute avec la spécialisation des t_i en u_i. On passe du polynôme minimal au polynôme éliminant en chassant les dénominateurs, et il est possible que des simplifications se soient effectuées dans le polynôme caractéristique "spécialisé", qui expliquent l'apparition possible de parasites.

Exemple [Apparition de parasites] Considérons dans $\mathbb{Q}[p_1,p_2,x_1,x_2]$ le système

$$\begin{array}{rcl} f_1 & = & x_2 - p_1 \\ f_2 & = & x_2 + p_2 x_1. \end{array}$$

Alors le polynôme $M_{\mathrm{Gen}}(t,t_1,t_2,p_1,p_2)$ est

$$p_2 t - p_1 p_2 t_2 + p_1 t_1.$$

Si on spécialise (t_1,t_2) en $(0,1)$, on obtient le polynôme minimal de x_2 à un facteur près :

$$M_{\text{Gen}}(t,t_1,t_2,0,1) = p_2 t - p_1 p_2,$$

alors que le polynôme éliminant de x_2 est $t - p_1$. L'apparition de ce facteur est due au fait qu'à p_1 fixé, quand p_2 tend vers 0, la variable x_2 reste bornée, alors que toute autre combinaison des variables tend vers l'infini.

8.3 Degrés d'une représentation univariée

Le lemme suivant est la base des bornes en degré des résolutions génériques :

Lemme 7 *Soit $t = \sum u_i x_i$ un élément séparant génériquement les points de $\mathcal{U}$, à coefficients dans k. Alors les paramétrisations des variables x_i dans $\mathcal{U}$ sont données par la spécialisation des (t_i) en (u_i) de :*

$$\frac{\partial M_{Gen}}{\partial t} x_i \quad = \quad -\frac{\partial M_{Gen}}{\partial t_i}.$$

En particulier, ces polynômes appartiennent à l'idéal $\mathcal{I}(\mathcal{U})$.

Démonstration. Le polynôme $M_{\text{Gen}}(p_1,\ldots,p_m,t_1,\ldots,t_n,\sum t_i x_i)$ s'annule sur la variété décrite par l'idéal $\mathcal{I}(\mathcal{U})$ pris dans l'anneau $k[t_1,\ldots,t_n,p_1,\ldots,p_m,x_1,\ldots,x_n]$. Il en est de même pour sa dérivée partielle par rapport à t_i ; la spécialisation des t_i en u_i fournit la paramétrisation recherchée. □

Le résultat de séparabilité obtenu précédemment montre que pour une valeur générique des paramètres, $\frac{\partial M_{\text{Gen}}}{\partial t}$ est inversible modulo M_{Gen}, la paramétrisation précédente a donc bien un sens. En particulier, en toute valeur des paramètres où la spécialisation de M_{Gen} est sans racine multiple, la spécialisation de ces paramétrisations décrit la "trace" des points de $\mathcal{U}$.

De l'argument précédent nous déduisons l'existence d'une représentation univariée des zéros de l'idéal $\mathcal{I}(\mathcal{U})$ étendu dans $k(p_1,\ldots,p_m)[x_1,\ldots,x_n]$, pour laquelle on dispose d'une borne intrinsèque en degré.

Théorème 8 *Soit t un élément génériquement séparant, de la forme $\sum u_i x_i$, où les u_i sont dans k. La résolution $\mathcal{R}(t)$*

$$q(t) = 0, \quad \left\{ \begin{array}{rcl} \frac{dq(t)}{dt} x_1 & = & v_1(t), \\ & \vdots & \\ \frac{dq(t)}{dt} x_n & = & v_n(t), \end{array} \right.$$

satisfait les propriétés suivantes : les polynômes q et v_i sont dans $k(p_1,\ldots,p_m)[t]$, le polynôme q est unitaire de degré $\mathcal{D}_{\mathbf{x}}$, les polynômes v_i sont de degré strictement inférieur à $\mathcal{D}_{\mathbf{x}}$ et les numérateurs et le ppcm des dénominateurs des coefficients de ces polynômes sont de degré en $(p_1,\ldots,p_m)$ inférieur à $\mathcal{D}_{\mathbf{p},\mathbf{x}}$.

Autrement dit, le degré $\mathcal{D}(t)$ est inférieur à $\mathcal{D}_{\mathbf{p},\mathbf{x}}$ dans une représentation de Kronecker.

Comme nous l'avons dit plus haut, il est possible que la spécialisation du polynôme éliminant générique en les coefficients u_i fasse apparaître un facteur en $(p_1,\ldots,p_m)$. Ce facteur disparaît du polynôme éliminant si on le rend unitaire, pas nécessairement des paramétrisations.

Exemple Reprenons l'exemple donné dans la section 8.2.2 : $(f_1,f_2) = (x_2 - p_1, x_2 + p_2 x_1)$. La variable x_2 est élément séparant, et la représentation est :

$$\begin{array}{rcl} x_1 & = & -\frac{1}{p_2} x_2 \\ x_2 & = & p_1. \end{array}$$

Le résultat ci-dessus est vrai en caractéristique positive. Il faut cependant prendre des précautions en travaillant avec des polynômes définis sur un corps fini : si k est un corps fini de cardinal trop petit, il n'existe pas nécessairement de forme linéaire $\sum u_i x_i$ à coefficients dans k séparant les points de $\mathcal{U}$.
Les éléments non séparants annulent le discriminant du polynôme éliminant générique. C'est un polynôme non nul de $k[p_1,\ldots,p_m][t_1,\ldots,t_n]$, dont le degré en $(t_1,\ldots,t_n)$ est inférieur à $2d_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^n$. En conséquence, si le cardinal de k est supérieur à $2d_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^n$, nous sommes sûrs qu'il existe un élément séparant à coefficients dans k, par le lemme de Zippel-Schwartz 3 du chapitre 5.

En renforçant les hypothèses. En imposant des hypothèses géométriques plus fortes, on peut raffiner le résultat. Nous retrouvons ainsi les conclusions de [190, Proposition 1] et [160].

Corollaire 6 (Position de Noether) *Si la variété $\mathcal{U}$ est en position de Noether avec pour variables libres $(p_1,\ldots,p_m)$, alors le polynôme $M_g(t)$ défini dans la proposition 27 est unitaire en t. Ainsi les coefficients des polynômes intervenant dans une représentation de Kronecker générique sont des polynômes en $(p_1,\ldots,p_m)$.*

Corollaire 7 (Position de Noether projective) *Si en outre il existe une valeur des paramètres $\mathbf{p}_0$ ayant $\mathcal{D}_{\mathbf{p},\mathbf{x}}$ antécédents par $\pi|_{\mathcal{U}}$, nous dirons que la variété $\mathcal{U}$ est en position de Noether projective, avec pour variables libres $(p_1,\ldots,p_m)$. Dans ce cas, tous les polynômes intervenant dans une représentation de Kronecker générique sont de degré total inférieur à $\mathcal{D}_{\mathbf{p},\mathbf{x}}$.*

8.4 Quantification des fibres non génériques

Nous déduisons des résultats précédents une quantification de certaines dégénérescences géométriques des fibres de la projection π. Les bornes données ici s'énoncent naturellement en termes de degrés géométriques. Elles serviront à borner la probabilité d'erreur des algorithmes du chapitre 10, aussi les donne-t-on en fonction de données syntaxiques, c'est-à-dire les degrés des générateurs dont on dispose.

Lemme 8 (Fibres infinies sur $\mathcal{U}$) *Soit $\mathbf{p}$ une valeur des paramètres qui n'annule pas le coefficient dominant du polynôme M_{Gen}. Alors la fibre de π restreinte à $\mathcal{U}$ au-dessus de $\mathbf{p}$ est finie.*

Démonstration. Presque toute combinaison linéaire t des x_i à coefficients dans $\overline{k}$ est séparante. Le coefficient dominant du polynôme éliminant M_t est un diviseur du coefficient dominant de M_{Gen}, donc est non nul en $\mathbf{p}$. Ainsi, t ne peut prendre qu'un nombre fini de valeurs au-dessus de $\mathbf{p}$. □

Nous définissons maintenant une hypersurface discriminante, en-dehors de laquelle il est possible de spécialiser une résolution générique. Cette hypersurface est construite à partir d'un élément séparant t ; avec le vocabulaire du chapitre précédent, elle caractérise les valeurs des paramètres satisfaisant les *conditions de généricité* pour t.

Proposition 28 (Spécialisation de la résolution générique) *Soit $\sum u_i x_i$ un élément séparant génériquement les points de $\mathcal{U}$. Il existe un polynôme $\mathcal{D}isc_t$ tel que toute valeur des paramètres n'annulant pas $\mathcal{D}isc_t$ satisfait toutes les hypothèses de généricité pour t. Le polynôme $\mathcal{D}isc_t$ a un degré inférieur à*

$$d_{\mathbf{p},\mathbf{x}}^n(2d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}}).$$

Démonstration. Soient $\mathcal{D}\mathrm{isc}_0$ l'équation d'une hypersurface de degré inférieur à $nd_{\mathbf{p},\mathbf{x}}d^n_{\mathbf{p},\mathbf{x}}$ contenant la projection de $\mathcal{U}\cap\mathcal{V}_{\mathrm{sing}}$ sur l'espace des paramètres, et $\mathcal{D}\mathrm{isc}_1$ le résultant de M_{Gen} et de sa dérivée par rapport à t, spécialisé en les u_i. Celui-ci a degré au plus $2d^n_{\mathbf{x}}d^n_{\mathbf{p},\mathbf{x}}$ d'après la proposition 2 du chapitre 3.
En dehors du lieu des zéros de $\mathcal{D}\mathrm{isc}_1$, la dérivée du polynôme minimal de q est inversible par rapport à q et q a degré plein, et aucun des dénominateurs n'apparaissant dans la résolution générique ne s'annule : la spécialisation de $\mathcal{R}$ décrit donc les $\mathcal{D}_{\mathbf{x}}$ points de $\mathcal{U}$. En-dehors du lieu des zéros de $\mathcal{D}\mathrm{isc}_0$, ces points ne rencontrent pas le lieu singulier $\mathcal{V}_{\mathrm{sing}}$. Le polynôme $\mathcal{D}\mathrm{isc}_t = \mathcal{D}\mathrm{isc}_0\mathcal{D}\mathrm{isc}_1$ satisfait les conditions de l'énoncé. □

On voudra également spécialiser le polynôme minimal générique m_{x_i} de la variable x_i. Ce dernier corollaire trouvera son utilité dans le chapitre 18, où l'on cherchera à décrire les zéros simples génériques par des ensembles triangulaires.

Corollaire 8 (Spécialisation du polynôme minimal) *Soit t un élément séparant génériquement les points de $\mathcal{U}$, et $\mathcal{D}isc_t$ le polynôme discriminant associé à t. Pour toute variable x_i, il existe un polynôme $\mathcal{D}isc_{x_i}$ tel qu'en toute valeur des paramètres $\mathbf{p}$ n'annulant pas $\mathcal{D}isc_t\mathcal{D}isc_{x_i}$, le polynôme minimal de x_i parmi les zéros simples de $\mathbf{f}(\mathbf{p},.)$ est la spécialisation du polynôme minimal générique m_{x_i}.*
Le polynôme $\mathcal{D}isc_{x_i}$ peut être pris de degré inférieur à

$$2d^n_{\mathbf{p},\mathbf{x}}d^n_{\mathbf{x}}.$$

Démonstration. Soit $(t=\sum u_ix_i,q,\mathbf{w})$ une résolution des zéros simples génériques mise sous forme Shape Lemma. Le polynôme caractéristique de x_i dans la fibre générique s'obtient par un calcul de résultant entre la paramétrisation $x_i = w_i(t)$ et le polynôme minimal q ; ensuite, le polynôme minimal de x_i est obtenu comme la partie sans carrés du polynôme caractéristique.
Soit $\mathcal{D}\mathrm{isc}_{x_i}$ le résultant en t du polynôme éliminant M_{x_i} avec sa dérivée ; il a le degré annoncé. Soit $\mathbf{p}$ une valeur des paramètres n'annulant pas $\mathcal{D}\mathrm{isc}_t\mathcal{D}\mathrm{isc}_{x_i}$. La résolution univariée se spécialise en $\mathbf{p}$, ainsi que le calcul du polynôme caractéristique de x_i. La spécialisation du polynôme m_{x_i} est sans carré, et divise le polynôme caractéristique de x_i. Le degré du polynôme minimal de x_i ne peut pas augmenter à la spécialisation, d'après [98, Proposition 1], ce qui prouve notre assertion. □

Remarque Revenons pour finir sur la preuve de la proposition 28. On ne peut pas déduire que le déterminant jacobien est non nul dans la fibre uniquement du fait que le discriminant du polynôme éliminant est non nul, comme le prouve le contre-exemple suivant.

Exemple Considérons dans $\mathbb{Q}[p_1,x_1]$ le système $f_1 = (p_1-1)(x_1-1)$. La variété $\mathcal{U}$ est une droite, et le polynôme minimal de toute forme linéaire est $t-1$. À l'intersection avec la composante verticale $\mathcal{V}_{\mathrm{sing}}$ définie par $p_1 = 1$, le discriminant de t reste égal à 1. Par contre, la dérivée de f_1 s'annule.

Chapitre 9

Fonctions auxiliaires

Résumé

Ce chapitre est une transition entre la présentation géométrique du chapitre précédent et l'algorithme qui en découle dans le chapitre suivant.

Nous présentons ici les sous-routines qui seront utilisées, pour lesquelles nous donnons des résultats de complexité et de probabilité d'erreur, basés sur les estimations de degré du chapitre précédent. L'extraction de ces fonctions auxiliaires du corps du chapitre suivant permet, on l'espère, de clarifier l'exposition de l'ensemble de l'algorithme. En contrepartie, le présent chapitre se présente comme une juxtaposition de résultats techniques.

– La première section décrit un processus de reconstruction rationnelle de polynômes et de résolutions univariés, à coefficients dans un anneau de séries formelles, en se basant sur les résultats du chapitre 5.
– Dans la deuxième section, nous introduisons une fonction notée `Temoin`, qui sera à la base des tests d'arrêt de remontée de Newton. Cette fonction repose sur des calculs spécialisés, et nous discutons la validité des décisions qui en dépendent.

Sommaire

9.1 Reconstruction Rationnelle

Cette section décrit un algorithme probabiliste de reconstruction rationnelle des coefficients de polynômes et de résolutions dont les coefficients sont des séries formelles multivariées de $k[[p'_1, \ldots ,p'_m]]$ — la notation $(p'_1, \ldots ,p'_m)$ indique que l'origine des coordonnées a été déplacée. Nous en donnons la complexité et mesurons la probabilité d'erreur.

9.1.1 Description des algorithmes

Reconstruction d'une fraction rationnelle à partir de son développement de Taylor. Dans la partie 5, nous avons présenté un algorithme probabiliste de calcul de reconstruction rationnelle d'une fraction rationnelle à partir de son développement en une série multivariée s. Pour une série s à précision 2^i, nous cherchons un approximant dont le numérateur et le dénominateur sont de degrés bornés par 2^{i-1}. L'algorithme est paramétré par un changement de variables γ choisi dans k^{m-1} et est noté `ReconstructionRationnelle`(s,γ). Il renvoie une valeur booléenne indiquant si la reconstruction de s est possible et, si oui, ladite reconstruction.

Reconstruction d'un polynôme à coefficients dans les séries. Nous appliquerons la fonction `ReconstructionRationnelle` au calcul d'approximants de tous les coefficients d'un polynôme univarié f de $k[[p'_1, \ldots ,p'_m]][t]$, en gardant le nom `ReconstructionRationnelle`$(f,\gamma)$. Les fractions sont normalisées en mettant le coefficient constant des dénominateurs à 1. Enfin, si les coordonnées étaient $(p'_1, \ldots ,p'_m)$, nous revenons dans les coordonnées d'origine $(p_1, \ldots ,p_m)$ par translation.

Proposition 29 *Si f est un polynôme de degré d à coefficients dans les séries $k[[p'_1, \ldots ,p'_m]]$ à précision 2^i, en cas de succès, le coût de la reconstruction de tous ses coefficients est*

$$O_{log}(dm\mathcal{M}_s(2^i,m))$$

opérations de k.

Démonstration. L'estimation du coût de la reconstruction rationnelle et de la translation données dans le chapitre 5 rentrent toutes les deux dans la borne annoncée. □

Reconstruction d'une résolution univariée. Nous donnons maintenant une procédure de reconstruction rationnelle d'une résolution univariée $\mathcal{R} = (t = \sum u_i x_i, q(t), \mathbf{w}(t))$, à coefficients dans les séries $k[[p'_1, \ldots ,p'_m]]$. Cette procédure sera utilisée à la sortie d'une étape de remontée de Newton, la résolution sera donc sous forme “Shape Lemma”.
On commence par la mettre sous forme de Kronecker, puisque les bonnes bornes sur le degré des coefficients des paramétrisations données dans le chapitre précédent sont valables dans ce cas. On applique alors le procédé de reconstruction rationnelle à tous les coefficients de la résolution. Ensuite, si les coordonnées étaient $(p'_1, \ldots ,p'_m)$, on revient par translation aux coordonnées initiales $(p_1, \ldots ,p_m)$. Nous appellerons toujours cette procédure `ReconstructionRationnelle`$(\mathcal{R},\gamma)$.

Proposition 30 *Si on travaille à partir de séries en m variables à précision 2^i, le coût de la procédure de reconstruction d'une résolution univariée $\mathcal{R}$ en cas de succès est*

$$O_{log}(nm\mathcal{M}(\mathcal{D}_{\mathbf{x}})\mathcal{M}_s(2^i,m))$$

opérations de k.

FIG. 9.1 – *Reconstruction de polynôme univarié sur des séries formelles*

procedure `ReconstructionRationnelle`(p,γ)

\# **Entree**
\# $p \in k[[p'_1,\dots,p'_m]][t]$ à précision d
\# γ changement de variables
\# **Description**
\# Applique `ReconstructionRationnelle`(.,γ) aux coefficients de p
\# Met les coefficients constants des dénominateurs à 1
\# Passe des coordonnées $(p'_1,\dots,p'_m)$ aux coordonnées $(p_1,\dots,p_m)$
\# **Sortie**
\# b,p_r : b booléen ; si b est vrai, p_r est la reconstruction de p
end;

Démonstration. Détaillons le coût de chacune des étapes :

- D'après la proposition 3 du chapitre 1, la mise sous forme de Kronecker se fait en
$$O_{\log}(n\mathcal{M}(\mathcal{D}_{\mathbf{x}})\mathcal{M}_s(2^i,m))$$
opérations de k.
- D'après le paragraphe précédent, la reconstruction rationnelle et la translation sur tous les coefficients se font en
$$O_{\log}(n\mathcal{D}_{\mathbf{x}}m\mathcal{M}_s(2^i,m))$$
opérations de k.

Ces coûts rentrent dans l'estimation annoncée. □

9.1.2 Probabilité d'erreur

Nous donnons ici une quantification des changements de variables γ qui font échouer la reconstruction rationnelle d'une résolution. Pour cela, nous reprenons les résultats du chapitre 5. Rappelons que si t est un élément séparant génériquement les points de $\mathcal{U}$, $\mathcal{D}(t)$ désigne le degré en $(p_1,\dots,p_m)$ des numérateurs et dénominateurs des coefficients de la résolution $\mathcal{R}(t)$.

Proposition 31 *Soient t un élément séparant génériquement les points de $\mathcal{U}$, et $\mathcal{R}(t)$ la résolution de $\mathcal{U}$ avec t pour élément séparant. Soit $\mathbf{p}_1$ un point qui satisfait toutes les hypothèses de généricité pour t. Supposons donnés les développements de Taylor en $\mathbf{p}_1$ de la résolution $\mathcal{R}(t)$ à l'ordre $\mathcal{D}$, où $2\mathcal{D}(t) \leq \mathcal{D} \leq 4\mathcal{D}(t)$.*
Alors les changements de variables γ qui ne permettent pas la reconstruction de tous les coefficients de $\mathcal{R}(t)$ sont inclus dans une hypersurface de $\mathbb{A}^{m-1}(\overline{k})$ degré au plus
$$16n(4d_{\mathbf{p},\mathbf{x}}^n + 1)d_{\mathbf{p},\mathbf{x}}^{2n}d_{\mathbf{x}}^n.$$

Démonstration. D'après la proposition 81 du chapitre 5, pour chaque coefficient de la résolution, il existe un polynôme de degré $(\mathcal{D}+1)\mathcal{D}^2 \leq 16(4\mathcal{D}(t)+1)\mathcal{D}(t)^2$, dont les zéros font échouer la reconstruction. Le résultat demandé s'en déduit en prenant en compte tous les coefficients, et en majorant le degré $\mathcal{D}(t)$ par la borne de Bézout $d_{\mathbf{p},\mathbf{x}}^n$. □

FIG. 9.2 – *Reconstruction de Résolution Univariée sur des séries formelles*

procedure ReconstructionRationnelle($\mathcal{R}$,γ)

\# **Entree**
\# $\mathcal{R} = (t,q,\mathbf{w})$ résolution à coefficients dans $k[[p'_1,\ldots,p'_m]]$ à précision d
\# **Description**
$\mathcal{R} \leftarrow$ KroneckerParametrisation($\mathcal{R}$);
$q \leftarrow$ ReconstructionRationnelle(q,γ);
$v_i \leftarrow$ ReconstructionRationnelle(v_i,γ) : $i = [1\ldots n]$;
\# Passe des coordonnées $(p'_1,\ldots,p'_m)$ aux coordonnées $(p_1,\ldots,p_m)$
\# **Sortie**
\# b,$\mathcal{R}_r$: b booléen ; si b est vrai, $\mathcal{R}_r$ est la reconstruction de $\mathcal{R}$
\# à coefficients dans $k(p_1,\ldots,p_m)$
end;

9.2 Critère d'arrêt pour une résolution univariée

Soit $\mathcal{R}'(t) = (t = \sum u_i x_i, q'(t), \mathbf{v}'(t))$ une résolution univariée sous forme de Kronecker, à coefficients dans le corps $k(p_1,\ldots,p_m)$ et $\mathcal{J}$ l'idéal $(p_1 - p_{1,1},\ldots,p_m - p_{1,m})$, pour une valeur $(p_{1,1},\ldots,p_{1,m})$ de k^m.
Supposons que les dénominateurs de $\mathcal{R}'(t)$ ne s'annulent pas en $\mathcal{J}$. Nous présentons dans cette section une fonction `Temoin`, qui évalue la spécialisation en $\mathcal{J}$ de $\mathcal{R}'(t)$ sur le système $\mathbf{f}$ et renvoie une valeur booléenne indiquant si cette évaluation est zéro ou non. Cette fonction servira ultérieurement dans le contexte plus précis où $k = \mathbb{Q}$, sa présentation générique permettra d'éviter de la réécrire en intégralité.

9.2.1 Présentation générale

Le test consiste en l'évaluation du système $\mathbf{f}$ sur la résolution $\mathcal{R}'(t)$ modulo l'idéal $\mathcal{J}$. On teste également la nullité du déterminant jacobien **jac** de $\mathbf{f}$ par rapport à $\mathbf{x}$ sur les points décrits par $\mathcal{R}'(t)$, pour des raisons qui seront détaillées dans la prochaine sous-section.

Proposition 32 (Coût de la comparaison) *Notons $b(\mathcal{R}'(t),\mathcal{J})$ le coût de la réduction d'un coefficient de $\mathcal{R}'(t)$ modulo $\mathcal{J}$, et $a(\mathcal{J})$ le coût des opérations dans le quotient $k[p_1,\ldots,p_m]/\mathcal{J}$. Alors le coût de la réduction et du test est*

$$O_{log}\big(\mathcal{M}(\mathcal{D}_{\mathbf{x}})(nb(\mathcal{R}'(t),\mathcal{J}) + (n^{\Omega} + nL)a(\mathcal{J}))\big)$$

opérations.

Démonstration. La réduction se fait en traitant les $n\mathcal{D}_{\mathbf{x}}$ coefficients de la résolution. L'évaluation nécessite d'abord de mettre la résolution sous forme "Shape Lemma", en $O(n\mathcal{M}(\mathcal{D}_{\mathbf{x}})\log \mathcal{D}_{\mathbf{x}})$ opérations du quotient. Ensuite, l'évaluation du système $\mathbf{f}$ et de son déterminant jacobien dans le quotient se font en $O(n^{\Omega} + nL)$ opérations. On regroupe le tout dans l'estimation ci-dessus. □

FIG. 9.3 – *Comparaison aux valeurs témoins, cas univarié*

procedure `Temoin`($\mathbf{f}$,$\mathcal{J}$,$\mathcal{R}'(t)$)

\# **Entree**
\# $\mathbf{f}$ système de $k[p_1,\dots,p_m][x_1,\dots,x_n]$
\# $\mathcal{J}$ idéal de $k[p_1,\dots,p_m]$
\# $\mathcal{R}'(t) = (t,q,\mathbf{w})$ résolution à coefficients dans $k(p_1,\dots,p_m)$
\# **Sortie**
\# **true** si modulo $\mathcal{J}$ aucun dénominateur ne s'annule, q est sans carré,
\# $\mathbf{f}(\mathbf{v}/q') = 0 \mod q$ et $\mathbf{jac}(f_1,\dots,f_n)(\mathbf{v}/q') \neq 0 \mod q$.
\# **false** sinon
end;

Utilisation : Cette fonction sera utilisée dans les chapitres suivants, à la base des tests d'arrêt de remontée de Newton, pour tester si une résolution candidate $\mathcal{R}'(t)$ à coefficients dans $k(p_1,\dots,p_m)$ est bien la description de l'ensemble des zéros simples de $\mathbf{f}$. Si le test répond "vrai", il y a un risque d'erreur, qui est quantifié ci-dessous.

9.2.2 Quantification des témoins

Notations Soit $\mathcal{R}'(t)$ le triplet $(t = \sum u_i x_i, q', \mathbf{v}')$ où q' et v'_i sont dans $k(p_1,\dots,p_m)[t]$, et q' a pour degré $\mathcal{D}_{\mathbf{x}}$. Soit $\mathcal{D}$ une borne sur le degré des numérateurs et dénominateurs des coefficients des polynômes de $\mathcal{R}'(t)$. Supposons que t est élément séparant générique pour $\mathcal{U}$ et que $\mathcal{R}'(t)$ est différent de la résolution générique $\mathcal{R}(t)$. Sous ces hypothèses, nous appellerons *bons témoins* pour $(\mathcal{R}'(t),\mathcal{R}(t))$ les points $\mathbf{p}$ satisfaisant les conditions de généricité pour t, où les spécialisations de $\mathcal{R}'(t)$ et $\mathcal{R}(t)$ sont possibles et ne coïncident pas.

Proposition 33 *Les points* $\mathbf{p}$ *mauvais témoins pour* $(\mathcal{R}'(t),\mathcal{R}(t))$ *sont inscrits dans une hypersurface de* $\mathbb{A}^m(\overline{k})$ *de degré au plus* $\mathcal{D}(t) + \mathcal{D}$.

Démonstration. Par hypothèse, tous les coefficients des monômes de $\mathcal{R}(t)$ et $\mathcal{R}'(t)$ ne sont pas égaux. Soient c et c' deux coefficients d'un même monôme, dans respectivement $\mathcal{R}(t)$ et $\mathcal{R}'(t)$, supposés distincts. Ce sont des fractions rationnelles, qui coïncident sur une hypersurface de $\mathbb{A}^m(\overline{k})$ de degré au plus $\mathcal{D}(t) + \mathcal{D}$. □

Soit $\mathbf{p}$ un point satisfaisant toutes les hypothèses de généricité pour t ; les solutions du système $\mathbf{f}(\mathbf{p},.)$ sont données par :

– la spécialisation de $\mathcal{R}(t)$ en $\mathbf{p}$,
– les traces au-dessus de $\mathbf{p}$ des composantes de $\mathcal{V}(f_1,\dots,f_n)$ dans $\mathbb{A}^{m+n}(\overline{k})$ sur lesquelles le déterminant jacobien de $(f_1,\dots,f_n)$ est nul,

et ces deux ensembles sont disjoints. Ainsi $\mathbf{p}$ est mauvais témoin si et seulement si $\mathcal{R}'(t)$ annule le système $\mathbf{f}(\mathbf{p},.)$ mais pas le déterminant jacobien, donc si et seulement si la fonction `Temoin` répond "vrai".

Chapitre 10

Calcul d'une résolution de Kronecker générique

Résumé

Nous présentons dans ce chapitre un algorithme probabiliste qui calcule une représentation de Kronecker des zéros simples génériques d'un système $\mathbf{f} = (f_1, \ldots, f_n)$ satisfaisant les hypothèses du chapitre 7. La résolution générique est calculée par approximations successives, à partir de la donnée d'une résolution spécialisée.

La complexité du processus de remontée est linéaire en la complexité d'évaluation du système $\mathbf{f}$, polynomiale en le nombre de paramètres et d'inconnues et en la taille de la sortie, elle-même bornée à partie du *degré géométrique* des objets considérés. Les choix des paramètres menant à l'échec sont enfermés dans des fermés algébriques dont le degré est dominé à partir des estimations des chapitres précédents.

Sommaire

Hypothèses et résultats

Soit $(f_1,\ldots,f_n)$ un système satisfaisant les hypothèses du chapitre 7. Pour une valeur $\mathbf{p}$ des paramètres, $\mathcal{C}_{\text{Zéro}}(\mathbf{f},\mathbf{p},k)$ désigne le coût en nombre d'opérations arithmétiques du calcul des points simples du système $\mathbf{f}(\mathbf{p},.)$ sur le corps k. On suppose ici que le cardinal de k est supérieur à $2d_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^n$. Enfin, si t est un élément séparant génériquement les points de $\mathcal{U}$, $\mathcal{D}(t)$ désigne le degré en $(p_1,\ldots,p_m)$ des numérateurs et dénominateurs des coefficients de la résolution $\mathcal{R}(t)$.

Théorème 9 (Complexité du calcul de la résolution générique) *On peut calculer par un algorithme probabiliste une résolution univariée des zéros simples génériques du système* $\mathbf{f}$ *en*

$$\mathcal{C}_{Zéro}(\mathbf{f},\mathbf{p},k) + O_{log}\big((n^{\Omega} + nL + nm)\mathcal{M}(\mathcal{D}_{\mathbf{x}})\mathcal{M}_s(4\mathcal{D}(t),m)\big)$$

opérations de k*, où* $\mathbf{p}$ *est une valeur aléatoire des paramètres et* t *est un élément séparant génériquement les points de* $\mathcal{U}$*. Les choix des valeurs aléatoires intervenant dans l'algorithme qui mènent à l'échec sont enfermés dans une hypersurface algébrique de* $\mathbb{A}^{3m-1}(\overline{k})$ *de degré au plus* $80nd_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^{3n}$.

Le coût du calcul est linéaire en la grandeur $\mathcal{M}_s(4\mathcal{D}(t),m)$, qui représente le coût de la multiplication des séries à m variables, à la précision $4\mathcal{D}(t)$. Si l'on dispose d'un algorithme de multiplication rapide des séries, ce terme peut devenir quasiment linéaire en le nombre de monômes de degré $4\mathcal{D}(t)$ en m variables, alors que l'on recherche des fractions rationnelles dont les numérateurs et dénominateurs sont des polynômes de degré $\mathcal{D}(t)$ en m variables.

Les premières sections de ce chapitre suivent les différentes phases du calcul, pour en préciser la complexité et la probabilité d'erreur. Ces résultats sont réunis dans la section 10.5, pour donner la complexité et la probabilité d'erreur de l'ensemble du processus. La dernière section est consacrée à la présentation d'une méthode probabiliste de vérification.

Pour conclure, mentionnons deux raffinements possibles de l'énoncé ci-dessus.

- S'il existe parmi les composantes des zéros simples génériques de $(f_1,\ldots,f_n)$ une composante $\mathcal{U}'$ dont on sait identifier la *trace* au-dessus de toute valeur des paramètres, cet algorithme permet d'en calculer une représentation. Nous avons vu dans le chapitre 8 que les bornes de degré s'adaptent directement à ce cas. Le principe du calcul reste le même : on sélectionne la composante lors du calcul spécialisé, et on effectue la remontée de Newton uniquement sur cette composante.

  L'estimation de complexité est analogue au cas général, et ne dépend cette fois que du degré de la composante recherchée. Je ne prouve pas explicitement ce résultat : la démarche et l'analyse des probabilités d'erreur sont exactement les mêmes que dans le cas général.
- Nous avons déjà mentionné plusieurs fois que les bornes de degré données ici sont grossières : nos résultats restent valides en remplaçant tous les termes de type $d_{\mathbf{p},\mathbf{x}}^m$ par des termes en le produit des degrés.

10.1 Description de l'algorithme

L'algorithme est basé sur l'opérateur de Newton formel décrit dans le chapitre 6.

- On calcule tout d'abord une résolution univariée $\mathcal{R}_0(t)$ des zéros simples de $(f_1,\ldots,f_n)$ au-dessus d'un paramètre $\mathbf{p}_1 = (p_{1,1},\ldots,p_{1,m})$ tiré au hasard.
- À partir de cette estimation initiale, on calcule des approximations successives $\mathcal{R}_i(t)$ de la résolution générique dans les séries formelles en les paramètres, autour de $(p_{1,1},\ldots,p_{1,m})$, par l'opérateur de Newton formel.

– Quand la précision est suffisante, on reconstruit les fractions rationnelles formant les coefficients de $\mathcal{R}(t)$.

On donne tout de suite la description de l'algorithme ; les sous-routines sont détaillées dans les sections suivantes. Outre le système $\mathbf{f}$, les arguments de cette procédure sont deux ensembles $\Omega \subset k$ et $\Gamma \subset k$, dans lesquels on va tirer les paramètres aléatoires intervenant au cours du calcul.

FIG. 10.1 – *Résolution de la fibre générique*

procedure `ResolutionFibreGenerique`($\mathbf{f}$,Ω,Γ)

\# **Entree**
\# le système $\mathbf{f}$
\# Ω et Γ : sous-ensembles de k
\# **Sortie**
\# $\mathbf{p}_1$,$\mathbf{p}_2$: Paramètres aléatoires de Ω^m;
\# γ : Changement de variables aléatoire de Γ^{m-1};
\# $\mathcal{R}(t) = (t,q,\mathbf{v})$ résolution des zéros de $\mathbf{f}$ à coefficients dans $k(p_1,\ldots,p_m)$

$\mathbf{p}_1,\mathbf{p}_2 \leftarrow$ Paramètres aléatoires de Ω^m;
$\gamma \leftarrow$ Changement de variables aléatoire de Γ^{m-1};
$b,(t,q_1,\mathbf{v}_1) \leftarrow$ `Resolution`($\mathbf{f}(\mathbf{p}_1,.)$);
if not b **then return** "echec" **fi** ;
$\mathcal{I} \leftarrow (p_1 - p_{1,1},\ldots,p_m - p_{1,m})$;
return $\mathbf{p}_1,\mathbf{p}_2,\gamma$, `RemonteeNewton`($\mathbf{f},\mathcal{I},(t,q_1,\mathbf{v}_1)$,`Arret`($\mathbf{f},\mathbf{p}_2,\gamma$),$4d^n_{\mathbf{p},\mathbf{x}}$);
end;

Cet algorithme travaille par réduction : on procède à une spécialisation des variables libres. À l'instar d'autres algorithmes de ce type, il n'est pas à l'abri du choix de mauvaises valeurs de spécialisation. Précisément, le caractère probabiliste relève des faits suivants :

– On effectue tous les calculs à partir d'une fibre spécialisée. Cette fibre peut ne pas avoir degré maximal, et sans information extérieure, on ne peut pas s'en assurer.
– La borne dont on dispose a priori pour arrêter le processus de remontée est une borne de Bézout donnée dans le chapitre 8. Pour éviter de monter jusqu'à cette borne, on utilise un test d'arrêt probabiliste.
– Enfin, le calcul de reconstruction rationnelle est probabiliste.

Les valeurs des différents paramètres menant à l'échec sont toutes enfermées dans des hypersurfaces algébriques.

Proposition 34 *On peut limiter les sources d'erreur dans les cas suivants :*

– *Si on travaille avec un seul paramètre, le calcul de reconstruction rationnelle devient déterministe.*
– *Si la variété $\mathcal{U}$ est position de Noether avec pour variables libres $(p_1,\ldots,p_m)$, alors toutes les fibres réduites ont même degré, et les coefficients dans la paramétrisation de Kronecker sont des polynômes en les paramètres.*

– *Si la variété $\mathcal{U}$ est en position de Noether projective avec pour variables libres $(p_1,\dots,p_m)$, alors le degré total des polynômes que l'on recherche est le degré d'une fibre spécialisée réduite.*

Dans le deuxième cas, seul le critère d'arrêt reste probabiliste, puisqu'on choisit de ne pas monter jusqu'à la borne de Bézout. Dans le dernier cas, on utilise un test d'arrêt en degré, l'algorithme ne peut plus donner de réponse fausse. Son temps d'exécution reste une variable aléatoire, puisqu'il faut trouver une fibre réduite où spécialiser, on a donc un algorithme Las-Vegas. Nous retrouvons les résultats de [79, 100], qui constituent en fait une étape de l'algorithme de résolution géométrique 1.1.3 du chapitre 1.

Dans les prochaines sections, nous donnons les complexités et probabilités d'erreur des fonctions :
- `Resolution` (résolution de la fibre spécialisée),
- `RemonteeNewton` (calcul des approximations successives),
- `Arret` (test d'arrêt de la remontée).

10.2 Résolution de la fibre spécialisée

Tirons au hasard un jeu de paramètres $\mathbf{p}_1 = (p_{1,1},\dots,p_{1,m})$. Il doit satisfaire toutes les conditions de généricité : il faut que les zéros dans $\mathcal{U}$ du système $\mathbf{f}(\mathbf{p}_1,.)$ soient en nombre fini et maximal, et simples. D'après l'hypothèse sur le cardinal de k, il existe un élément séparant t à coefficients dans k^n (voir la section 8.3 du chapitre 8). D'après la proposition 28 du chapitre 8, les mauvais choix pour le point $\mathbf{p}_1$ sont enfermés dans l'hypersurface discriminante de $\mathbb{A}^m(\overline{k})$ associée à t, qui est de degré au plus $d_{\mathbf{p},\mathbf{x}}^n(2d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}})$.

La première étape est le calcul d'une résolution $\mathcal{R}_0(t)$ des zéros simples du système $\mathbf{f}(\mathbf{p}_1,.)$; nous notons `Resolution` une telle procédure, puis $\mathcal{C}_{\text{Zéro}}(\mathbf{f},k)$ son coût en nombre d'opérations arithmétiques. Pour obtenir les zéros simples, il faut travailler en-dehors du lieu des zéros du jacobien $\mathbf{jac}(\mathbf{f})$ de $\mathbf{f}$ par rapport aux variables $\mathbf{x}$. Bon nombre d'algorithmes suivent l'astuce de Rabinowicz et procèdent par adjonction d'une variable supplémentaire T, et d'une équation $1 - T\mathbf{jac}(\mathbf{f})$. Dans le cas où le corps k est infini et parfait, l'utilisation de l'algorithme de résolution géométrique présenté dans la section 1.1.3 du chapitre 1 permet d'éviter ce détour, et de calculer directement les points simples.

Proposition 35 *Notons $\mathcal{V}_i$ l'adhérence de $\mathcal{V}(f_1,\dots,f_i) - \mathcal{V}(\mathbf{jac}(\mathbf{f}))$ et δ le maximum des degrés des variétés $(\mathcal{V}_1,\dots,\mathcal{V}_{n-1})$ sur une clôture algébrique de $k(p_1,\dots,p_m)$. On peut calculer les zéros simples de la fibre spécialisée par un algorithme probabiliste en*

$$O\big(n(nL + n^{\Omega})\mathcal{M}(d_{\mathbf{x}}\delta)^2\big)$$

opérations de k.

En pratique, cette première phase du calcul est de coût négligeable devant celui de la phase de remontée.

10.3 Remontée des séries par Newton-Hensel

Pour calculer les approximations successives de la résolution générique, nous utilisons l'algorithme `RemonteeNewton` présenté dans le chapitre 6.
- Après avoir effectué le changement de variables $p'_k = p_k - p_{1,k}$, on travaille modulo les puissances successives de l'idéal $\mathcal{I} = (p'_1,\dots,p'_m)$ de l'anneau $\mathcal{A} = k[[p'_1,\dots,p'_m]]$.

- Le point initial est la résolution $\mathcal{R}_0(t)$ des points simples du système $\mathbf{f}(\mathbf{p}_1,.)$; les approximations suivantes $\mathcal{R}_i(t)$ sont à coefficients dans les séries $k[[p'_1,\ldots,p'_m]]$ à précision 2^i.
- Le critère d'arrêt est présenté dans la section suivante.
- Le nombre maximal d'étapes est donné par la borne de Bézout sur le degré de la sortie.

Si $\mathbf{p}_1$ satisfait toutes les hypothèses de généricité, la remontée est possible et les coefficients des résolutions $\mathcal{R}_i(t)$ sont les développements en $\mathbf{p}_1$ des fractions rationnelles intervenant dans la résolution générique.

Notons $\mathcal{I}_i = \mathcal{I}^{2^i}$. Alors, avec nos notations de complexité, le coût d'une multiplication modulo $\mathcal{I}_i$ est $\mathcal{M}_s(2^i,m)$. À la $i$ème étape de remontée, on travaille avec des séries à précision 2^i, ce qui mène à l'estimation :

Proposition 36 (Coût de la remontée) *Le coût de la ième étape de remontée est*

$$O\big((n^\Omega + nL)\mathcal{M}_s(2^i,m)\mathcal{M}(\mathcal{D}_{\mathbf{x}})\big)$$

opérations de k.

Cette estimation est moins précise que celle donnée dans la section 6.5 du chapitre 6, mais plus lisible. De même, il est possible d'utiliser les éventuelles accélérations du calcul présentées dans la section 6.7.2 du même chapitre, quand la possibilité s'en présente.

Proposition 37 (Nombre d'étapes de remontée) *Supposons que $\mathbf{p}_1$ satisfait toutes les conditions de généricité. Alors il suffit de*

$$K = \lceil \log_2(\mathcal{D}(t)) \rceil + 1$$

étapes de remontée pour avoir assez de précision pour reconstruire tous les coefficients de la résolution.

Démonstration. Après k passages dans la boucle principale, on dispose d'une approximation de la résolution générique à précision 2^k. On peut donc reconstruire des fractions rationnelles dont les numérateurs et dénominateurs ont degré inférieur ou égal à 2^{k-1}. Les fractions intervenant dans la résolution générique ont un degré inférieur à $\mathcal{D}(t)$, et donc il faut au plus $\lceil \log_2(\mathcal{D}(t)) \rceil + 1$ étapes de remontée pour avoir suffisamment de précision pour reconstruire tous les coefficients de la résolution. □

10.4 Fonction d'arrêt

Nous décrivons maintenant la fonction `Arret`, qui met fin au processus de remontée. Ses arguments sont :

- Le système $\mathbf{f}$,
- une valeur témoin des paramètres $\mathbf{p}_2 = (p_{2,1},\ldots,p_{2,m})$,
- un changement de variables γ,
- la résolution $\mathcal{R}_i(t)$ à reconstruire.

À la $i$ème étape de remontée, on tente la reconstruction rationnelle des coefficients de la résolution $\mathcal{R}_i(t)$, par appel à la fonction `ReconstructionRationnelle`$(\mathcal{R}_i(t),\gamma)$. Si la reconstruction est possible, on la spécialise en la valeur témoin $\mathbf{p}_2$. Si la spécialisation est solution de $\mathbf{f}(\mathbf{p}_2,.)$ et décrit des points simples de $\mathcal{V}(f_1,\ldots,f_n)$, on décrète que la précision est suffisante, et que la résolution est la bonne. Ce test s'effectue par appel à la fonction `Temoin` de la section 9.2 du chapitre 9, pour l'idéal $\mathcal{J} = (p_1 - p_{2,1}, \ldots, p_m - p_{2,m})$ de $k[p_1, \ldots, p_m]$.

La fonction `RemonteeNewton` reçoit alors en argument la fonction partielle :

$$\texttt{Arret}(\mathbf{f},\mathbf{p},\gamma) \ : \mathcal{R} \mapsto \texttt{Arret}(\mathbf{f},\mathbf{p},\gamma,\mathcal{R}).$$

FIG. 10.2 – *Critère d'arrêt et reconstruction*

procedure `Arret`($\mathbf{f}$,$\mathbf{p}$,γ,$\mathcal{R}$)

\# **Entree**
\# le système $\mathbf{f}$
\# $\mathbf{p} = (p_{2,1}, \ldots, p_{2,m})$: valeur des paramètres
\# $\gamma \in k^{m-1}$: changement de variables
\# $\mathcal{R}$: résolution à coefficients dans $k[[p'_1, \ldots, p'_m]]$
Sortie
\# b,$\mathcal{R}_r$: b booléen ; si b est vrai, $\mathcal{R}_r$ est la reconstruction de $\mathcal{R}$

b,$\mathcal{R}' \leftarrow$ `ReconstructionRationnelle`($\mathcal{R}$,γ);
if not b **then return false** ; **fi** ;
$\mathcal{J} \leftarrow (p_1 - p_{2,1}, \ldots, p_m - p_{2,m})$;
$b \leftarrow$ `Temoin`($\mathbf{f}$,$\mathcal{J}$,$\mathcal{R}'$);
if b **then return true**,$\mathcal{R}'$; **else return false** ; **fi** ;
end;

Proposition 38 (Coût de la fonction `Arret`) *À la ième étape de remontée, le coût total de la fonction d'arrêt est*

$$O_{log}\big(\mathcal{M}(\mathcal{D}_{\mathbf{x}})(nm\mathcal{M}_s(2^i,m) + (n^{\Omega} + nL))\big)$$

opérations de k

Démonstration. Les complexités de chacune des opérations `ReconstructionRationnelle` et `Temoin` sont données dans un cadre générique dans la partie 9. Avec les notations de la partie 9.2, le coût $b(\mathcal{R},\mathcal{J})$ de la réduction d'un coefficient de $\mathcal{R}$ modulo $\mathcal{J}$ est le coût de l'évaluation de ce coefficient en $\mathbf{p}_2$, soit $\mathcal{T}(2^{i-1},m)$ opérations de k. Le quotient $k[p_1, \ldots, p_m]/\mathcal{J}$ est isomorphe à k, le coût $a(\mathcal{J})$ d'une opération dans ce quotient est celui d'une opération de k. Le résultat en découle. □

Probabilité d'erreur du critère d'arrêt. Si les points $(\mathbf{p}_1,\mathbf{p}_2)$ satisfont toutes les hypothèses de généricité, le test d'arrêt peut échouer pour deux raisons :
- il peut décrèter trop tôt que la précision est suffisante ;
- la reconstruction rationnelle de la dernière étape peut échouer.

Le premier cas d'échec se produit si la reconstruction rationnelle $\mathcal{R}'(t)$ d'une des premières étapes est possible, et, avec le vocabulaire de la section 9.2.2 du chapitre 9, si $\mathbf{p}_2$ est *mauvais témoin* pour $(\mathcal{R}'(t),\mathcal{R}(t))$.

Proposition 39 *Les points* $\mathbf{p}_2$ *mauvais témoins pour la reconstruction des zéros simples génériques sont enfermés dans une hypersurface de* $\mathbb{A}^m(\overline{k})$ *de degré au plus*

$$d_{\mathbf{p},\mathbf{x}}^n(\lceil \log_2(d_{\mathbf{p},\mathbf{x}}^n)\rceil + 2).$$

Démonstration. Les degrés des numérateurs et dénominateurs produits dans les étapes successives sont $1,2,4,\ldots,2^{K-1}$. On peut appliquer la proposition 33 du chapitre 9 à chacune de ces étapes. Ainsi, les mauvais témoins sont enfermés dans une hypersurface dont le degré est inférieur à

$$(\mathcal{D}(t)+1) + (\mathcal{D}(t)+2) + \cdots + (\mathcal{D}(t)+2^{K-1}).$$

Cette somme est majorée par $K\mathcal{D}(t) + 2^K$, et la borne de Bézout sur $\mathcal{D}(t)$ donnée dans le chapitre 8 et la borne sur K calculée dans la proposition 37 permettent de conclure. □

Le deuxième cas d'échec est l'échec de la dernière reconstruction rationnelle. D'après la section 9.1.2 du chapitre 9, les changements de variables γ qui font échouer la dernière reconstruction sont enfermés dans une hypersurface de $\mathbb{A}^{m-1}(\overline{k})$ de degré au plus

$$16n(4d_{\mathbf{p},\mathbf{x}}^n + 1)d_{\mathbf{p},\mathbf{x}}^{2n}d_{\mathbf{x}}^n.$$

10.5 Complexité et probabilité d'erreur

Résumons tous les points vus jusqu'ici. Nous appellerons $M = M(n,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}})$ la somme de :
- $2d_{\mathbf{p},\mathbf{x}}^n(2d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}})$ (pour contrôler les valeurs non génériques des paramètres),
- $d_{\mathbf{p},\mathbf{x}}^n(\lceil \log_2(d_{\mathbf{p},\mathbf{x}}^n)\rceil + 2)$ (pour contrôler les mauvais témoins),
- $16n(4d_{\mathbf{p},\mathbf{x}}^n + 1)d_{\mathbf{p},\mathbf{x}}^{2n}d_{\mathbf{x}}^n$ (pour contrôler les changements de variables non chanceux).

La borne en degré mentionnée dans le théorème de l'introduction est un majorant de l'entier M.

Théorème 10 *Les valeurs des paramètres* $(\mathbf{p}_1,\mathbf{p}_2,\gamma)$ *qui font échouer l'algorithme sont incluses dans une hypersurface de* $\mathbb{A}^{3m-1}(\overline{k})$ *de degré au plus* M.

Démonstration. L'algorithme réussit si :
- le point $\mathbf{p}_1$ satisfait toutes les conditions de généricité ;
- le point $\mathbf{p}_2$ satisfait toutes les conditions de généricité pour l'élément primitif obtenu lors de la résolution du système $\mathbf{f}(\mathbf{p}_1,.)$;
- le point $\mathbf{p}_2$ est bon témoin pour toutes les reconstructions intermédiaires ;
- le point γ permet la reconstruction rationnelle de la résolution.

Chacune de ces conditions se traduit par la non-annulation d'un polynôme en les coordonnées de $(\mathbf{p}_1,\mathbf{p}_2,\gamma)$ et d'après les sections précédentes, la somme des degrés de ces polynômes est inférieure à M. □

Exemple Soit ℓ un entier arbitraire, nous prenons pour espaces de paramètres Ω et Γ des sous-ensembles de k de cardinal ℓM. Choisissons les points $\mathbf{p}_1$ et $\mathbf{p}_2$ uniformément dans Ω^m, et le changement de variables γ paramétrant l'algorithme de reconstruction rationnelle uniformément dans Γ^{m-1}. Alors d'après le lemme de Zippel-Schwartz (corollaire 3 du chapitre 5), l'algorithme 10.1 s'arrête et renvoie une réponse correcte avec une probabilité supérieure à $1 - \frac{1}{\ell}$.

Théorème 11 (Complexité du calcul) *En cas de succès, la complexité de l'algorithme 10.1 est :*

$$\mathcal{C}_{\textit{Zéro}}(\mathbf{f},k) + O_{\textit{log}}\big((n^{\Omega} + nL + nm)\mathcal{M}(\mathcal{D}_{\mathbf{x}})\mathcal{M}_s(4\mathcal{D}(t),m)\big)$$

opérations de k.

Démonstration. D'après la proposition 37 ci-dessus, à la dernière étape de remontée de Newton on travaille sur les séries en les paramètres à précision au plus $4\mathcal{D}(t)$. La précision à laquelle on travaille est multipliée par 2 à chaque étape, la somme des coûts des étapes de remontée et de reconstruction est donc de l'ordre du coût de la dernière étape. Après quelques réductions, on obtient la borne demandée. $\square$

10.6 Vérification probabiliste d'une solution

Dans cette dernière section, nous voulons vérifier la validité d'une résolution candidat, à coefficients dans $k(p_1, \ldots, p_m)$. Une solution directe consiste à évaluer le système $\mathbf{f}$ sur la solution candidat, mais le coût de cette vérification est proche du coût du calcul générique. Nous chiffrons donc comment des calculs spécialisés peuvent témoigner que la résolution candidat décrit bien une composante des zéros simples génériques de $\mathbf{f}$.

L'analyse des probabilités faite ici est proche des estimations précédentes. Les hypothèses sont légèrement différentes, ce qui justifie cette digression :

- nous n'avons pas à prendre en compte la probabilité d'erreur d'une reconstruction rationnelle malheureuse ;
- par contre, nous ne nous autoriserons qu'à *évaluer* la résolution candidate sur des valeurs des paramètres. Nous ne pourrons donc pas détecter s'il *manque* des composantes.

Soit donc $\mathcal{R}'(t) = (t,q,\mathbf{v})$ une résolution à coefficients dans $k(p_1, \ldots, p_m)$, dont les coefficients sont des quotients de polynômes de $k[p_1, \ldots, p_m]$ ayant tous même dénominateur, de degré au plus $\mathcal{D}$. Nous supposons que $\mathcal{R}'(t)$ n'est pas la résolution d'une composante des points simples génériques de $(f_1, \ldots, f_n)$, et que le discriminant du polynôme minimal q n'est pas identiquement nul.

10.6.1 En degré maximal

Supposons pout commencer que la résolution $\mathcal{R}'(t)$ a le bon degré $\mathcal{D}_{\mathbf{x}}$ en t. Cette information peut provenir de diverses sources :

- un argument extérieur peut donner le nombre générique de solutions simples de $(f_1, \ldots, f_n)$;
- il est également possible de le déterminer par un autre calcul, tel un calcul de base de Gröbner pour un ordre du degré dans $k(p_1, \ldots, p_m)[x_1, \ldots, x_n]$.

Proposition 40 *Posons $M = \mathcal{D} + d_{\mathbf{p},\mathbf{x}}^n$. Soit ℓ un entier arbitraire, posons $\Omega = \{1, \ldots, \ell M\}$, et supposons que le cardinal de Ω est inférieur au cardinal de k. Tirons de manière uniforme un point $\mathbf{p}$ dans Ω^m. Si la spécialisation de $\mathcal{R}'(t)$ en $\mathbf{p}$ est possible, la probabilité pour qu'elle décrive des points simples de $(f_1, \ldots, f_n)$ et annule $\mathbf{f}$ est inférieure à $1/\ell$.*

Démonstration. Si la spécialisation de $\mathcal{R}'(t)$ en $\mathbf{p}$ est possible, décrit des points simples de $(f_1, \ldots, f_n)$ et annule $\mathbf{f}$, alors nécessairement la spécialisation de la véritable résolution générique est également possible et ces deux résolutions coïncident.
D'après la proposition 9.2.2 du chapitre 9, les points $\mathbf{p}$ pour lesquels ces résolutions coïncident sont inclus dans une hypersurface de degré au plus M. L'application du lemme de Zippel-Schwartz 3 du chapitre 3 donne le résultat. $\square$

Remarque Si on effectue le test avec succès sur $(M+1)^m$ valeurs des paramètres de $\{1, \ldots, M+1\}^m$, le lemme de Zippel-Schwartz implique que la solution est certifiée.

10.6.2 Dans le cas général

Nous ne faisons plus d'hypothèse sur le degré en t du polynôme minimal de $\mathcal{R}'(t)$, et nous appelons d ce degré. Nous n'avons plus l'assurance que t est élément séparant pour l'ensemble des zéros simples génériques, il faut donc s'intéresser à l'intersection du lieu défini par $\mathcal{R}'(t)$ avec la variété $\mathcal{U}$. Pour ne pas faire exploser les degrés, nous travaillons à partir d'une projection bien choisie.

Proposition 41 *Posons $M = d^n_{\mathbf{p},\mathbf{x}}(2d^n_{\mathbf{x}} + nd_{\mathbf{p},\mathbf{x}}) + (d^n_{\mathbf{x}} - d + 1)(2d-1)\mathcal{D} + d^n_{\mathbf{p},\mathbf{x}}$. Soient ℓ un entier arbitraire, et Ω l'ensemble $\{1,\dots,\ell M\}$, dont nous supposons le cardinal inférieur au cardinal de k. Tirons au hasard un point $\mathbf{p}$ dans Ω^m. Si la spécialisation de $\mathcal{R}'(t)$ est possible en $\mathbf{p}$, la probabilité qu'elle annule $\mathbf{f}$ est inférieure à $1/\ell$.*

Démonstration. Supposons que l'on a chassé tous les dénominateurs de $\mathcal{R}'(t)$, et appelons $\mathcal{D}$isc le résultant du polynôme minimal q avec sa dérivée. Soit $\mathcal{V}$ l'adhérence de la variété décrite par la résolution $\mathcal{R}'(t)$ en dehors du lieu des zéros du polynôme $\mathcal{D}$isc, et supposons que $\mathcal{V}$ n'est pas incluse dans $\mathcal{U}$. Alors il existe une forme linéaire $u = \sum u_i x_i$ telle que la projection de $\mathcal{U}$ sur l'espace des variables $(p_1,\dots,p_m,u)$ ne contienne pas celle de $\mathcal{V}$.

- D'après le chapitre 8, il existe un polynôme $P_{\mathcal{U}}$ de degré inférieur à $d^n_{\mathbf{p},\mathbf{x}}$ dont le lieu des zéros est l'adhérence de la projection de $\mathcal{U}$; c'est le polynôme éliminant de u dans l'ensemble des zéros simples de la fibre générique.
- Dans $\mathcal{V}$, la valeur de u est donnée par $q'(t)u - \sum u_i v_i(t) = 0$. La projection est décrite par le résultant $P_{\mathcal{V}}$ en t entre cette paramétrisation et $q(t)$; ce polynôme a un degré inférieur à $\mathcal{D}_{\mathcal{V}} := (2d-1)\mathcal{D}$.

Soit $\mathbf{p}$ une valeur des paramètres satisfaisant toutes les hypothèses de généricité pour u. Si la résolution $\mathcal{R}'(t)$ se spécialise en $\mathbf{p}$, alors les points décrits par $\mathcal{R}'(t)$ spécialisée en $\mathbf{p}$ n'annulent le système $\mathbf{f}$ que si la spécialisation de $P_{\mathcal{V}}$ y divise celle de $P_{\mathcal{U}}$. D'après le lemme suivant, il existe un polynôme Q non nul tel que c'est le cas seulement si $\mathbf{p}$ annule Q, et on peut prendre Q de degré inférieur à $(d^n_{\mathbf{x}} - d + 1)\mathcal{D}_{\mathcal{V}} + d^n_{\mathbf{p},\mathbf{x}}$.

Posons $M = d^n_{\mathbf{p},\mathbf{x}}(2d^n_{\mathbf{x}} + nd_{\mathbf{p},\mathbf{x}}) + (d^n_{\mathbf{x}} - d + 1)(2d-1)\mathcal{D} + d^n_{\mathbf{p},\mathbf{x}}$. Les points de Ω^m qui ne satisfont pas toutes les hypothèses de généricité ou annulent Q sont en cardinal inférieur à $M|\Omega|^{m-1}$ d'après le lemme de Zippel-Schwartz rappelé dans le chapitre 3. Cela prouve notre assertion. □

Voici la démonstation du lemme utilisé dans la dernière partie de l'argumentation.

Lemme 9 *Soient a et b des polynômes de $k[p_1,\dots,p_m][t]$ de degrés d_a et d_b en t et de degr'es $\mathcal{D}_a$ et $\mathcal{D}_b$ en $(p_1,\dots,p_m)$. Supposons que a ne divise pas b.*
Alors il existe un polynôme Q de $k[p_1,\dots,p_m]$ tel que, si $\mathbf{p}$ n'annule ni Q ni les coefficients dominants de a et b, alors la spécialisation de a en $\mathbf{p}$ ne divise pas la spécialisation de b en $\mathbf{p}$. Le polynôme Q peut être pris de degré inférieur à $(d_b - d_a + 1)\mathcal{D}_a + \mathcal{D}_b$.

Démonstration. Pour d entier, notons P_d l'espace vectoriel des polynômes à coefficients dans l'anneau $k[p_1,\dots,p_m]$ et de degré inférieur à d. Soit φ l'application

$$\begin{array}{rccc} \varphi : & P_{d_b-d_a} & \to & P_{d_b} \\ & f & \mapsto & af. \end{array}$$

Notons $\mathbf{S}$ la matrice de φ exprimée dans les bases canoniques de $P_{d_b-d_a}$ et P_{d_b}, à laquelle on adjoint la colonne des coefficients de b. Par hypothèse, b n'appartient pas à l'image de φ, la matrice $\mathbf{S}$ a donc rang maximal $d_b - d_a + 2$. Il existe donc une sous-matrice $\mathbf{s}$ de $\mathbf{S}$ de taille $(d_b - d_a + 2) \times (d_b -$

$d_a + 2)$ dont le déterminant Q est non nul. Ce déterminant est un polynôme de degré inférieur à $(d_b - d_a + 1)\mathcal{D}_a + \mathcal{D}_b$.
Soit $\mathbf{p}$ une valeur des paramètres. Si les termes de tête de a et b ne s'annulent pas en $\mathbf{p}$ alors la construction précédente commute avec la spécialisation. En conséquence, la spécialisation de a ne peut diviser la spécialisation de b que si le rang de la spécialisation de $\mathbf{S}$ chute. Cela implique que le déterminant Q défini ci-dessus s'annule en $\mathbf{p}$. □

Chapitre 11

Mise en pratique

Résumé

Pour conclure cette partie, nous décrivons l'implantation de l'algorithme présenté dans le chapitre 10. L'implantation est faite dans le langage Magma, et se base sur le prototype `kronecker` [140].

Nous comparons les temps de calculs effectués dans des corps finis aux fonctions de résolution de systèmes polynomiaux proposées par Magma version 2.6. Ces résultats sont encourageants, puisque nous faisons aboutir des calculs qui sont inaccessibles aux fonctions déjà existantes.

Sommaire

Nous décrivons pour finir la mise en œuvre de l'algorithme présenté dans le chapitre précédent. Après une brève description de l'implantation, nous comparons notre méthode de résolution aux fonctions proposées par Magma sur une série d'exemples. Il en ressort que notre méthode se comporte bien, et permet de faire aboutir des calculs autrement impossibles.

11.1 Implantation

Pour implanter l'algorithme du chapitre 10, il faut une procédure de résolution en dimension zéro sur k, une procédure de remontée de Newton dans un anneau de séries multivariées, pour une résolution univariée et une procédure de reconstruction rationnelle de séries multivariées.

De nombreux outils répondent à la première demande : l'implantation `kronecker`, l'utilisation des fonctions de résolution disponibles en Magma, ou une procédure de résolution adaptée à une situation particulière. Selon les exemples, nous pouvons choisir l'une ou l'autre possibilité. Ensuite, le code de `kronecker` fournit une fonction de remontée de Newton pour une résolution univariée, valable sur un anneau de séries formelles. Enfin, la fonction de calcul de reconstruction rationnelle multivariée est présentée dans l'annexe C, son implantation ne pose pas de difficulté.

Nous avons vu dans le chapitre 6 qu'à taille d'objets équivalente, la remontée des séries univariées est moins rapide que la remontée des entiers modulo les puissances successives d'un nombre premier, alors que ces deux questions sont de natures très proches. Cela laisse présager qu'il reste un gain potentiel conséquent pour nos algorithmes, avec une arithmétique plus efficace.

11.2 Résultats des calculs

11.2.1 Comparaison avec d'autres implantations

Nous avons comparé notre implantation avec la solution existante la plus aisément accessible en Magma version 2.6, le calcul d'une base de Gröbner pour un ordre lexicographique. Le moteur d'élimination de Magma est reconnu comme l'un des meilleurs parmi les implantations disponibles, et permet une comparaison sur une même plateforme. Notons que l'implantation FGb [1, 70] de J.-C. Faugère annonce des performances sensiblement supérieures.

Selon les cas, la base est calculée dans l'anneau $k(p_1,\dots,p_m)[x_1,\dots,x_n]$, ou dans l'anneau de polynômes $k[p_1,\dots,p_m,x_1,\dots,x_n]$; nous avons toujours trouvé une variable séparante dans ces exemples.

La première solution rend la même sortie que nos algorithmes, mais mise sous forme “Shape Lemma” ; la taille de la sortie sera plus importante. La seconde solution provoque le calcul des dégénérescences de la projection sur l'espace des paramètres, que l'on désire éviter. Si ces dégénérescences sont peu nombreuses, les résultats des deux calculs sont très semblables. Le calcul dans l'anneau de polynômes devient plus rapide que celui sur le corps des fractions : on évite l'empilement d'un domaine de polynômes sur un domaine de fractions. Au contraire, si les composantes dégénérées sont nombreuses, le calcul dans l'anneau de polynômes $k[p_1,\dots,p_m,x_1,\dots,x_n]$ peut donner des résultats difficilement utilisables.

11.2.2 Choix des systèmes

Nous traitons ici les systèmes suivants, pour la plupart tirés des parties d'applications IX et X.

- **P19** : la mise en équation est détaillée dans le chapitre 23, et le système est donné dans l'annexe E.2.

- **Birkhoff** : ces systèmes sont étudiés dans le chapitre 26, et donnés dans l'annexe E.7.
- $\mathbf{j_1j_2j_3}$: l'origine et les applications de ce système sont décrits dans le chapitre 25 ; le système est donné dans l'annexe E.5.
- **Hawes4** : ce système-test est donné dans l'annexe E.9.

Excépté le système P19, ces systèmes *s'évaluent bien* : le nombre de multiplications est faible par rapport aux nombre de monômes qui apparaissent dans les formes développées.

Dans cette partie, nous avons mesuré la complexité arithmétique de notre algorithme en comptant les opérations dans le corps de base à coût constant ; une manière de se ramener dans ce cadre est de travailler dans des corps finis. Le système P19 nous a explicitement été donné à résoudre sur le corps $\mathbb{F}_p$, où $p = 10^{19} + 51$; les autres sont des systèmes à coefficients rationnels, réduits modulo le même nombre premier. Les systèmes $j_1j_2j_3$ et Hawes4 sont ici à titre de tests : l'algorithme de la partie VII, reposant sur des calculs d'ensembles triangulaires, est plus adapté à leur résolution.

Le calcul produit des polynômes à coefficients dans des corps de fractions rationnelles. La taille du résultat est donc contrôlée par la combinatoire du nombre de monômes qui peuvent intervenir dans ces fractions. Pour des fractions de degré $\mathcal{D}$ à m variables libres, le nombre de monômes peut aller jusqu'à $\binom{\mathcal{D}+m}{m}$. Il est donc peu raisonnable de demander à écrire un polynôme éliminant qui dépend de 8,9,... variables, si son degré dépasse quelques unités. Les systèmes sur lesquels nous avons travaillé mettent en jeu une, deux ou trois variables libres.

11.2.3 Résultats

Tous les calculs sont faits sur une machine Compaq XP/1000 - Alpha EV6 de l'UMS MEDICIS [3], et les résultats sont résumés dans le tableau suivant.

- Les premières lignes donnent la mesure de ces systèmes, les grandeurs notées $(n,m,d_{\mathbf{p},\mathbf{x}},L)$; le degré partiel $d_{\mathbf{x}}$ est majoré par $d_{\mathbf{p},\mathbf{x}}$.
- Les deux lignes suivantes donnent respectivement le nombre de points simples génériques, puis le degré en les paramètres de la résolution générique.
- La ligne *Prob. Erreur de Calcul* donne la probabilité d'erreur du calcul, où nous avons tiré les paramètres intervenant dans l'algorithme uniformément dans le corps de base. Nous utilisons la majoration du degré de l'hypersurface à éviter, donnée dans l'introduction de cette partie, dans sa forme la plus fine.
- Nous avons donné dans le chapitre 10 la probabilité d'erreur d'un test de vérification d'une résolution pour laquelle le calcul s'est bien passé. Nous avons distingué deux cas, selon que l'on a l'assurance que le nombre de points décrits par la résolution candidat est bien le nombre de solutions simples génériques ou non. Dans la ligne *Prob. Erreur Vérification*, nous mettons ce test en pratique.

  Dans les trois premiers exemples, nous sommes dans le cas favorable pour le test de vérification. Dans le premier cas, des arguments théoriques nous protègent, dans les deux seconds, c'est un calcul de base de Gröbner pour un ordre du degré qui donne ce résultat. Pour les systèmes $j_1j_2j_3$ et Hawes4, nous n'avons plus cette assurance. Le test peut uniquement assurer que nous avons bien la description d'*une* composante de l'ensemble des zéros simples.
- La ligne *Temps* indique le temps de calcul de notre implantation ; le système spécialisé est résolu par un calcul de base de Gröbner en dimension zéro.

FIG. 11.1 – *Résolution modulo* NextPrime(10^{19})

Système	P19	Birkhoff-3	Birkhoff-10	$j_1j_2j_3$	Hawes4
Variables n	4	3	3	4	6
Paramètres m	2	1	1	2	2
Degré Max.$d_{\mathbf{p},\mathbf{x}}$	10	6	6	10	9
Mult., Add. L	253 ,209	91,83	80,86	41,23	55,52
Degré générique $\mathcal{D}_{\mathbf{x}}$	64	34	136	30	120
Degré en les paramètres $\mathcal{D}(t)$	44	5	19	11	46
Prob. Erreur de Calcul	0.01 %	2 10^{-6}%	2 10^{-6}%	3 %	0.26 %
Prob. Erreur Vérification	1.5 10^{-14}%	2 10^{-15}%	2 10^{-15}%	1.5 10^{-8}%	6.3 10^{-9}%
Temps	2.6 h.	6 s.	8 min.	1 min.	6 h.
Gröbner 1	∞	11 s.	> 5 h.	> 5 h.	∞
Gröbner 2	∞	22 s.	> 5 h.	> 5 h.	∞

– La ligne *Gröbner 1* donne le temps le calcul d'une base de Gröbner lexicographique dans $k(p_1,\dots,p_m)[x_1,\dots,x_n]$, et la ligne *Gröbner 2* indique le temps de calcul d'une base de Gröbner lexicographique dans l'anneau $k[p_1,\dots,p_m,x_1,\dots,x_n]$. Dans ce dernier cas, dans l'exemple Birkhoff-3, la base comporte 17 éléments, sa taille est de 620 Ko., alors que nos calculs donnent un résultat de taille 11 Ko.

  Le signe ∞ indique que le processus a été arrêté quand la quantité de mémoire occupée a étouffé les calculs.

Conclusion Notre implantation a un bon comportement en pratique, et mène à bien des calculs où échouent les fonctions proposées par Magma. Ce comportment est typique des méthodes modulaires : on n'effectue que les calculs nécessaires, les expressions intermédiaires et la sortie ont une taille optimale. Dans tous ces exemples, la simple taille d'une base de Gröbner lexicographique explique certainement la difficulté à mener son calcul à terme.

Il faut garder à l'esprit que l'on compare ici un algorithme à un algorithme déterministe. Malgré tout, nous n'avons pas rencontré de situation où nous avons pu constater que sa sortie était erronnée. Pour les applications où nous l'avons utilisé, les résultats ont donné satisfaction.

11.3 Remarques pratiques

11.3.1 Calcul en dimension zéro

La première phase de calcul est toujours la résolution d'un système de dimension zéro sur le corps de base. En pratique, nous avons choisi entre les deux possibilités suivantes.

- Le calcul d'une résolution géométrique par l'implantation `kronecker` tient compte de la complexité d'évaluation du système d'entrée. L'algorithme est probabiliste, mais nous n'avons pas constaté de cas de mauvaise réponse.
- Les performances du moteur de base de Gröbner de Magma sont de haut niveau. Cette solution offre une sortie certifiée, mais ne sait pas tenir compte de la complexité d'évaluation du système.

Voici les performances de ces implantations pour les systèmes vus ci-dessus. Dans ces situations de dimension zéro sur un corps fini, la taille d'une base de Gröbner lexicographique est sensiblement la même que celle d'une résolution de Kronecker. L'implantation `kronecker` est interprêtée et se base sur une arithmétique probablement moins optimisée, ce qui explique certainement l'écart des temps.

FIG. 11.2 – *Résolution en dimension zéro*

Système	P19	Birkhoff-3	Birkhoff-10	$j_1j_2j_3$	Hawes4
`kronecker`	483 s.	70 s.	215 s.	85 s.	880 s.
Gröbner Magma	6.5 s.	0.05 s.	4.5 s.	1.6 s.	17.2 s.

11.3.2 Importances relatives des différentes phases

L'estimation de complexité ne fait pas ressortir les importances relatives des différentes phases du calcul en pratique. Les trois points autour desquels le calcul s'articule sont :

- la résolution en dimension zéro,
- la phase de remontée des séries,
- la reconstruction rationnelle des coefficients.

Le temps de calcul est passé quasiment intégralement dans la phase de remontée de Newton. La dernière section du chapitre 6 a montré que comportement dépend de la manière dont se factorise la résolution spécialisée : en présence de nombreux petits facteurs, les temps de calcul sont bien meilleurs. Nos résultats sont donc tributaires du choix du point de spécialisation.

11.3.3 Importance de la complexité d'évaluation

À chaque étape de remontée, le temps se partage entre l'évaluation du système $\mathbf{f}$ et de sa jacobienne sur la résolution en cours, suivies de produits matriciels. La première phase est conditionnée par la complexité d'évaluation du système ; nous avons montré dans le chapitre 6 son influence sur l'exemple du système Hawes4. Sur l'ensemble de exemples traités, on retrouve un comportement sensiblement linéaire en ce paramètre.

La plupart des systèmes que nous avons eu à résoudre peuvent se mettre sous une forme qui s'évalue bien, suivant le credo mis en avant dans les articles du groupe TERA. Parmi les exemples de ce chapitre, seul le système P19 ne se présente pas sous une forme simple. Les autres systèmes présentés dans les parties d'applications suivent souvent cette bonne ligne de conduite. La plupart des logiciels de calcul formel codent les polynômes par la liste de leur monômes, qui fait disparaître d'éventuelles bonnes propriétés d'évaluation : il a souvent fallu reprendre les systèmes à la source, pour obtenir la suite des calculs qui les constituent.

Le logiciel Maple [151] ne développe pas systématiquement les polynômes ; cette propriété permet de conserver des calculs d'évaluation de longueur modérée lors de la mise en équations d'un problème. Maple dispose en outre d'une fonction `optimize`, systématiquement utilisée en pratique, qui permet de diminuer le nombre d'opérations, via diverses heuristiques (recherche de motifs,...). La détermination d'un calcul d'évaluation de coût optimal pour un système donné reste un problème des plus complexes — qui inclut la factorisation, la décomposition..., hors de notre portée.

Sixième partie

Considérations de complexité binaire

Chapitre 12

Vue d'ensemble

12.1 Hypothèses et notations

Nous considérons désormais un système de polynômes $\mathbf{f} = (f_1, \ldots, f_n)$ de $\mathbb{Q}[p_1, \ldots, p_m, x_1, \ldots, x_n]$. Nous proposons un raffinement des idées développées dans la partie précédente à ce cas particulier, les autres hypothèses restant inchangées.

Notations 3 *Nous noterons $\mathcal{U}$ la réunion des composantes de $\mathcal{V}(f_1, \ldots, f_n) \subset \mathbb{C}^{m+n}$ où le déterminant jacobien de $\mathbf{f}$ par rapport aux variables $\mathbf{x}$ n'est pas identiquement nul.*

L'entrée est mesurée par des grandeurs syntaxiques :

- *le système $(f_1, \ldots, f_n)$ est donné par un calcul d'évaluation de taille L ;*
- *le degré partiel en $(x_1, \ldots, x_n)$ des polynômes $(f_1, \ldots, f_n)$ est borné par $d_{\mathbf{x}}$;*
- *le degré total des polynômes $(f_1, \ldots, f_n)$ est borné par $d_{\mathbf{p},\mathbf{x}}$, supposé au moins égal à 2 ;*
- *les polynômes $(f_1, \ldots, f_n)$ sont à coefficients entiers, de hauteur bornée par h.*

Les objets sous-jacents sont mesurés par des degrés géométriques :

- *$\mathcal{D}_{\mathbf{x}}$ est le nombre générique de solutions simples du système $(f_1, \ldots, f_n)$. Par le théorème de Bézout, ce nombre est inférieur au produit des degrés partiels en $\mathbf{x}$ des polynômes f_i, donc à $d_{\mathbf{x}}^n$.*
- *$\mathcal{D}_{\mathbf{p},\mathbf{x}}$ désigne le degré de la variété $\mathcal{U}$. Par le théorème de Bézout, ce degré est inférieur au produit des degrés des polynômes f_i, donc à $d_{\mathbf{p},\mathbf{x}}^n$;*
- *pour un élément t séparant les points de $\mathcal{U}$, nous noterons $\mathcal{D}(t)$ et $\mathcal{H}(t)$ le degré en $(p_1, \ldots, p_m)$ et la hauteur de la résolution générique correspondante.*

Sous ces hypothèses, on s'intéresse à une description des zéros simples génériques du système $(f_1, \ldots, f_n)$ par une résolution de Kronecker à coefficients dans $\mathbb{Q}(p_1, \ldots, p_m)$. Le schéma de cette partie est analogue à la partie précédente : nous nous consacrons à borner des hauteurs, pour en déduire un algorithme de calcul d'une résolution générique, dont la complexité *binaire* est polynomiale.

Nous adoptons les notations de complexité rappelées dans l'introduction du chapitre 7 et nous utiliserons la notation supplémentaire $\mathcal{M}_{\mathbb{N}}(d)$, qui désigne le nombre d'opérations binaires nécessaires à la multiplication de deux entiers de d chiffres.

12.2 Résultats et plan des chapitres suivants

Le chapitre 13 est l'analogue du chapitre 8 de la partie précédente. Nous y donnons des bornes sur la hauteur des coefficients qui peuvent apparaître dans la résolution recherchée.

Théorème 12 (Hauteur d'une résolution de Kronecker) *Soient t un élément séparant génériquement les points de $\mathcal{U}$, à coefficients dans $\mathbb{Z}$ de hauteur h_t et $\mathcal{R}(t)$ la résolution correspondante :*

$$q(t) = 0, \quad \left\{ \begin{array}{rcl} \frac{dq(t)}{dt} x_1 & = & v_1(t) \\ & \vdots & \\ \frac{dq(t)}{dt} x_n & = & v_n(t). \end{array} \right.$$

Les polynômes q et v_i sont dans $\mathbb{Q}(p_1, \ldots ,p_m)[t]$, le polynôme q est unitaire, de degré $\mathcal{D}_{\mathbf{x}}$ et les polynômes v_i sont de degré strictement inférieur à $\mathcal{D}_{\mathbf{x}}$. Alors tous les coefficients de ces polynômes peuvent s'écrire comme des quotients de polynômes de $\mathbb{Z}[p_1, \ldots ,p_m]$ de degré inférieur à $\mathcal{D}_{\mathbf{p},\mathbf{x}}$ et de hauteur inférieure à

$$2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + h_t + 6(n+2m+3)\log(2n+m+2)\big).$$

Nous définissons ensuite un *entier discriminant*, qui jouera un rôle similaire à l'hypersurface discriminante de la partie précédente : dans le cas général, nous avons cherché des valeurs des paramètres en-dehors de cette hypersurface ; ici, nous allons chercher des nombres premiers qui ne divisent pas cet entier, en vue de faire du calcul modulaire.

Théorème 13 (Premiers de bonne réduction d'une résolution) *Soient t un élément qui sépare génériquement les points de $\mathcal{U}$, à coefficients dans $\mathbb{Z}$ de hauteur h_t, $\mathcal{R}(t)$ la résolution correspondante, et $\mathbf{p}$ une valeur des paramètres à coordonnées entières bornées par M, satisfaisant toutes les hypothèses de généricité pour t.*
Il existe un nombre entier $e_{Red}(\mathbf{p},t)$ non nul tel que pour tout nombre premier p ne divisant pas $e_{Red}(\mathbf{p},t)$:

- *les dénominateurs et le discriminant du polynôme minimal de $\mathcal{R}(t)$ spécialisée en $\mathbf{p}$ sont non nuls modulo p,*
- *le déterminant jacobien du système $\mathbf{f}$ est non nul sur les points décrits par $\mathcal{R}(t)$ modulo p.*

Dans ce cas, la réduction modulo p de la spécialisation de $\mathcal{R}(t)$ en $\mathbf{p}$ est une résolution des zéros simples du système $\mathbf{f}(\mathbf{p},.)$ pris modulo p. La hauteur de $e_{Red}(\mathbf{p},t)$ peut être prise inférieure à

$$\mathcal{H}_{Red}(h,h_t,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}},M) = 8d_{\mathbf{x}}^{2n}(n+2)^3\big(h + h_t + d_{\mathbf{p},\mathbf{x}}(\log M + \log(m+1)) + 6\log(2n+2)\big).$$

Le chapitre 14 adapte les résultats du chapitre 9 à cette nouvelle situation : nous étendons les fonctions qui y étaient présentées au cas du calcul sur les rationnels, et donnons des conditions de non-dégénérescence pour des calculs spécialisés. Aux conditions d'ordre géométrique du chapitre 9, nous rajoutons des conditions de non-divisibilité qui assurent que les calculs se comportent bien par réduction modulaire. Ces conditions sont quantifiées à partir des bornes de hauteur établies dans le chapitre 13.

Dans le chapitre 15, nous étendons l'algorithme du chapitre 10 au calcul sur les rationnels. À nouveau, l'algorithme utilise un processus de remontée de Newton, et la majoration de la probabilité d'erreur prend en compte toutes les possibilités de dégénérescence du calcul. Cette fois, nous nous intéressons à la complexité binaire du processus, pour obtenir le résultat ci-dessous.

Si p_1 est un nombre premier et $\mathbf{p}_1$ une valeur des paramètres à coordonnées entières, $\mathcal{C}_{\text{Zéro}}(\mathbf{f},\mathbf{p}_1,p_1)$ désigne le coût de la résolution du système $\mathbf{f}(\mathbf{p}_1,.)$ pris modulo p_1. L'expression de l'entier h_1 ci-dessous est un majorant d'une grandeur définie dans le chapitre 15, et contrôle le choix du nombre premier p_1.

Théorème 14 (Complexité du calcul de la résolution générique sur les rationnels) *Soit $h_1 = 10^4 h n^2 (n + m + 10) d_{\mathbf{p},\mathbf{x}}^{5n+2}$. On peut trouver un élément séparant générique t et calculer la résolution générique $\mathcal{R}(t)$ en*

$$\mathcal{C}_{Z\acute{e}ro}(\mathbf{f},\mathbf{p}_1,p_1) + O_{log}\big(\mathcal{M}(\mathcal{D}_{\mathbf{x}})(n^{\Omega} + nL + nm)\mathcal{M}_s(4\mathcal{D}(t),m)\mathcal{M}_{\mathbb{N}}(\mathcal{H}(t) + \log h + mn \log d_{\mathbf{p},\mathbf{x}})\big)$$

opérations binaires, où p_1 est un nombre premier compris entre h_1 et $2h_1$, $\mathbf{p}_1$ une valeur des paramètres à coordonnées entières et t un élément séparant génériquement les points de $\mathcal{U}$, le tout avec une probabilité d'erreur inférieure à $1/2$.

Par rapport au résultat de la partie précédente, cette borne de complexité signifie que le temps de calcul est *quasiment linéaire* en la hauteur de la sortie.

Dans le dernier chapitre, nous mettons en pratique cet algorithme. À nouveau, puisque nous ne considérons que des résolutions sous forme univariée, nous utilisons l'algorithme remontée de Newton issu du code `kronecker` écrit par G. Lecerf. L'expérience montre que l'implantation permet de résoudre des systèmes inaccessibles aux calculs de bases de Gröbner permis par Magma version 2.6. Comme dans la partie précédente, nous n'avons pas rencontré de situation où la sortie des calculs apparaissait erronée.

Remarque Il n'est pas difficile de voir que tous les nombres premiers sauf un nombre fini sont de “bonne réduction” pour l'ensemble du calcul, mais il est moins immédiat d'obtenir des estimations explicites. À partir des résultats du chapitre 13, leur mise en place dans les chapitres 14 et 15 nécessite un travail sans réelle difficulté, mais assez technique.

Les références mentionnées dans le chapitre 7 sur la question des systèmes à paramètres se retrouvent dans ce cas particulier. Citons en outre [123, 79, 84] et [182], dont les algorithmes prennent explicitement en compte une notion de hauteur du système.

Chapitre 13

Bornes de hauteur

Résumé

Nous proposons ici des majorations sur la hauteur d'objets liés à la représentation des zéros simples génériques du système $(f_1, \ldots, f_n)$, qui interviendront constamment pour contrôler la complexité et la probabilité d'erreur des algorithmes des chapitres suivants.

Nous prouvons tout d'abord des bornes sur la hauteur des coefficients qui peuvent apparaître dans une représentation de Kronecker générique ; elles sont démontrées via le théorème de Bézout arithmétique 14 du chapitre 4.

Nous en déduisons ensuite des bornes sur la taille des nombre premiers p pour lesquels une résolution décrivant les zéros simples du système $\mathbf{f}(\mathbf{p},.)$ ne dégénère pas modulo p — en un sens qui sera précisé. Ces bornes sont obtenues en utilisant le Nullstellensatz arithmétique 4 du chapitre 4. En conséquence, dans la section 13.4, nous définissons un entier e_{Red} qui jouera le rôle d'un entier discriminant.

Sommaire

Hypothèses et résultats

Ce chapitre poursuit l'étude théorique de son analogue le chapitre 8, en décrivant cette fois la *hauteur* des polynômes qui apparaissent dans les résolutions génériques.

Rappelons la définition du polynôme éliminant générique défini dans les sections 8.2.2 et 8.2.3 du chapitre 8 :

Définition 13 *Notons $\widetilde{\mathcal{U}}$ la sous-variété de $\mathbb{C}^{n+1} \times \mathbb{C}^{n+m}$:*

$$\widetilde{\mathcal{U}} = \{(t,t_1,\ldots,t_n,p_1,\ldots,p_m,x_1,\ldots,x_n) \text{ t.q. } (p_1,\ldots,p_m,x_1,\ldots,x_n) \in \mathcal{U} \text{ et } t = \sum t_i x_i\},$$

puis φ l'application

$$\begin{array}{rccl} \varphi : & \widetilde{\mathcal{U}} & \to & \mathbb{C}^{n+m+1} \\ & (t,t_1,\ldots,t_n,p_1,\ldots,p_m,x_1,\ldots,x_n) & \mapsto & (t,t_1,\ldots,t_n,p_1,\ldots,p_m). \end{array}$$

Nous appelons polynôme éliminant générique et notons M_{Gen} un polynôme décrivant l'image de φ.

Dans la première section, nous bornons la hauteur de ce polynôme, puis, par les techniques de la section 8.3 du chapitre 8, nous en déduisons une borne sur la taille des entiers apparaissant dans une résolution de Kronecker générique. Pour un élément séparant t de hauteur h_t, l'estimation est d'ordre de grandeur $(n+m)(h+h_t+\log(n+m+1))d_{\mathbf{p},\mathbf{x}}^n$.

Étant donné un élément séparant t et une valeur des paramètres $\mathbf{p}$, nous définissons ensuite un entier discriminant noté $e_{\text{Red}}(\mathbf{p},t)$ qui contrôle les premiers p pour lesquels la résolution de l'ensemble des zéros simples du système $\mathbf{f}(\mathbf{p},.)$ donnée pour l'élément séparant t ne dégénère pas modulo p. La première étape, dans la section 13.3, est l'étude pour un système ne dépendant pas de paramètres ; on en déduit le résultat pour le cas général dans la section 13.4. Pour une valeur des paramètres à coefficients bornés par M, l'entier discriminant a une hauteur d'ordre $n^{O(1)}(h+h_t)d_{\mathbf{p},\mathbf{x}}^{2n}\log M$.

La suite de cette partie propose un algorithme de calcul d'une résolution sur les rationnels par des techniques de calcul modulaire. Les deux bornes de hauteur introduites ci-dessus interviendront dans le choix des premiers de réduction.

13.1 Borne sur la hauteur du polynôme éliminant générique

Nous donnons pour commencer une borne sur la hauteur d'un multiple bien choisi de M_{Gen}. Cette borne est une borne de Bézout ; il faut rajouter quelques termes correctifs pour l'écrire exactement.

Lemme 10 *La hauteur de la variété $\widetilde{\mathcal{U}}$ vérifie l'inégalité :*

$$h(\widetilde{\mathcal{U}}) \leq 2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + (3n+m+2)\log(2n+m+2)\big).$$

Démonstration. La variété $\widetilde{\mathcal{U}}$ est incluse dans l'ensemble algébrique défini par $(f_1,\ldots,f_n)$ et l'équation supplémentaire $t - \sum t_i x_i$. Appliquons la borne de Bézout arihtmétique donnée dans la proposition 14 du chapitre 4, en remplaçant (n,s) par $(2n+m+1,n+1)$. Nous obtenons un majorant sur la hauteur de l'ensemble des zéros du système précédent, donc en particulier sur la hauteur de $\widetilde{\mathcal{U}}$. □

Lemme 11 *La projection de la variété $\widetilde{\mathcal{U}}$ par l'application φ a une hauteur majorée par*

$$2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + 6(n+m+1)\log(2n+m+2)\big).$$

Démonstration. Le lemme [124, Lemma 2.5] donne une borne sur la hauteur de l'image de $\widetilde{\mathcal{U}}$ par une projection linéaire. L'adaptation de cette borne à notre contexte donne le résultat. □

Proposition 42 (Hauteur du polynôme éliminant générique) *On peut prendre le polynôme M_{Gen} à coefficients entiers de hauteur majorée par*

$$2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + 6(n+m+2)\log(2n+m+2)\big).$$

Démonstration. On applique la proposition 13 du chapitre 4, qui donne une majoration sur la hauteur d'un polynôme définissant une hypersurface. □

Remarque Il est également possible de borner la hauteur de M_{Gen} en fonction de la hauteur de la variété $\mathcal{U}$. On préfère ici avoir une borne directement en fonction des données du système $(f_1, \dots, f_n)$. En effet, cette borne servira *in fine* à contrôler des probabilités d'erreur, il faut donc la donner en fonction des mesures dont on dispose.

Les résultats précédents ne concernent que le polynôme éliminant générique, qui pourra ensuite se spécialiser en le polynôme éliminant d'un élément séparant donné. Si la variable x_i n'est pas séparante, il faut revenir à la définition de son polynôme éliminant M_{x_i} comme image d'une projection, pour éviter le calcul d'une partie sans carré. Par des arguments similaires à ceux développés ci-dessus, on peut prouver la proposition suivante, qui interviendra dans la partie VII.

Proposition 43 (Hauteur du polynôme éliminant) *Pour toute variable x_i, les coefficients du polynôme éliminant de x_i peuvent s'écrire comme des polynômes de $\mathbb{Z}[p_1, \dots, p_m]$ de degré inférieur à $d_{\mathbf{p},\mathbf{x}}^n$ et de hauteur inférieure à*

$$d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + 6(n+m+2)\log(n+m+1)\big).$$

13.2 Borne sur la hauteur d'une résolution de Kronecker

Nous pouvons maintenant donner une borne sur la hauteur des coefficients intervenant dans une résolution de Kronecker générique, à partir de la borne sur la hauteur de M_{Gen}.

Théorème 15 (Hauteur d'une résolution de Kronecker) *Soient t un élément séparant générique à coefficients dans $\mathbb{Z}$ de hauteur h_t et $\mathcal{R}(t)$ la résolution correspondante :*

$$q(t) = 0, \quad \left\{ \begin{array}{rcl} \frac{dq(t)}{dt} x_1 & = & v_1(t) \\ & \vdots & \\ \frac{dq(t)}{dt} x_n & = & v_n(t). \end{array} \right.$$

Les polynômes q et v_i sont dans $\mathbb{Q}(p_1, \dots, p_m)[t]$, le polynôme q est unitaire, de degré $\mathcal{D}_{\mathbf{x}}$ et les polynômes v_i sont de degré strictement inférieur à $\mathcal{D}_{\mathbf{x}}$. Alors tous les coefficients de ces polynômes peuvent s'écrire comme des quotients de polynômes de $\mathbb{Z}[p_1, \dots, p_m]$ de degré inférieur à $\mathcal{D}_{\mathbf{p},\mathbf{x}}$ et de hauteur inférieure à

$$2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + h_t + 6(n+2m+3)\log(2n+m+2)\big).$$

Démonstration. Nous avons déjà obtenu la borne en degré dans le théorème 8 du chapitre 8, il ne reste à montrer que la borne sur les hauteurs. Pour ceci, rappelons que le polynôme éliminant générique est à coefficients dans $\mathbb{Z}$, et que sa hauteur est inférieure à

$$2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + 6(n+m+2)\log(2n+m+2)\big).$$

Le degré t_i et en t de M_{Gen} est inférieur à $\mathcal{D}_{\mathbf{x}}$; on en déduit par une majoration grossière que la hauteur de ses dérivées partielles par rapport aux variables t_i et t est inférieure à

$$2d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + 6(n+m+2)\log(2n+m+2)+1\big).$$

Nous avons vu dans la section 8.3 du chapitre 8 que la spécialisation dans ces dérivées partielles des variables (t_i) en les valeurs (u_i) donne les paramétrisations $\mathbf{v}$. La borne 6 du chapitre 3 permet de borner tous les coefficients du polynôme éliminant et de ses dérivées par

$$2d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + 6(n+m+2)\log(2n+m+2)+1\big) + \log \mathcal{D}_{\mathbf{x}} + \mathcal{D}_{\mathbf{x}}(h_t + \log(n+1)).$$

On majore $\mathcal{D}_{\mathbf{x}}$ par le nombre de Bézout $d^n_{\mathbf{x}}$ pour obtenir la borne :

$$2d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + h_t + 6(n+m+3)\log(2n+m+2)\big).$$

Rendons le polynôme éliminant unitaire ; d'éventuelles simplifications peuvent avoir lieu, qui feront croître la hauteur des coefficients. L'application de la borne de Mignotte (proposition 8 du chapitre 3) indique que la hauteur des facteurs croît au plus de $m(d^n_{\mathbf{p},\mathbf{x}} + \log(d^n_{\mathbf{p},\mathbf{x}} + m))$. On peut finalement la majorer par

$$2d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + h_t + 6(n+2m+3)\log(2n+m+2)\big),$$

ce qui est le résultat demandé. □

Remarque Si nous décidons de garder la résolution à coefficients dans $\mathbb{Z}[p_1,\dots,p_m][x_1,\dots,x_n]$, nous avons la même borne sur la hauteur des coefficients.

Définition 14 *Nous adopterons les définitions suivantes :*

- *Pour t élément séparant générique à coefficients entiers, nous noterons $\mathcal{H}(t)$ la hauteur des coefficients de la résolution $\mathcal{R}(t)$. Le polynôme minimal est supposé unitaire ; les coefficients sont écrits comme des quotients de polynômes de $\mathbb{Z}[p_1,\dots,p_m]$, dont les contenus du numérateur et du dénomiteur ont un pgcd égal à un.*
- *Sous les mêmes hypothèses, si t est de hauteur h_t, $\mathcal{H}(h,h_t,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}})$ désignera la* majoration *ci-dessus sur $\mathcal{H}(t)$.*

$$\begin{aligned}
\mathcal{H}(h,h_t,n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}}) &= 2d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + h_t + 6(n+2m+3)\log(2n+m+2)\big)\\
&= (h+h_t)(nmd^n_{\mathbf{p},\mathbf{x}})^{O(1)},\\
\textit{avec}\quad \mathcal{H}(t) &\leq \mathcal{H}(h,h_t,n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}}).
\end{aligned}$$

13.3 Étude des dégénérescences en dimension zéro

Dans cette section, nous supposons que les polynômes $(f_1,\dots,f_n)$ sont dans $\mathbb{Q}[x_1,\dots,x_n]$: il n'y a (temporairement) plus de paramètres en jeu. Si les zéros de $(f_1,\dots,f_n)$ dans $\mathbb{C}^n$ forment un ensemble fini, il n'y a pas de raison que cette propriété se converse modulo un nombre premier. C'est un phénomène analogue à la présence de fibres de dimension positive pour une projection dominante et génériquement finie, et c'est le cas dans l'exemple suivant, tiré de [63].

Exemple [Dégénérescence modulo p] Considérons les polynômes

$$\mathbf{f} = [xy^2 - 2y, x^2y + 3x].$$

Ce système admet pour seule solution (0,0) dans $\mathbb{C}$. Cependant, la réduction modulo 5 fait apparaître la composante de dimension positive définie par $xy + 3 = 0$.

Nous nous intéressons ici à une composante l'ensemble des zéros simples de $(f_1, \dots, f_n)$, et cherchons à savoir quand elle reste simple modulo un nombre premier p. Pour cela, nous allons utiliser le Nullstellensatz arithmétique du chapitre 4. Pour commencer, le lemme suivant vient mesurer la hauteurs des "variétés intermédiaires" qui interviennent dans l'énoncé de ce Nullstellensatz.

Lemme 12 *Soient* $\mathbf{f} = (f_1, \dots, f_n)$ *des polynômes de* $\mathbb{Q}[x_1, \dots, x_n]$ *de degré au plus d et de hauteur au plus h, et* $\mathcal{U}$ *une composante de dimension zéro de l'ensemble des zéros de* $(f_1, \dots, f_n)$ *dans* $\mathbb{C}^n$. *Soit* $t = \sum u_i x_i$ *un élément séparant les points de* $\mathcal{U}$, *de hauteur* h_t *et notons* $\widetilde{\mathcal{U}}$ *la variété* $\mathcal{U} \times \mathcal{V}(t - \sum u_i x_i)$ *de* $\mathbb{C}^{n+1}$, *dont les coordonnées sont* $(t, x_1, \dots, x_n)$. *Pour i entre* 1 *et* $n+1$, *notons ensuite* $\widetilde{\mathcal{U}}_i$ *l'image de la projection de* $\widetilde{\mathcal{U}}$ *sur le plan des i premières coordonnées. Alors* $\widetilde{\mathcal{U}}_i$ *est de degré inférieur à* d^n *et sa hauteur est inférieure à*

$$(n+3)d^n\big(h + h_t + 2\log(n+2)\big).$$

Démonstration. La borne sur le degré de $\widetilde{\mathcal{U}}_i$ est immédiate. Pour la hauteur, le lemma 2.5 de [124], déjà utilisé dans la première section, borne la hauteur d'une projection de $\widetilde{\mathcal{U}}$ par

$$h(\widetilde{\mathcal{U}}) + 3\deg\widetilde{\mathcal{U}}\log(n+2).$$

Ensuite la hauteur $h(\widetilde{\mathcal{U}})$ vérifie l'inégalité de Bézout 14 du chapitre 4, où l'on remplace (n,s) par $(n+1, n+1)$, et h par $h + h_t$:

$$h(\widetilde{\mathcal{U}}) \leq (n+1)d^n\big(h + h_t + 2\log(n+2)\big).$$

La somme de ces deux dernières inégalités donne le résultat. □

Proposition 44 (Premier de bonne réduction) *Avec les mêmes hypothèses et notations que dans le lemme précédent, nous supposons en outre que* $\mathcal{U}$ *est décrite par une résolution de Kronecker* $\mathcal{R}(t) = (t, q(t), \mathbf{v}(t))$ *pour un élément séparant t de hauteur* h_t *et que le déterminant jacobien* $\mathbf{jac}(\mathbf{f})$ *ne s'y annule pas.*
Alors il existe un entier e_{Red} *tel que modulo tout nombre premier p ne divisant pas* e_{Red}, *les dénominateurs de* $\mathcal{R}(t)$ *et le discriminant de q sont non nuls, et le déterminant jacobien est inversible sur les points décrits par* $\mathcal{R}(t)$ *prise modulo p. On peut prendre* e_{Red} *de hauteur inférieure à*

$$8d^{2n}(n+2)^3\big((h+h_t) + 6\log(2n+2)\big).$$

Démonstration. Il faut travailler localement, puisque l'on ne suppose pas que *tous* les zéros de $\mathbf{f}$ satisfont au critère jacobien. Nous considérons donc le système d'équations de $\mathbb{Q}[t, x_1, \dots, x_n]$

$$(g_1, \dots, g_{n+2}) = \big(q(t), (q'(t)x_i - v_i(t))_{i=1,\dots,n}, \mathbf{jac}(\mathbf{f})\big).$$

Ce système est sans zéro et satisfait aux hypothèses du Nullstellensatz arithmétique 4 du chapitre 4, en considérant les projections successives sur les plans des variables $(x_n, \dots, x_1), \dots, (x_n)$.
On peut supposer le système (g_i) donné par des équations à coefficients entiers, ceux-ci sont alors de hauteur inférieure à

$$\mathcal{H} = 2d^n\big((n+1)h + h_t + 6(n+3)\log(2n+2)\big),$$

qui est la borne sur la hauteur des paramétrisations d'une représentation de Kronecker de $\mathcal{U}$ donnée précédemment, spécialisée dans dans le cas $m = 0$, qui majora également la hauteur du déterminant jacobien.
Notons ensuite $\mathcal{V}_i$ la variété décrite par les i premières équations du système. La variété $\mathcal{V}_i$ est un cylindre construit sur la variété $\widetilde{\mathcal{U}}_i$ définie dans le lemme précédent. La proposition 15 du chapitre 4 implique que la hauteur de $\mathcal{V}_i$ est inférieure à

$$h(\widetilde{\mathcal{U}}_i) + 4(n+3)\log(n+1)\deg \widetilde{\mathcal{U}}_i :$$

on remplace les lettres (n,p) de l'énoncé par $(i,n+1-i)$, en on conclut par une petite majoration. Nous déduisons du lemme précédent une majoration sur la hauteur de $\mathcal{V}_i$ en fonction des mesures du système d'entrée :

$$h(\mathcal{V}_i) \leq (n+3)d^n\big(h + h_t + 6\log(n+2)\big).$$

Le Nullstellensatz 4 du chapitre 4 donne alors l'existence d'une constante a et de coefficients de Bézout r_i telle que

$$a = r_1 g_1 + \cdots + r_{n+2} g_{n+2},$$

où la hauteur de a est inférieure à

$$2(n+1)^2 d^n h(\mathcal{V}_i) + \big((n+2)\mathcal{H} + 2(n+1)(2n+7)\log(n+2)d^n\big)(n+1)d^n.$$

Pour tout nombre premier p ne divisant pas a, le déterminant jacobien est inversible sur les points annulant $\mathcal{R}(t)$ prise modulo p.

Il faut assurer que ni le discriminant de q ni son terme de tête ne s'annulent modulo p ; le produit b de ces deux entiers est, au signe près, le résultant de q avec sa dérivée. Les polynômes de $\mathcal{R}(t)$ sont pris à coefficients entiers donc d'après la proposition 2 du chapitre 3, b est de hauteur inférieure à $2d^n(\mathcal{H} + 3\log(2d^n))$. Nous prenons pour e_{Red} le produit ab. Mettant toutes les estimations de hauteur ensemble, après quelques réductions, on arrive à la borne de hauteur demandée. □

Dans cette proposition, la non-dégénerescence du discriminant du polynôme éliminant n'implique pas que le jacobien est inversible dans la fibre, comme montre l'exemple suivant.

Exemple Considérons dans $\mathbb{Q}[x,y]$ les polynômes

$$\mathbf{f} = [x(xy-5), y(xy+10)],$$

Dans $\mathbb{C}^2$, la seule solution de ce système est $(0,0)$ et cette solution est simple : le polynôme minimal de toute forme linéaire est t. Réduit modulo 5, ce polynôme est bien sans carré. Cependant, la réduction de ce système en $p = 5$ admet une infinité de solutions décrites par $xy = 0$. Cet ensemble de solutions contient le point $(0,0)$, en lequel le déterminant jacobien ne peut pas être inversible. C'est le même type de dégénérescence que dans l'exemple 8.4 du chapitre 8.

13.4 Étude des dégénérescences pour les systèmes paramétrés

Nous nous contentons de reprendre les résultats de la section précédente. Nous aurions également pu demander une borne mettant en jeu les polynômes $(f_1, \ldots, f_n)$ dans $\mathbb{Q}(p_1, \ldots, p_m)[x_1, \ldots, x_n]$, mais notre argumentation repose sur un Nullstellensatz arithmétique, énoncé sur (une extension finie de) $\mathbb{Q}$, et non pas sur un corps de fonctions.

Théorème 16 (Premiers de bonne réduction d'une résolution) *Soient t un élément qui sépare génériquement les points de $\mathcal{U}$, à coefficients dans $\mathbb{Z}$, de hauteur h_t, $\mathcal{R}(t)$ la résolution correspondante, et $\mathbf{p}$ une valeur des paramètres à coordonnées entières bornées par M, satisfaisant toutes les hypothèses de généricité pour t.*
Il existe un nombre entier $e_{Red}(\mathbf{p},t)$ non nul tel que pour tout nombre premier p ne divisant pas $e_{Red}(\mathbf{p},t)$:

- *les dénominateurs et le discriminant du polynôme minimal de $\mathcal{R}(t)$ spécialisée en $\mathbf{p}$ sont non nuls modulo p,*
- *le déterminant jacobien du système $\mathbf{f}$ est non nul sur les points décrits par $\mathcal{R}(t)$ modulo p.*

Dans ce cas, la réduction modulo p de la spécialisation de $\mathcal{R}(t)$ en $\mathbf{p}$ est une résolution des zéros simples du système $\mathbf{f}(\mathbf{p},.)$ pris modulo p. La hauteur de $e_{Red}(\mathbf{p},t)$ peut être prise inférieure à

$$\mathcal{H}_{Red}(h,h_t,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}},M) = 8d_{\mathbf{x}}^{2n}(n+2)^3\big(h + h_t + d_{\mathbf{p},\mathbf{x}}(\log M + \log(m+1)) + 6\log(2n+2)\big).$$

Démonstration. Le système $(f_1,\ldots,f_n)$ spécialisé au-dessus de $\mathbf{p}$ satisfait les hypothèses de la proposition 44 ci-dessus. La hauteur des équations $\mathbf{f}$ spécialisées en $\mathbf{p}$ est inférieure à

$$h + d_{\mathbf{p},\mathbf{x}}\big(\log M + \log(m+1)\big),$$

ce qui donne la première partie du résultat, après une petite majoration. Ensuite, la proposition 22 du chapitre 6 montre que sous ces conditions, on obtient tous les points simples du système $\mathbf{f}(\mathbf{p},.)$ pris modulo p. $\square$

Notations Nous appellerons $\mathcal{H}_{\mathrm{Red}}(h,h_t,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}},M)$ la borne précédente :

$$\begin{aligned}\mathcal{H}_{\mathrm{Red}} &= 8d_{\mathbf{x}}^{2n}(n+2)^3\big(h + h_t + d_{\mathbf{p},\mathbf{x}}(\log M + \log(m+1)) + 6\log(2n+2)\big)\\ &= (h+h_t)(nmd_{\mathbf{p},\mathbf{x}}d_{\mathbf{x}})^{O(1)}.\end{aligned}$$

Chapitre 14

Fonctions auxiliaires

Tout comme le chapitre 9 de la partie V, ce chapitre fait office de transition entre les estimations de hauteur obtenues dans le chapitre 13 et leur application dans les algorithmes du chapitre suivant.

Nous reprenons les points developpés dans le chapitre 9 : reconstruction rationnelle et évaluation témoin. Nous avions étudié les degrés des lieux de dégénérescence liés à ces opérations ; ici, grâce aux estimations de hauteur du chapitre précédent, nous pouvons estimer les hauteurs d'entiers "de dégénérescence".

- Dans la première section, nous introduisons un entier $e_{\text{Padé}}$, qui quantifie les nombres premiers p pour lesquels la procédure de reconstruction rationnelle d'une série entière de $\mathbb{Q}[[p_1, \ldots, p_m]]$, effectuée modulo p, est la trace du calcul sur les rationnels. La hauteur de cet entier est explicitement bornée par une fonction notée $\mathcal{H}_{\text{Padé}}$, qui ne dépend que de la mesure syntaxique du système $(f_1, \ldots, f_n)$.
- Dans la deuxième section, nous retrouvons la fonction `Temoin`. Dans le chapitre 9, nous l'utilisions pour faire des calculs spécialisés ; maintenant, nous allons faire des calculs modulaires. Nous introduisons donc un entier baptisé $e_{\text{Témoin}}$ qui quantifie les premiers appelés *bons témoins* pour lesquels l'utilisation de cette fonction ne provoquera pas d'erreur. La hauteur de cet entier est elle aussi bornée en fonction de la mesure syntaxique du système $(f_1, \ldots, f_n)$.

Sommaire

14.1 Reconstruction Rationnelle

Dans un premier temps, nous quantifions la probabilité d'échec de la procédure de reconstruction rationnelle d'une résolution univariée, comme nous l'avons fait dans la section 9.1 du chapitre 9. Cette fois, nous nous intéressons à estimer la taille des nombres premiers p pour lesquels le calcul modulo p est la trace du calcul sur les rationnels. La première partie de l'énoncé de la proposition suivante est tirée de la section 9.1.2 du chapitre 9.

Proposition 45 *Soient t un élément séparant générique à coefficients entiers de hauteur bornée par h_t, $\mathcal{R}(t)$ la résolution correspondante et $\mathbf{p}$ un point qui satisfait toutes les hypothèses de généricité pour t.*
Supposons donnés les développements de Taylor en $\mathbf{p}$ de la résolution $\mathcal{R}(t)$ à l'ordre $\mathcal{D}$, où $2\mathcal{D}(t) \leq \mathcal{D} \leq 4\mathcal{D}(t)$. Alors les changements de variables γ qui ne permettent pas la reconstruction de tous les coefficients de $\mathcal{R}(t)$ sont inclus dans une hypersurface de $\mathbb{C}^{m-1}$ degré au plus

$$16n(4d_{\mathbf{p},\mathbf{x}}^n + 1)d_{\mathbf{p},\mathbf{x}}^{2n}d_{\mathbf{x}}^n.$$

Supposons que $\mathbf{p}$ est à coordonnées entières majorées par M. Alors pour tout point γ chanceux de coordonnées entières inférieures à M, il existe un entier $e_{Padé}(\mathbf{p},t,\gamma)$ non nul tel que pour tout nombre premier p ne divisant pas $e_{Red}(\mathbf{p},t)e_{Padé}(\mathbf{p},t,\gamma)$, la procédure de reconstruction modulo p est la trace du calcul sur les rationnels. On peut majorer la hauteur de $e_{Padé}(\mathbf{p},t,\gamma)$ par

$$8n(4d_{\mathbf{p},\mathbf{x}}^n + 1)^2 d_{\mathbf{p},\mathbf{x}}^{3n}\big((n+1)h + h_t + 2\log M + 6(n+2m+5)\log(2n+m+2)\big).$$

Démonstration. Nous n'avons à montrer que la seconde assertion. Les coefficients que l'on cherche à reconstruire sont les développements de fractions de degré inférieur à $\mathcal{D}(t)$, et de hauteur inférieure à

$$2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + h_t + \log M + 6(n+2m+4)\log(2n+m+2)\big).$$

On obtient cette majoration à partir de la borne de hauteur donnée dans le chapitre 13, en majorant la croissance de hauteur due au déplacement de l'origine de coordonnées en $\mathbf{p}$.

En conséquence, pour chaque point γ chanceux de coordonnées inférieures à M, d'après la proposition 5 du chapitre 5, il existe un entier $e_{\text{Padé}}(\mathbf{p},t,\gamma)$ non nul, de hauteur bornée par

$$8(4d_{\mathbf{p},\mathbf{x}}^n + 1)^2 d_{\mathbf{p},\mathbf{x}}^{2n}\big((n+1)h + h_t + 2\log M + 6(n+2m+5)\log(2n+m+2)\big)$$

tel que si p ne divise pas $e_{\text{Red}}(\mathbf{p},t)e_{\text{Padé}}(\mathbf{p},t,\gamma)$, la reconstruction modulo p est la trace de la reconstruction sur les rationnels. On obtient l'existence de l'entier $e_{\text{Padé}}(\mathbf{p},t,\gamma)$ et l'inégalité sur sa hauteur en considérant tous les coefficients des polynômes de la résolution. □

Remarque Si (γ,p) est un point chanceux, on est en particulier assuré que les fractions que l'on reconstruit ne se simplifient pas modulo p.

Notations Nous appellerons $\mathcal{H}_{\text{Padé}}(h,h_t,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}},M)$ la borne de hauteur introduite ci-dessus :

$$\begin{aligned}\mathcal{H}_{\text{Padé}} &= 8n(4d_{\mathbf{p},\mathbf{x}}^n + 1)^2 d_{\mathbf{p},\mathbf{x}}^{3n}\big((n+1)h + h_t + 2\log M + 6(n+2m+5)\log(2n+m+2)\big)\\ &= (h+h_t)(nmd_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^n \log M)^{O(1)}.\end{aligned}$$

14.2 Critère d'arrêt modulaire

Nous reprenons maintenant les considérations de la section 9.2 du chapitre 9, en les étendant au cadre du calcul modulaire.

Nous considérons une résolution univariée $\mathcal{R}'(t) = (t = \sum u_i x_i, q'(t), \mathbf{v}'(t))$, à coefficients dans le corps $\mathbb{Q}(p_1, \ldots, p_m)$ et $\mathcal{J}$ l'idéal $(p_1 - p_{1,1}, \ldots, p_m - p_{1,m}) + (p)$, pour une valeur $(p_{1,1}, \ldots, p_{1,m})$ de $\mathbb{Z}^m$ et un nombre premier p. À nouveau, nous voulons tester si $\mathcal{R}'(t)$ est la résolution des zéros simples génériques du système $(f_1, \ldots, f_n)$. Nous procédons en évaluant le système $\mathbf{f}$ sur la spécialisation de $\mathcal{R}'(t)$ en $\mathcal{J}$. Ce test s'effectue par l'appel à la fonction `Temoin` présentée dans la section 9.2 du chapitre 9, avec pour paramètres $(\mathbf{f}, \mathcal{J}, \mathcal{R}'(t))$.

Utilisation. Cette fonction sera utilisée dans le chapitre suivant, pour décider de l'arrêt du processus de remontée de Newton. La résolution $\mathcal{R}'(t)$ sera candidate à être la résolution générique des points simples de $\mathbf{f}$. Si le test répond "vrai", nous décrèterons que $\mathcal{R}'(t)$ est bien la résolution générique ; il faut quantifier le risque d'erreur provoqué par ce choix.

Les polynômes q' et v_i' sont dans $\mathbb{Q}(p_1, \ldots, p_m)[t]$. Nous supposons que le polynôme q' a pour degré $\mathcal{D}_{\mathbf{x}}$, puis que les coefficients de $\mathcal{R}'(t)$ sont des quotients de polynômes de $\mathbb{Z}[p_1, \ldots, p_m]$ de degré au plus $\mathcal{D}$ et de hauteur au plus H. Supposons également que t est élément séparant générique pour $\mathcal{U}$ et que $\mathcal{R}'(t)$ est différent de la résolution générique $\mathcal{R}(t)$. Sous ces hypothèses, nous appellerons *bons témoins* pour $(\mathcal{R}'(t), \mathcal{R}(t))$ les couples $(\mathbf{p}, p)$, où $\mathbf{p}$ est à coordonnées entières et satisfait toutes les hypothèses de généricité pour t, et p est un nombre premier, tels que les spécialisations de $\mathcal{R}'(t)$ et $\mathcal{R}(t)$ en $(\mathbf{p}, p)$ sont possibles et ne coïncident pas.

Pour que le couple $(\mathbf{p}, p)$ soit bon témoin, il est nécessaire que les spécialisations en $\mathbf{p}$ de $\mathcal{R}'(t)$ et $\mathcal{R}(t)$ soit distinctes sur $\mathbb{Q}$. Avec le vocabulaire du chapitre 9, cela signifie que le point $\mathbf{p}$ doit lui-même être un bon témoin. Sous ces conditions, on peut alors chercher à quantifier les nombres premiers pour lesquels le couple $(\mathbf{p}, p)$ reste bon témoin. C'est l'objet de la proposition suivante.

Proposition 46 *Pour tout point $\mathbf{p}$ bon témoin pour $(\mathcal{R}'(t), \mathcal{R}(t))$, à coordonnées entières bornées par M, il existe un entier $e_{Témoin}(\mathbf{p}, t, \mathcal{R}'(t))$ non nul tel que pour tout nombre premier p, si p ne divise pas $e_{Témoin}(\mathbf{p}, t, \mathcal{R}'(t))$ alors $(\mathbf{p}, p)$ est bon témoin pour $(\mathcal{R}'(t), \mathcal{R}(t))$.*

Supposons que le degré $\mathcal{D}$ des coefficients de $\mathcal{R}'(t)$ est inférieur à $d^n_{\mathbf{p},\mathbf{x}}$. Sous ces hypothèses, l'entier $e_{Témoin}(\mathbf{p}, t, \mathcal{R}'(t))$ peut être pris de hauteur inférieure à

$$2d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + h_t + 2\log M + 6(n+2m+5)\log(2n+m+2)\big) + H + 1.$$

Démonstration. Les spécialisations de $\mathcal{R}'(t)$ et $\mathcal{R}(t)$ sont supposées distinctes en $\mathbf{p}$, et $(\mathbf{p}, p)$ est un mauvais témoin si les réductions de ces deux résolutions modulo p sont les mêmes.
Si $\mathbf{p}$ est à coefficients entiers bornés par M, alors l'évaluation d'un coefficient de $\mathcal{R}'(t)$ en $\mathbf{p}$ est un rationnel de hauteur au plus $H + \mathcal{D}(\log M + \log(m+1))$. De même, d'après le théorème 12 du chapitre 13, les coefficients de la résolution $\mathcal{R}(t)$ sont des quotients de polynômes de $\mathbb{Z}[p_1, \ldots, p_m]$ de hauteur au plus

$$2d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + h_t + 6(n+2m+3)\log(2n+m+2)\big),$$

donc leurs évaluations en $\mathbf{p}$ sont des rationnels de hauteur au plus

$$2d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + h_t + \log M + 6(n+2m+4)\log(2n+m+2)\big).$$

Soient c et c' les coefficients d'un même monôme, respectivement dans les spécialisations de $\mathcal{R}'(t)$ et $\mathcal{R}(t)$ en $\mathbf{p}$, supposés distincts. Si le premier p est de mauvaise réduction, alors $c - c' = 0 \mod p$. Le numérateur de $c - c'$ est non nul, et a pour hauteur la borne annoncée. □

Soit $\mathbf{p}$ une valeur des paramètres satisfaisant toutes les hypothèses de généricité et supposons que $\mathbf{p}$ est un bon témoin pour $(\mathcal{R}'(t),\mathcal{R}(t))$. Soit ensuite p un nombre premier ne divisant pas l'entier $e_{\mathrm{Red}}(\mathbf{p},t)$ défini dans la dernière section du chapitre précédent. Alors $(\mathbf{p},p)$ est un mauvais témoin pour $(\mathcal{R}'(t),\mathcal{R}(t))$ si et seulement si, modulo p, $\mathcal{R}'(t)$ annule le système $\mathbf{f}(\mathbf{p},.)$ mais pas le déterminant jacobien, donc si et seulement si l'appel à la fonction `Temoin` répond "vrai".

Chapitre 15

Calculs modulaires

Résumé

Nous donnons dans ce chapitre un algorithme probabiliste, qui calcule une résolution générique des zéros simples du système $(f_1,\ldots,f_n)$, par des techniques modulaires. Une première étape est le calcul des zéros simples génériques de $(f_1,\ldots,f_n)$ modulo un nombre premier p_1. Par une seconde application de l'opérateur de Newton, nous remontons les coefficients de cette résolution modulo les puissances successives de p_1, pour finalement reconstruire une résolution à coefficients dans $\mathbb{Q}(p_1,\ldots,p_m)$.

À partir des bornes de hauteur données dans les deux chapitres précédents, nous contrôlons la taille des nombres premiers intervenant dans l'algorithme, ce qui permet d'obtenir à la fois :

– une complexité *binaire* quasiment linéaire en la hauteur de la sortie et

– une probabilité d'erreur du processus de remontée inférieure à 1/2.

Dans la dernière section, nous donnons un procédé probabiliste de vérification d'un candidat à être la résolution recherchée. L'analyse est dans le même esprit que les autres estimations de probabilités de ce chapitre, mais est indépendante d'un procédé de calcul.

Sommaire

Hypothèses et résultats

Pour énoncer le résultat principal de cette partie, nous utilisons la notation suivante : si p_1 est un nombre premier et $\mathbf{p}_1$ une valeur des paramètres à coordonnées entières, $\mathcal{C}_{\text{Zéro}}(\mathbf{f},\mathbf{p}_1,p_1)$ désigne le coût de la résolution du système $\mathbf{f}(\mathbf{p}_1,.)$ pris modulo p_1.

Théorème 17 (Complexité du calcul de la résolution générique sur les rationnels) *Soit $h_1 = 10^4 hn^2(n+m+10)d_{\mathbf{p},\mathbf{x}}^{5n+2}$. On peut trouver un élément séparant générique t et calculer la résolution générique $\mathcal{R}(t)$ en*

$$\mathcal{C}_{Zéro}(\mathbf{f},\mathbf{p}_1,p_1) + O_{log}\big(\mathcal{M}(\mathcal{D}_\mathbf{x})(n^\Omega + nL + nm)\mathcal{M}_s(4\mathcal{D}(t),m)\mathcal{M}_\mathbb{N}(\mathcal{H}(t) + \log h + mn\log d_{\mathbf{p},\mathbf{x}})\big)$$

opérations binaires, où p_1 est un nombre premier compris entre h_1 et $2h_1$, $\mathbf{p}_1$ une valeur des paramètres à coordonnées entières et t un élément séparant génériquement les points de $\mathcal{U}$, le tout avec une probabilité d'erreur inférieure à $1/2$.

Ce résultat signifie que le coût de la remontée est :
- linéaire en la complexité d'évaluation L,
- polynomial en $(n,m,\log d_{\mathbf{p},\mathbf{x}},\log h)$,
- quasiment linéaire en la hauteur de la sortie.

L'entier h_1 est une borne largement surestimée, mais plus lisible, sur un entier $\mathcal{H}_1$ qui sera défini dans la section 15.2, et qui contrôle en partie des calculs modulaires. Le coût des calculs modulaires est polynomial en le nombre de chiffres de h_1, donc en son logarithme. Ce dernier est d'ordre $\log h + \log m + n\log d_{\mathbf{p},\mathbf{x}}$, il explique en partie le terme correctif dans la formule $\mathcal{M}_\mathbb{N}(\mathcal{H}(t) + \log h + mn\log d_{\mathbf{p},\mathbf{x}})$.

La preuve du théorème précédent est donnée dans l'avant-dernière section du chapitre. On procède par calcul modulaire, et le gros du travail est l'estimation de bornes sur les nombres premiers à utiliser. Ces estimations arithmétiques se rajoutent la quantification des dégénérescences géométriques déjà rencontrées dans la partie précédente, ce qui donne un aspect assez technique au propos. Tout comme son analogue le chapitre 10 de la partie V, ce chapitre se termine par une procédure probabiliste de vérification d'une résolution, indépendante de l'algorithme de calcul utilisé.

S'il existe une composante $\mathcal{U}'$ des zéros simples génériques de $(f_1,\ldots,f_n)$ dont on sait identifier la trace (par exemple par intersection avec une équation supplémentaire) au-dessus d'une valeur générique des paramètres, on peut en calculer une représentation univariée de manière analogue au cas général, ce qui permet en particulier de travailler en dehors d'une hypersurface donnée. La complexité du calcul ne prend en compte que la taille de la représentation de la composante cherchée et l'analyse des probabilités d'erreur n'est pas modifiée.

15.1 Description de l'algorithme

Nous allons pousser le principe de remontée de Newton, en enchaînant remontée des séries formelles modulo un nombre premier, puis remontée des entiers, suivant l'exemple de la section 6.6.3 du chapitre 6. Notre nouvel algorithme se présente ainsi :

- On choisit deux nombres premiers p_1 et p_2 et trois valeurs des paramètres $\mathbf{p}_1$, $\mathbf{p}_2$ et $\mathbf{p}_3$ dans $\mathbb{Z}^m$.

- Dans un premier temps, on calcule les zéros simples du système $\mathbf{f}(\mathbf{p}_1,.)$ modulo p_1. On en déduit la résolution générique modulo p_1 par l'algorithme présenté dans le chapitre 10, en utilisant la valeur témoin $\mathbf{p}_2$ pour arrêter la remontée.
- On passe à la résolution générique sur les rationnels par un nouvelle application de l'opérateur de Newton, en remontant les entiers modulo les puissances successives de p_1. La résolution du système $\mathbf{f}(\mathbf{p}_3,.)$, modulo le nombre premier p_2, sert de témoin pour arrêter la remontée.

Les bornes, notées $\mathcal{H}_1,\mathcal{H}_2,M,M'$, qui contrôlent la taille des nombres premiers p_1 et p_2 et les choix de $\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3$, sont données dans la suite du chapitre.

Fig. 15.1 – *Résolution de la fibre générique sur* $\mathbb{Q}$

procedure `ResolutionFibreGeneriqueRationnelle`($\mathbf{f}$)

\# **Entree**
\# le système $\mathbf{f}$
\# **Sortie**
\# $\mathcal{R}(t) = (t,q,\mathbf{v})$ une résolution des zéros de $\mathbf{f}$ à coefficients dans $\mathbb{Q}(p_1,\ldots,p_m)$

\# on remonte les séries modulo un nombre premier p_1 :
$p_1,\mathbf{p}_1,\mathbf{p}_2,\gamma,\mathcal{R} \leftarrow$ `ResolutionFibreGeneriqueModulaire`($\mathbf{f}$);
\# $(K,M',\mathcal{H}_2)$ sont définis dans les sections 15.3 et 15.5
$\mathbf{p}_3 \leftarrow$ Paramètre aléatoire de $\{1,2,\ldots,20M'\}^m$;
$p_2 \leftarrow$ nombre premier compris entre $\mathcal{H}_2$ et $2\mathcal{H}_2$;
\# on remonte les entiers sur ces séries :
return `RemonteeNewton`($\mathbf{f}$,(p_1),$\mathcal{R}$,`Arret`($\mathbf{f}$,$\mathbf{p}_3$,p_2,γ),K)
end;

Dans la prochaine section, nous revenons sur l'algorithme de calcul des zéros génériques, effectué modulo un nombre premier. Nous nous intéresserons ensuite au processus de remontée des entiers de cette résolution modulaire. Nous en bornons le nombre d'étapes dans la section 15.3, détaillons le critère d'arrêt dans la section 15.4, pour rassembler toutes ces estimations dans la section 15.5.

15.2 Résolution modulaire de la fibre générique

Appelons `ResolutionFibreGeneriqueModulaire` l'algorithme `ResolutionFibreGenerique` 10.1 du chapitre 10, effectué modulo un nombre premier p_1. Dans le reste de cette section, nous donnons la complexité et la probabilité d'erreur de cette première phase du calcul. Pour cela, on reprend les différentes étapes de l'algorithme `ResolutionFibreGenerique`, en imposant des conditions supplémentaires de non-dégénérescence. Les paramètres sur lesquelles ces conditions vont porter sont :

- les valeurs des paramètres $\mathbf{p}_1$ et $\mathbf{p}_2$,
- le nombre premier p_1,
- le changement de variables γ.

En conclusion, nous définirons l'entier $\mathcal{H}_1$ qui contrôle le choix du nombre premier p_1, et l'entier M qui contrôle le choix des paramètres $(\mathbf{p}_1,\mathbf{p}_2)$. Ces entiers sont définis uniquement en fonction des données syntaxiques du problème $(h,n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}})$.

FIG. 15.2 – *Remontée des séries modulo un nombre premier*

procedure `ResolutionFibreGeneriqueModulaire`($\mathbf{f}$)

\# **Entree**
\# le système $\mathbf{f}$
\# **Sortie**
\# $\mathbf{p}_1,\mathbf{p}_2$: Paramètres aléatoires de Ω^m;
\# γ : Changement de variables aléatoire de Γ^{m-1};
\# p_1 nombre premier inférieur a $\mathcal{H}_1$
\# $(t,q,\mathbf{v})$ une résolution des zéros de $\mathbf{f}$ à coefficients dans $\mathbb{Z}/p_1\mathbb{Z}(p_1,\ldots,p_m)$

\# $(M,\mathcal{H}_1)$ sont définis dans le dernier paragraphe ci-dessous
$\Omega,\Gamma \leftarrow \{1,2,\ldots,20M\}$;
$p_1 \leftarrow$ premier compris entre $\mathcal{H}_1$ et $2\mathcal{H}_1$;
\# on calcule dans $\mathbb{Z}/p_1\mathbb{Z}$:
return p_1,`ResolutionFibreGenerique`($\mathbf{f}$,Ω,Γ);
end;

Résolution de la fibre spécialisée. D'après la section 1.1.2 du chapitre 1, il existe un élément t de hauteur inférieure à $n^2 \log d_{\mathbf{x}}$ séparant les points de $\mathcal{U}$. Il suffit que la fibre au-dessus de $\mathbf{p}_1$ remplisse toutes les conditions de généricité pour t. D'après la section 8.4 du chapitre 8, les mauvais points $\mathbf{p}_1$ sont enfermés dans une hypersurface de $\mathbb{C}^m$ de degré au plus $d_{\mathbf{p},\mathbf{x}}^n(2d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}})$.
D'après la section 13.4 du chapitre 13, il existe un entier $e_{\mathrm{Red}}(\mathbf{p}_1,t)$ tel que si le nombre premier p_1 ne divise pas $e_{\mathrm{Red}}(\mathbf{p}_1,t)$, la procédure `Resolution` renvoie la spécialisation en $(\mathbf{p}_1,p_1)$ d'une résolution $\mathcal{R}(t)$ des points de $\mathcal{U}$. Dans ce cas, l'algorithme de remontée de Newton appliqué modulo p_1 calcule la réduction modulo p_1 du développement de Taylor de $\mathcal{R}(t)$ en $\mathbf{p}_1$.

Reconstruction rationnelle. Dans la section 14.1 du chapitre 14, nous avons quantifié la probabilité d'échec de la procédure de reconstruction rationnelle, paramétrée par un changement de variables γ, et défini l'entier $e_{\text{Padé}}(\mathbf{p}_1,t,\gamma)$ assurant la non-dégénérescence modulaire :

- Les changements de variables non chanceux pour la reconstruction sur les rationnels sont enfermés dans une hypersurface de $\mathbb{C}^{m-1}$ de degré au plus $16n(4d_{\mathbf{p},\mathbf{x}}^n + 1)d_{\mathbf{p},\mathbf{x}}^{2n}d_{\mathbf{x}}^n$.
- Si la reconstruction est possible sur $\mathbb{Q}$, si γ est un changement de variables chanceux et si p_1 ne divise pas $e_{\mathrm{Red}}(\mathbf{p}_1,t)e_{\text{Padé}}(\mathbf{p}_1,t,\gamma)$, alors la reconstruction est possible modulo p_1, et c'est la trace du calcul sur les rationnels.

Comparaison à une valeur témoin. Si la reconstruction rationnelle est possible, le test d'arrêt repose sur l'évaluation du système $\mathbf{f}$ sur la reconstruction spécialisée en $\mathbf{p}_2$, le tout modulo p_1. Le succès est assuré par les conditions suivantes :

- Il faut que le point $\mathbf{p}_2$ remplisse toutes les conditions de généricité pour t. Les mauvais points $\mathbf{p}_2$ sont inclus dans une hypersurface de $\mathbb{C}^m$ de degré au plus
$$d_{\mathbf{p},\mathbf{x}}^n(2d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}}).$$
- Il faut que p_1 ne divise pas l'entier discriminant $e_{\mathrm{Red}}(\mathbf{p}_2,t)$: dans ce cas, les seuls zéros simples du système $\mathbf{f}(\mathbf{p}_2,.)$ pris modulo p_1 sont donnés par la spécialisation de la résolution générique $\mathcal{R}(t)$ en $\mathbf{p}_2$, modulo p_1.
- Dans ce cas, le test d'arrêt réussit si $\mathbf{p}_2$ est un bon témoin pour toutes les reconstructions intermédiaires. D'après le corollaire 39 du chapitre 10, pour tout nombre premier p_1, les mauvais témoins sont enfermés dans une hypersurface de $\overline{\mathbb{F}_{p_1}}^m$ de degré au plus
$$d_{\mathbf{p},\mathbf{x}}^n(\lceil \log_2(d_{\mathbf{p},\mathbf{x}}^n)\rceil + 2).$$

Complexité et probabilité d'erreur. Nous rassemblons toutes les conditions rencontrées jusqu'ici dans le lemme suivant :

Lemme 13 *Les conditions suivantes assurent que la sortie de l'algorithme* `ResolutionFibreGeneriqueModulaire` *est la réduction modulo p_1 d'une résolution des zéros simples de $\mathcal{U}$.*

- *$\mathbf{p}_1$ et $\mathbf{p}_2$ remplissent toutes les conditions de généricité et l'élément séparant t choisi pour $\mathbf{p}_1$ est aussi séparant pour $\mathbf{p}_2$.*
- *$\mathbf{p}_2$ est un bon témoin pour la reconstruction modulo p_1.*
- *γ est un changement de variables chanceux pour le dernier calcul de reconstruction rationnelle.*
- *p_1 ne divise pas $e_{Red}(\mathbf{p}_1,t)e_{Red}(\mathbf{p}_2,t)e_{Padé}(\mathbf{p}_1,t,\gamma)$.*

Notations Nous réutilisons la grandeur M définie dans la section 10.5 du chapitre 10 : $M = M(n,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}})$ est la somme de :

- $2d_{\mathbf{p},\mathbf{x}}^n(2d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}})$,
- $d_{\mathbf{p},\mathbf{x}}^n(\lceil \log_2(d_{\mathbf{p},\mathbf{x}}^n)\rceil + 2)$,
- $16n(4d_{\mathbf{p},\mathbf{x}}^n + 1)d_{\mathbf{p},\mathbf{x}}^{2n}d_{\mathbf{x}}^n$.

Nous prenons pour Ω et Γ l'ensemble $\{1,2,\ldots,\ell M\}$, pour un paramètre ℓ qui sera fixé ultérieurement à une valeur entière de l'ordre de 20. Celui-ci n'apparaît pas dans les estimations de complexité asymptotique, car ce n'est pas une variable ; par contre, son utilisation facilite l'écriture des estimations de probabilité. Ensuite, nous notons e_{Red} et $e_{\mathrm{Padé}}$ les maxima respectifs de $e_{\mathrm{Red}}(\mathbf{p},t)$ et $e_{\mathrm{Padé}}(\mathbf{p},t,\gamma)$, pour $\mathbf{p}$ décrivant Ω^m, γ décrivant Γ^{m-1} et t élément séparant de la fibre générique de hauteur inférieure à $n^2 \log d_{\mathbf{x}}$.

Définition 15 *Nous appellerons $\mathcal{H}_1 = \mathcal{H}_1(h,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}})$ l'entier défini par :*

$$\begin{aligned}\mathcal{H}_1(h,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}}) &= \ell \max\{M(n,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}}),\\ &\quad (2\mathcal{H}_{Red} + \mathcal{H}_{Padé})(h,n^2\log d_{\mathbf{x}},n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}},\ell M(n,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}}))\}\end{aligned}$$

C'est un majorant de $\max\{\ell M,\ell \log e_{Red}^2 e_{Padé}\}$ et il vérifie l'inégalité :

$$\mathcal{H}_1 = h(mnd_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^n)^{O(1)}.$$

L'entier h_1 défini dans l'introduction est un majorant grossier de $\mathcal{H}_1$.

Notons $\mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}$ l'ensemble des nombres premiers compris entre $\mathcal{H}_1$ et $2\mathcal{H}_1$. Cet ensemble comporte au moins $\mathcal{H}_1/(2\log\mathcal{H}_1)$ éléments d'après la proposition 3.3.2 du chapitre 3. Soit $\mathcal{E}$ l'ensemble des évènements

$$\Omega^m \times \Omega^m \times \Gamma^{m-1} \times \mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}.$$

Dans la proposition suivante, nous tirons de manière uniforme les paramètres $(\mathbf{p}_1,\mathbf{p}_2,\gamma,p_1)$ dans $\mathcal{E}$, et supposons que l'élément séparant t est une fonction de $\mathbf{p}_1$.

Proposition 47 (Probabilité d'erreur du calcul modulaire) *La probabilité de choisir un quadruplet $(\mathbf{p}_1,\mathbf{p}_2,\gamma,p_1)$ dans $\mathcal{E}$ qui fait échouer l'algorithme de reconstruction de la fibre générique modulo p_1 est inférieure à*

$$\frac{6}{\ell}.$$

Démonstration. Étudions les effets de chacune des conditions données dans le lemme précédent. Systématiquement, nous utilisons la version géométrique du lemme de Zippel-Schwartz ou son pendant arithmétique pour estimer le nombre de points de $\mathcal{E}$ qui peuvent tomber dans les cas de dégénérescence (voir la dernière section du chapitre 3 pour ces lemmes d'évitement). Rappelons enfin que l'entier M a été choisi comme un majorant de l'ensemble des degrés des hypersurfaces à eviter. Cela explique son apparition dans les trois premiers points ci-dessous.

- D'après la section 15.2, la première condition discrimine au plus $d_{\mathbf{p},\mathbf{x}}^n(2d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}})|\Omega|^{m-1}$ valeurs de $\mathbf{p}_1$ et autant de valeurs de $\mathbf{p}_2$, soit au plus
$$2M|\Omega|^{m-1}|\Omega|^m|\Gamma|^{m-1}|\mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}|$$
points de $\mathcal{E}$.
- D'après la section 15.2, à $\mathbf{p}_1$ et p_1 fixés, la deuxième condition supprime au plus
$$d_{\mathbf{p},\mathbf{x}}^n(\lceil\log_2(d_{\mathbf{p},\mathbf{x}}^n)\rceil + 2)|\Omega|^{m-1}$$
valeurs de $\mathbf{p}_2$ parmi les points de l'image de Ω^m modulo p_1. La majoration $\mathcal{H}_1 \geq \ell M$ implique que pour tout p_1, les images modulo p_1 des points de Ω sont toutes distinctes. Ainsi, cette condition supprime au plus
$$M|\Omega|^{m-1}|\Omega|^m|\Gamma|^{m-1}|\mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}|$$
points de $\mathcal{E}$.
- D'après la section 15.2, pour tout $\mathbf{p}_1$, la troisième condition discrimine au plus $16n(4d_{\mathbf{p},\mathbf{x}}^n + 1)d_{\mathbf{p},\mathbf{x}}^{2n}d_{\mathbf{x}}^n|\Gamma|^{m-2}$ valeurs des changements de variables, soit au plus
$$M|\Omega|^m|\Omega|^m|\Gamma|^{m-2}|\mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}|$$
points de $\mathcal{E}$.
- Pour toute valeur de $(\mathbf{p}_1,\mathbf{p}_2,\gamma)$ non encore éliminée, d'après la section 15.2, la dernière condition supprime au plus $\log e_{\mathrm{Red}}^2 e_{\mathrm{Padé}}/\log\mathcal{H}_1$ nombres premiers, soit au plus
$$\frac{|\Omega|^m|\Omega|^m|\Gamma|^{m-1}\log e_{\mathrm{Red}}^2 e_{\mathrm{Padé}}}{\log\mathcal{H}_1}$$
points de $\mathcal{E}$.

Au total, on a enlevé au plus

$$3M|\Omega|^{m-1}|\Omega|^m|\mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}||\Gamma|^{m-1} + \frac{|\Omega|^m|\Omega|^m|\Gamma|^{m-1}\log e_{\mathrm{Red}}^2 e_{\mathrm{Padé}}}{\log \mathcal{H}_1} + M|\Omega|^m|\Omega|^m|\Gamma|^{m-2}|\mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}|$$

points de $\mathcal{E}$. Cela représente une proportion au plus égale à

$$3\frac{M}{|\Omega|} + \frac{\log e_{\mathrm{Red}}^2 e_{\mathrm{Padé}}}{\log \mathcal{H}_1|\mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}|} + \frac{M}{|\Gamma|}.$$

D'après la minoration sur le cardinal de $\mathcal{P}_{\mathcal{H}_1,2\mathcal{H}_1}$, cette proportion est inférieure à

$$3\frac{M}{|\Omega|} + 2\frac{\log e_{\mathrm{Red}}^2 e_{\mathrm{Padé}}}{\mathcal{H}_1} + \frac{M}{|\Gamma|}.$$

Finalement, on peut la majorer par

$$\frac{6}{\ell}.$$

□

Proposition 48 (Complexité du calcul modulaire) *En cas de succès, la complexité de l'algorithme de reconstruction de la fibre générique modulo p_1, est*

$$\mathcal{C}_{Zéro}(\mathbf{f}) + O_{log}\big((n^\Omega + nL + nm)\mathcal{M}(\mathcal{D}_\mathbf{x})\mathcal{M}_s(4\mathcal{D}(t),m)\mathcal{M}_\mathbb{N}(\log h + n\log md_{\mathbf{p},\mathbf{x}})\big)$$

opérations binaires.

Démonstration. On applique le résultat principal du chapitre 10, pour obtenir une première estimation de

$$\mathcal{C}_{\mathrm{Zéro}}(\mathbf{f}) + O_{\mathrm{log}}\big((n^\Omega + nL + nm)\mathcal{M}(\mathcal{D}_\mathbf{x})\mathcal{M}_s(4\mathcal{D}(t),m)\mathcal{M}_\mathbb{N}(\log p_1)\big)$$

opérations binaires. Ensuite, la borne donnée sur $\mathcal{H}_1$ dans la définition 15 ci-dessus permet de conclure. □

15.3 Remontée des entiers par Newton-Hensel

Arrivés à ce stade, si aucun choix malchanceux n'a été fait, nous disposons de la résolution des zéros simples de la fibre générique modulo le nombre premier p_1. Nous cherchons maintenant à obtenir cette résolution sur les rationnels, en appliquant la fonction `RemonteeNewton` avec les arguments suivants :

- On travaille modulo les puissances successives de l'idéal (p_1) de $\mathbb{Z}[[p'_1,\dots,p'_m]]$, après avoir effectué le changement de variables $p'_k = p_k - p_{1,k}$.
- Le point initial est la résolution qui a été produite par l'algorithme `ResolutionFibreGeneriqueModulaire`, dont les fractions sont développées à l'ordre qui a suffi pour les reconstruire ; cet ordre est inférieur à $4d_{\mathbf{p},\mathbf{x}}^n$.
- Le critère d'arrêt est décrit dans la section suivante.
- Le nombre maximal d'itérations est fixé par l'entier K défini ci-dessous.

À la $i$ème étape, on traite des nombres de $2^i \log p_1$ chiffres, donc le coût d'une opération modulo $p_1^{2^i}$ est $\mathcal{M}_{\mathbb{N}}(2^i \log p_1)$ opérations binaires. Cela mène à l'estimation de complexité suivante :

Proposition 49 *Le coût de la ième étape de la remontée de Newton est*

$$O\big((n^{\Omega} + nL)\mathcal{M}_s(4\mathcal{D}(t),m)\mathcal{M}(\mathcal{D}_{\mathbf{x}})\mathcal{M}_{\mathbb{N}}(2^i \log p_1)\big)$$

opérations binaires.

Les résultats suivants viennent mesurer la puissance du nombre premier p_1 à laquelle il faut monter pour pouvoir reconstruire les rationnels de la résolution. Rappelons que nous appelons $\mathcal{H}(t)$ la hauteur de la résolution des points simples de la fibre générique pour l'élément séparant t. Cette grandeur est définie dans le chapitre 13, où nous en donnons une majoration explicite par une fonction notée $\mathcal{H}(h,h_t,n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}})$.

Lemme 14 *Soit $f = n/d$ dans $\mathbb{Q}(p_1,\dots,p_m)$. On suppose que les hauteurs de tous les coefficients de n et d sont bornées par h.*
Soit p_1 un nombre premier ne divisant le dénominateur d'aucun coefficient et posons $k = \lceil \frac{2h+1}{\log p_1} \rceil$. Si on connaît les réductions de tous les coefficients de f modulo p^k, alors on peut tous les reconstruire.

Démonstration. Pour pouvoir reconstruire une fraction à partir de sa réduction modulo p^k, on normalise cette fraction en mettant un des coefficients de son dénominateur à 1. Il suffit alors que la hauteur de $p^k/2$ soit supérieure à deux fois celle du numérateur et du dénominateur. La valeur de k annoncée remplit cette condition. □

Proposition 50 *Si $(\mathbf{p}_1,p_1)$ est un choix chanceux, il suffit de*

$$\lceil \log_2(\lceil \frac{\mathcal{H}(t)+1}{\log p_1} \rceil) \rceil$$

étapes de remontée de Newton pour pouvoir reconstruire tous les coefficients des fractions intervenant dans la résolution $\mathcal{R}(t)$.

Démonstration. C'est une application du lemme précédent, puisque chaque coefficient de la résolution générique est un quotient de deux polynômes de $\mathbb{Z}[p_1,\dots,p_m]$ de hauteur au plus $\mathcal{H}(t)$. □

Proposition 51 *Soit K le nombre d'étapes de remontée. Alors nous avons l'inégalité*

$$\begin{aligned} K &\leq \lceil \log_2(\lceil \frac{\mathcal{H}(h,n^2\log d_{\mathbf{x}},n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}})+1}{\log \mathcal{H}_1} \rceil) \rceil \\ &= O(\log hn^3 \log n \log md_{\mathbf{p},\mathbf{x}} \log d_{\mathbf{x}}). \end{aligned}$$

Pour tout entier p_1 compris entre $\mathcal{H}_1$ et $2\mathcal{H}_1$, K et p_1 vérifient les inégalités :

$$\begin{aligned} 2^{K+1}\log p_1 &\leq 4(\mathcal{H}(t)+1+\log 2\mathcal{H}_1) \\ &\leq 4(\mathcal{H}(h,n^2\log d_{\mathbf{x}},n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}}) + m(d_{\mathbf{p},\mathbf{x}}^n + \log(d_{\mathbf{p},\mathbf{x}}^n + m)) + 1 + \log 2\mathcal{H}_1) \\ &= h(nmd_{\mathbf{p},\mathbf{x}}^n)^{O(1)}. \end{aligned}$$

Nous reprenons maintenant les différentes étapes de cette deuxième phase du calcul, en imposant des conditions de non-dégénérescence sur le paramètre $\mathbf{p}_3$ et le nombre premier p_2, ce qui permettra de définir l'entier $\mathcal{H}_2(h,n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}})$ qui contrôle le choix de p_2.

15.4 Fonction d'arrêt

La fonction d'arrêt du processus de remontée est présentée ci-dessous. Elle est semblable à celle de la section 10.4 du chapitre 10, avec le nombre premier p_2 comme paramètre supplémentaire, puisque l'on effectue le test d'arrêt modulo l'idéal de $\mathbb{Z}[p_1,\dots,p_m]$ engendré par $(p_1 - p_{3,1},\dots,p_m - p_{3,m})$ et p_2.

FIG. 15.3 – *Critère d'arrêt et reconstruction*

procedure `Arret`($\mathbf{f}$,$\mathbf{p}_3$,p_2,γ,$\mathcal{R}_i(t)$)

\# **Entree**
\# le système $\mathbf{f}$
\# $\mathbf{p}_3 = (p_{3,1},\dots,p_{3,m})$: valeur des parametres
\# p_2 : nombre premier
\# γ : changement de variables
\# $\mathcal{R}_i(t)$: résolution à coefficients dans $\mathbb{Z}/p_1^{2^i}\mathbb{Z}[[p'_1,\dots,p'_m]]$
\# **Sortie**
\# b,$\mathcal{R}'_i(t)$: b booléen ; si b est vrai, $\mathcal{R}'_i(t)$ est la reconstruction de $\mathcal{R}$ dans $\mathbb{Q}(p_1,\dots,p_m)$
\# et $\mathcal{R}'_i(t)$ annule $\mathbf{f}$ en $\mathbf{p}$ modulo p

b,$\mathcal{R} \leftarrow$ `ReconstructionRationnelle`($\mathcal{R}_i(t)$,γ);
b,$\mathcal{R}'_i(t) \leftarrow$ `ReconstructionNombresRationnels`($\mathcal{R}$);
if not b **then return false** ; **fi** ;
$\mathcal{J} \leftarrow (p) + (p_1 - p_{2,1},\dots,p_m - p_{2,m}) \subset \mathbb{Z}[p_1,\dots,p_m]$;
$b \leftarrow$ `Temoin`($\mathbf{f}$,$\mathcal{J}$,$\mathcal{R}'_i(t)$);
if b **then return true**,$\mathcal{R}'_i(t)$; **else return false** ; **fi** ;
end;

Proposition 52 (Complexité de la fonction d'arrêt) *À la ième étape de remontée des entiers, la complexité de la fonction d'arrêt est*

$$O_{log}\big(\mathcal{M}(\mathcal{D}_{\mathbf{x}})(nm\mathcal{M}_s(4\mathcal{D}(t),m)\mathcal{M}_{\mathbb{N}}(2^i \log p_1) + (n\mathcal{T}(\mathcal{D}(t),m) + (n^{\Omega} + nL))\mathcal{M}_{\mathbb{N}}(\log p_2))\big)$$

opérations binaires.

Démonstration. Les sous-sections suivantes donnent la complexité de chacune des étapes de la fonction d'arrêt. La borne ci-dessus est un majorant de la somme de ces complexités. □

15.4.1 Reconstruction des fractions rationnelles modulo $p_1^{2^i}$

À l'étape i, on applique l'algorithme de reconstruction rationnelle à des séries formelles appartenant à $\mathbb{Z}/p_1^{2^i}\mathbb{Z}[[p'_1,\dots,p'_m]]$. L'anneau de base $\mathbb{Z}/p_1^{2^i}\mathbb{Z}$ n'est pas intègre, mais cela ne va pas poser problème. Sous les hypothèses d'un choix chanceux du changement de variables γ et du premier

p_1, la reconstruction des fractions modulo p_1 est la trace du calcul sur $\mathbb{Q}$, pour le changement de variables γ. Il en est donc de même pour la reconstruction modulo les puissances de p_1.

Proposition 53 *À la ième étape de remontée des entiers, la complexité de la reconstruction des fractions rationnelles modulo $p_1^{2^i}$ est*

$$O_{log}\big(nm\mathcal{D}_{\mathbf{x}}\mathcal{M}(4\mathcal{D}(t))\mathcal{M}_{\mathbb{N}}(2^i \log p_1)\big)$$

opérations binaires.

Démonstration. On reprend l'estimation de complexité de la section 9.1 du chapitre 9, en travaillant avec des nombres à $2^i \log p_1$ chiffres. □

15.4.2 Reconstruction des nombres rationnels

La procédure `ReconstructionNombresRationnels` tente de reconstruire tous les coefficients des fractions rationnelles qui apparaissent comme coefficients dans la résolution $\mathcal{R}_i'(t)$ produite ci-dessus.

Proposition 54 *À la ième étape de remontée des entiers, la complexité de la reconstruction des nombres rationnels est*

$$O_{log}\big(n\mathcal{D}_{\mathbf{x}}\mathcal{T}(\mathcal{D}(t),m)\mathcal{M}_{\mathbb{N}}(2^i \log p_1)\big)$$

opérations binaires.

Démonstration. À la $i$ème étape, on traite des nombres inférieurs à $2^i \log p_1$ chiffres, la reconstruction d'un tel rationnel se fait en $O(\mathcal{M}_{\mathbb{N}}(2^i \log p_1)(i + \log\log p_1))$ opérations binaires. Négligeant les facteurs logarithmiques, on obtient le résultat demandé en prenant en compte tous les coefficients. □

15.4.3 Critère d'arrêt

Si toutes les reconstructions sont possibles, la résolution candidat est alors évaluée sur une valeur $\mathbf{p}_3 = (p_{3,1}, \ldots, p_{3,m})$ de Ω^m, puis vérifiée modulo le nombre premier p_2, par appel à la fonction `Temoin`, pour l'idéal de $\mathbb{Z}[p_1, \ldots, p_m]$ engendré par p_2 et $(p_1 - p_{3,1}, \ldots, p_m - p_{3,m})$. Si le test répond “vrai”, on décrète que la résolution est la bonne et on arrête les calculs.

Proposition 55 *Supposons que $\mathbf{p}_3$ satisfait toutes les hypothèses de généricité. Soit p_2 un nombre premier qui ne divise pas l'entier $e_{Red}(\mathbf{p}_3,t)$. À la ième étape, le coût de la fonction* `Temoin` *est*

$$O_{log}\big(\mathcal{M}(\mathcal{D}_{\mathbf{x}})(n\mathcal{T}(\mathcal{D}(t),m)\mathcal{M}_{\mathbb{N}}(\max\{2^i \log p_1, \log p_2\}) + (n^{\Omega} + nL)\mathcal{M}_{\mathbb{N}}(\log p_2))\big)$$

opérations binaires.

Démonstration. On applique l'estimation de complexité pour `Temoin` donnée dans la section 9.2 du chapitre 9. La réduction modulo p_2 et la spécialisation en $\mathbf{p}_3$ d'un coefficient de $\mathcal{R}_i'(t)$ se font en

$$O_{\log}\big(\mathcal{T}(\mathcal{D}(t),m)\mathcal{M}_{\mathbb{N}}(\max\{2^i \log p_1, \log p_2\})\big)$$

opérations binaires. Ensuite, le coût d'une opération dans le quotient est $\mathcal{M}_{\mathbb{N}}(\log p_2))$ opérations binaires, ce qui donne le coût des deux dernières étapes. □

Probabilité d'erreur. Le risque d'erreur est une fois de plus de s'arrêter trop tôt. Supposons que lors d'une étape intermédiaire de remontée, la reconstruction des nombres rationnels d'une résolution $\mathcal{R}_i(t)$ est possible. Soit $\mathcal{R}'_i(t)$ la résolution reconstruite, à coefficients dans $\mathbb{Q}(p_1,\ldots,p_m)$, et supposons que $\mathcal{R}'_i(t)$ n'est pas égale à la résolution $\mathcal{R}(t)$. Le critère d'arrêt échoue si le couple $(\mathbf{p}_3,p_2)$ est *mauvais témoin* pour $(\mathcal{R}'_i(t),\mathcal{R}(t))$. La proposition suivante quantifie ces témoins, en reprenant les notations de la section 14.2 du chapitre 14. Malgré l'aspect peu engageant de ces formules, le lecteur courageux pourra noter un certain parallèle avec les résultats de la section 10.4 du chapitre 10.

Proposition 56 *Les points $\mathbf{p}_3$ qui satisfont toutes les hypothèses de généricité, admettent t comme élément séparant et sont mauvais témoins pour les reconstructions des étapes intermédiaires sont inclus dans une hypersurface de $\mathbb{C}^m$ de degré au plus*

$$2K\mathcal{D}(t).$$

Pout tout point $\mathbf{p}_3$ bon témoin à coordonnées entières, il existe un entier $e_{Témoin}(\mathbf{p}_3,t)$ non nul tel que si p_2 ne divise pas $e_{Red}(\mathbf{p}_3,t)e_{Témoin}(\mathbf{p}_3,t)$, le couple $(\mathbf{p}_3,p_2)$ est bon témoin pour toutes ces étapes. Si les coordonnées de $\mathbf{p}_3$ sont inférieures à M, $e_{Témoin}(\mathbf{p},t)$ peut être pris de hauteur inférieure à

$$K\big(2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h+2\log(\ell M)+n^2\log d_{\mathbf{x}}+6(n+2m+5)\log(2n+m+2)\big)+1\big)+2^{K+1}(d_{\mathbf{p},\mathbf{x}}^n+1)^m\log p_1.$$

Démonstration. À l'étape i, si la reconstruction des nombres rationnels est possible, l'algorithme échoue si $\mathbf{p}_3$ est mauvais témoin pour $(\mathcal{R}'_i(t),\mathcal{R}(t))$. La quantification des points $\mathbf{p}_3$ mauvais témoins est faite dans la proposition 10.4 du chapitre 10 : ils sont inclus dans une hypersurface de degré au plus $2\mathcal{D}(t)$. Prenant toutes les étapes en compte, on arrive à la borne en degré annoncée.

Si $\mathbf{p}_3$ est bon témoin, l'algorithme échoue si p_2 divise le produit des entiers $e_{\text{Témoin}}(\mathbf{p}_3,t,\mathcal{R}'_i(t))$ associés aux $\mathcal{R}'_i(t)$ pour lesquelles la reconstruction est possible — ces entiers sont définis dans la section 14.2 du chapitre 14. Les coefficients de $\mathcal{R}'_i(t)$ sont des fractions rationnelles de degré au plus $\mathcal{D}(t)$, dont les coefficients sont des rationnels de hauteur au plus $2^i\log p_1$. On peut donc les écrire comme des quotients de polynômes de $\mathbb{Z}[p_1,\ldots,p_m]$ dont les coefficients ont une hauteur au plus $\mathcal{T}(d_{\mathbf{p},\mathbf{x}}^n,m)2^i\log p_1$, où $\mathcal{T}(d_{\mathbf{p},\mathbf{x}}^n,m)$ désigne le nombre de monômes de degré au plus $d_{\mathbf{p},\mathbf{x}}^n$ en m variables. On peut majorer cette hauteur par $2^i(d_{\mathbf{p},\mathbf{x}}^n+1)^m\log p_1$.

Rappelons que l'on s'impose de travailler avec un élément séparant de hauteur au plus $n^2\log d_{\mathbf{x}}$. D'après la majoration donnée dans la section 14.2 du chapitre 14, la hauteur de $e_{\text{Témoin}}(\mathbf{p}_3,t,\mathcal{R}'_i(t))$ est inférieure à

$$2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h+2\log(\ell M)+n^2\log d_{\mathbf{x}}+6(n+2m+5)\log(2n+m+2)\big)+1+2^i(d_{\mathbf{p},\mathbf{x}}^n+1)^m\log p_1$$

La somme sur toutes les étapes donne le résultat annoncé. □

Notations Nous noterons $\mathcal{H}_{\text{Témoin}}(h,n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}},M)$ la borne de hauteur introduite ci-dessus. En vertu des majorations données sur K et $2^{K+1}\log p_1$, cette borne vérifie l'inégalité :

$$\mathcal{H}_{\text{Témoin}} \quad = \quad h\log h(d_{\mathbf{x}}^n+1)^m(nmd_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^n)^{O(1)}.$$

Nous renonçons à en donner une écriture simplifiée.

15.5 Complexité et probabilité d'erreur

Résumons les conditions assurant que l'algorithme de remontée des entiers renvoie bien une résolution des zéros simples génériques sur $\mathbb{Q}$. Il s'agit juste de reprendre les conditions qui ont été signalées dans les sections précédentes.

Lemme 15 *Les conditions suivantes assurent le succès de la phase de remontée des entiers :*
- $\mathbf{p}_3$ *satisfait toutes les hypothèses de généricité et admet* t *comme élément séparant ;*
- $\mathbf{p}_3$ *est un bon témoin pour toutes les reconstructions intermédiaires ;*
- p_2 *ne divise pas* $e_{Red}(\mathbf{p}_3,t)e_{Témoin}(\mathbf{p}_3,t)$.

Notations Nous utilisons toujours le paramètre ℓ introduit dans la section 15.2. Soit M' le maximum de $d^n_{\mathbf{p},\mathbf{x}}(2d^n_{\mathbf{x}} + nd_{\mathbf{p},\mathbf{x}})$ et $2Kd^n_{\mathbf{p},\mathbf{x}}$, nous choisirons $\mathbf{p}_3$ dans le pavé défini sur $\Omega' = \{1,\ldots,\ell M'\}$. Soient ensuite e_{Red} et $e_{\mathrm{Témoin}}$ les maxima respectifs des entiers $e_{\mathrm{Red}}(\mathbf{p}_3,t)$ et $e_{\mathrm{Témoin}}(\mathbf{p}_3,t)$, pour $\mathbf{p}_3$ décrivant Ω'^m et t élément séparant de la fibre générique à coefficients entiers bornés par $n^2 \log d_{\mathbf{x}}$.

Définition 16 *Nous noterons* $\mathcal{H}_2(h,n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}})$ *le majorant suivant de* $\ell \log e_{Red}e_{Témoin}$.

$$\begin{aligned}\mathcal{H}_2(h,n,m,d_{\mathbf{p},\mathbf{x}},d_{\mathbf{x}}) &= \ell(\mathcal{H}_{Red} + \mathcal{H}_{Témoin})(h,n^2 \log d_{\mathbf{x}},n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}},\ell M') \\ &\leq h(d^n_{\mathbf{x}} + 1)^m (\log hnmd^n_{\mathbf{x}} d^n_{\mathbf{p},\mathbf{x}})^{O(1)}.\end{aligned}$$

Notons $\mathcal{P}_{\mathcal{H}_2,2\mathcal{H}_2}$ l'ensemble des nombres premiers compris entre $\mathcal{H}_2$ et $2\mathcal{H}_2$. Nous tirons les témoins $\mathbf{p}_3$ et p_2 uniformément dans l'espace de configurations

$$\mathcal{E}' = \Omega'^m \times \mathcal{P}_{\mathcal{H}_2,2\mathcal{H}_2}.$$

Proposition 57 *Le point* $(\mathbf{p}_3,p_2)$ *assure le succès de la phase de remontée des entiers avec une probabilité au moins*

$$\frac{4}{\ell}.$$

Démonstration. Étudions les effets de chacune des conditions énoncées dans le lemme ci-dessus. Dans les deux premiers points, nous utilisons le lemme de Zippel-Schwartz géométrique du chapitre 3. Dans le dernier point, c'est sa version arithmétique qui est utilisée.

- La première condition supprime au plus $d^n_{\mathbf{p},\mathbf{x}}(2d^n_{\mathbf{x}} + nd_{\mathbf{p},\mathbf{x}})|\Omega|^{m-1}$ valeurs de $\mathbf{p}_3$, soit

  $$M'|\Omega'|^{m-1}|\mathcal{P}_{\mathcal{H}_2,2\mathcal{H}_2}|$$

  couples.
- La deuxième condition supprime au plus $2Kd^n_{\mathbf{p},\mathbf{x}}|\Omega'|^{m-1}$ valeurs de $\mathbf{p}_3$, soit au plus

  $$M'|\Omega'|^{m-1}|\mathcal{P}_{\mathcal{H}_2,2\mathcal{H}_2}|$$

  couples.
- La dernière condition discrimine pour chaque point $\mathbf{p}_3$ les diviseurs de $e_{\mathrm{Red}}(\mathbf{p}_3,t)e_{\mathrm{Témoin}}(\mathbf{p}_3,t)$. On discrimine donc au plus

  $$\frac{|\Omega'|^m \log e_{\mathrm{Red}} e_{\mathrm{Témoin}}}{\log \mathcal{H}_2}$$

  couples.

Au total, on a donc éliminé au plus

$$2M'|\Omega'|^{m-1}|\mathcal{P}_{\mathcal{H}_2,2\mathcal{H}_2}| + \frac{|\Omega'|^m \log e_{\text{Red}} e_{\text{Témoin}}}{\log \mathcal{H}_2}$$

couples, ce qui représente une proportion inférieure à

$$\frac{4}{\ell},$$

d'après la définition de $\mathcal{H}_2$. □

Proposition 58 *Choisissons une fois pour toutes $\ell = 20$. Alors la probabilité d'erreur du processus de remontée des séries modulo p_1 suivi de la remontée des entiers est inférieure à $1/2$.*

Démonstration. Les probabilité d'erreur des deux phases du calculs ont été bornées respectivement par $6/\ell$ et $4/\ell$. Le choix $\ell = 20$ permet d'assurer une probabilité totale inférieure à $1/2$.

Théorème 18 *En cas de succès, la complexité totale du calcul de la résolution générique est*

$$O_{log}\big(\mathcal{M}(\mathcal{D}_{\mathbf{x}})(n^{\Omega} + nL + nm)\mathcal{M}_s(4\mathcal{D}(t),m)\mathcal{M}_{\mathbb{N}}(\mathcal{H}(t) + \log h + mn \log d_{\mathbf{p},\mathbf{x}})\big)$$

opérations binaires.

Démonstration. Le coût de la remontée des entiers est dominé par le coût de la dernière étape. Un premier pas consiste à rassembler le coût de toutes les étapes dans l'estimation

$$O_{\log}\big(\mathcal{M}(\mathcal{D}_{\mathbf{x}})((n^{\Omega}+nL+nm)\mathcal{M}_s(4\mathcal{D}(t),m)\mathcal{M}_{\mathbb{N}}(2^{K+1}\log p_1)+(n\mathcal{T}(\mathcal{D}(t),m)+(n^{\Omega}+nL))\mathcal{M}_{\mathbb{N}}(\log p_2))\big)$$

opérations binaires. Ce coût domine également celui de la remontée des séries modulo p_1. Ensuite, les estimations sur $2^{K+1}\log p_1$, puis $\mathcal{H}_1$ et $\mathcal{H}_2$ permettent de conclure qu'à des facteurs logarithmiques près, $\log p_1$ et $\log p_2$ contribuent à hauteur de $O(\log h + mn \log d_{\mathbf{p},\mathbf{x}})$. Quelques simplifications mènent au résultat demandé. □

15.6 Vérification probabiliste d'une solution

Les bornes sur les nombres premiers données dans les sections ci-dessus assurent que la complexité du calcul reste polynomiale et que la probabilite d'erreur est inférieure à $1/2$. Celles-ci peuvent malgré tout être trop importantes pour des applications pratiques, de sorte que nous pourrons être amenés à passer outre, et de choisir des nombres premiers de taille plus faible.
En conséquence, cette dernière section présente une procédure de vérification probabiliste d'une résolution $\mathcal{R}'(t) = (t,q,\mathbf{v})$ à coefficients dans $\mathbb{Q}(p_1,\ldots,p_m)$, indépendamment de l'algorithme qui a pu la calculer. Nous conservons le principe de la section 10.6 du chapitre 10 : le test consiste simplement à évaluer le système sur la résolution candidate spécialisée en $(\mathbf{p},p)$, pour une valeur des paramètres $\mathbf{p}$ à coordonnées entières et un nombre premier p.

Pour quantifier la probabilité d'erreur de ce test, nous reprenons l'argumentation de la dernière section du chapitre 10, en rajoutant dans la discussion des considérations de calcul modulaire, concernant la taille du nombre premier à choisir. À nouveau, nous distinguerons deux cas, selon les hypothèses que l'on peut faire sur la résolution candidat $\mathcal{R}'(t)$.

Dans tous les cas, nous supposons que les coefficients de la résolution $\mathcal{R}'(t)$ sont tous mis au même dénominateur, et sont écrits comme des quotients de polynômes de $\mathbb{Z}[p_1,\ldots,p_m]$. Nous notons d le degré de $\mathcal{R}'(t)$ en t, $\mathcal{D}$ une borne sur le degré en $(p_1,\ldots,p_m)$ des numérateurs et des dénominateurs des coefficients, et H une borne sur la hauteur des numérateurs de dénominateurs des coefficients.
Nous supposons enfin que $\mathcal{R}'(t)$ n'est pas la description d'*une* composante des zéros simples génériques de $(f_1,\ldots,f_n)$: sans hypothèse supplémentaire, notre test peut nier cette affirmation, mais pas détecter s'il manque des composantes.

15.6.1 En degré maximal

Dans ce premier paragraphe, nous supposons assuré que la résolution $\mathcal{R}'(t)$ a degré maximal en t. Dans la dernière section du chapitre 10, nous discutons de situations dans lesquelles cette information peut être acquise.
Nous pouvons alors appliquer les résultats de quantification des points témoins donnés dans le chapitre 14. Nous passons les détails de la preuve : ce sont les analogues exacts de la situation sur un corps k quelconque données dans le chapitre 10, où se rajoute le lemme de Zippel-Schwartz dans sa forme arithmétique.

Proposition 59 *Posons $M = \mathcal{D} + d^n_{\mathbf{p},\mathbf{x}}$. Choisissons alors un entier ℓ et posons $\Omega = \{1,\ldots,\ell M\}$, puis*

$$\mathcal{H} = 2\ell d^n_{\mathbf{p},\mathbf{x}}\big((n+1)h + 6(n+2m+5)\log(2n+m+2) + 2\log \ell M\big) + \ell(H+1).$$

Nous appelons $\mathcal{P}_{\mathcal{H},2\mathcal{H}}$ l'ensemble des nombres premiers compris entre $\mathcal{H}$ et $2\mathcal{H}$ et notons $\mathcal{E}$ l'espace des configurations $\Omega^m \times \mathcal{P}_{\mathcal{H},2\mathcal{H}}$.
Tirons de manière uniforme les points $(\mathbf{p},p)$ dans $\mathcal{E}$. Si la spécialisation de $\mathcal{R}'(t)$ en $(\mathbf{p},p)$ est possible, la probabilité pour que celle-ci annule le système $\mathbf{f}(\mathbf{p},.)$ pris modulo p et décrive des points simples de $(f_1,\ldots,f_n)$ est inférieure à

$$\frac{3}{\ell}.$$

15.6.2 Dans le cas général

Sans faire d'hypothèse sur le degré de $\mathcal{R}'(t)$, nous devons suivre l'argumentation de la section 10.6.2 du chapitre 10, en utilisant la projection sur l'espace des variables $(p_1,\ldots,p_m,u)$, où u est une forme linéaire qui sépare les points de $\mathcal{U}$ des points décrits par $\mathcal{R}'(t)$. D'après la proposition 2 du chapitre 1, nous pouvons choisir u de hauteur au plus $n^2(\log d_{\mathbf{x}} + 1)$.

La suite de calculs suivants permet d'estimer un majorant sur les hauteurs $(\mathcal{H}_{\mathcal{U}},\mathcal{H}_{\mathcal{V}},\mathcal{H}_Q)$ des objets $(P_{\mathcal{U}},P_{\mathcal{V}},Q)$ construits dans la section 10.6.2 du chapitre 10 ; leur preuve ne présente pas de difficulté. Pour que le test soit valable, le nombre premier p ne doit diviser ni l'entier $e_{\mathrm{Red}}(\mathbf{p},t)$, ni l'évaluation du polynôme Q au point $\mathbf{p}$; le nombre $\mathcal{H}$ vient mesurer la hauteur du produit de ces deux entiers.
Une fois de plus, nous introduisons une valeur entière arbitraire notée ℓ, qui permet de demander une probabilité d'erreur arbitrairement faible.

$$
\begin{aligned}
\mathcal{H}_{\mathcal{U}} &:= 2d_{\mathbf{p},\mathbf{x}}^n\big((n+1)h + n^2(\log d_{\mathbf{x}}+1) + 6(n+2m+3)\log(2n+m+2)\big),\\
\mathcal{D}_{\mathcal{V}} &:= (2d-1)\mathcal{D},\\
\mathcal{H}_{\mathcal{V}} &:= (2d-1)\big(H + n^2(\log d_{\mathbf{x}}+1) + \log n + \log(2d-1) + \mathcal{D}\log(m+2)\big),\\
\delta &:= d_{\mathbf{x}}^n - d,\\
\mathcal{D}_Q &:= (\delta+1)\mathcal{D} + d_{\mathbf{p},\mathbf{x}}^n\\
\mathcal{H}_Q &:= (\delta+1)\mathcal{H}_{\mathcal{V}} + \mathcal{H}_{\mathcal{U}} + (\delta\mathcal{D}_{\mathcal{V}} + \min\{(\delta+1)\mathcal{D}_{\mathcal{V}}, d_{\mathbf{p},\mathbf{x}}^n\})\log(m+1) + (\delta+2)\log(\delta+2)\\
M &:= d_{\mathbf{p},\mathbf{x}}^n(2d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}}) + \mathcal{D}_Q,\\
\mathcal{H} &:= 8\ell d_{\mathbf{x}}^{2n}(n+2)^3\big(h + h_t + d_{\mathbf{p},\mathbf{x}}(\log \ell M + \log(m+1)) + 6\log(2n+2)\big) + \ell\mathcal{H}_Q +\\
&\quad\ \ell\mathcal{D}_Q(\log \ell M + \log(m+1)).
\end{aligned}
$$

Proposition 60 *Soit Ω l'ensemble $\{1,\ldots,\ell M\}$, puis $\mathcal{P}_{\mathcal{H},2\mathcal{H}}$ l'ensemble des nombres premiers compris entre $\mathcal{H}$ et $2\mathcal{H}$; cet ensemble comporte au moins $\mathcal{H}/(2\log\mathcal{H})$ éléments. Nous notons enfin $\mathcal{E}$ l'espace des configurations $\Omega^m \times \mathcal{P}_{\mathcal{H},2\mathcal{H}}$.*
Tirons de manière uniforme les points $(\mathbf{p},p)$ dans $\mathcal{E}$. Si la spécialisation de $\mathcal{R}'(t)$ en $(\mathbf{p},p)$ est possible, la probabilité que celle-ci annule le système $\mathbf{f}$ est inférieure à

$$\frac{3}{\ell}.$$

Nous passerons les détails de la preuve. Elle suit la preuve de la proposition de la section 10.6.2 du chapitre 10.

Exemple Posons $(n = 6, m = 2, h = 3, d_{\mathbf{p},\mathbf{x}} = d_{\mathbf{x}} = 7)$. Supposons donnée une résolution candidate dont les coefficients sont des fractions rationnelles, toutes mises au même dénominateur, de degré en $(p_1,\ldots,p_m)$ au plus 200 et de hauteur (en base e) 20000, pour un élément primitif t de hauteur 1, de degré 200 en t. Posons $\ell = 100$, il nous faut alors travailler avec un nombre premier de 17 chiffres décimaux, pour une probabilité de l'ordre de 1/30. Sur un tel exemple, nous pouvons sans difficulté trouver un nombre premier de la taille requise et certifier qu'il est premier.

Sur le même exemple, les bornes $(\mathcal{H}_1,\mathcal{H}_2)$ des sections précédentes demandent de travailler avec des nombres premiers de 28 et 21 chiffres décimaux pour assurer un *calcul* avec la même probabilité d'erreur.

Chapitre 16

Mise en pratique

Résumé

Nous décrivons l'implantation de l'algorithme du chapitre précédent dans le langage Magma ; elle se base sur le prototype `kronecker`. Nous donnons ensuite les performances de l'implantation sur une série d'exemples, pour la plupart tirés des chapitres d'applications ; ces exemples montrent que l'implantation résoud des problèmes où échouent les fonctions proposées par Magma.

Sommaire

16.1 Implantation

Le code n'est pas sensiblement différent de la version sur un corps fini ; quasiment toutes les fonctions nécessaires à l'algorithme de calcul sur les rationnels ont été présentées dans la partie V. En particulier, nous utilisons à nouveau le code de remontée de Newton pour une représentation univariée issu de l'implantation `kronecker`, écrite par G. Lecerf. Seule la reconstruction des nombres rationnels à partir de leur développement p-adique est nouvelle ; cette fonction est présente dans le logiciel Magma.

16.2 Résultat des calculs

16.2.1 Comparaison à d'autres méthodes

Comme nous l'avons fait dans le chapitre 11, nous comparons notre implantation avec les fonctions de de calcul de bases de Gröbner proposées par Magma version 2.6. Deux possibilités sont envisagées : calcul dans l'anneau $\mathbb{Q}(p_1,\dots,p_m)[x_1,\dots,x_n]$, ou bien dans l'anneau $\mathbb{Q}[p_1,\dots,p_m,x_1,\dots,x_n]$. Nous n'avons pas comparé avec d'autres moteurs de calcul de bases de Gröbner, puisque celui de Magma est reconnu comme l'un des meilleurs systèmes disponibles. Notons malgré tout que l'implantation FGb de J.-C. Faugère [1, 70] offre des performances de haut niveau.

16.2.2 Choix des systèmes

Nous proposons une résolution des systèmes suivants, en partie tirés des parties d'applications IX et X.

- **Birkhoff** : ces systèmes sont présentés dans le chapitre 26, et donnés dans l'annexe E.7;
- $\mathbf{j_1j_2j_3}$: l'origine et les applications de ce système sont décrits dans le chapitre 25 ; le système est donné dans l'annexe E.5.
- **Hawes4** : ce système-test est écrit dans l'annexe E.9.

16.2.3 Tableau de temps

Le tableau ci-dessous indique le temps des calculs pour les systèmes mentionnés précédemment.

- La ligne (p_1,p_2) donne le nombre de chiffres décimaux des nombres premiers (p_1,p_2) utilisés dans l'algorithme, en respectant les bornes données dans le chapitre 15 : l'ensemble du processus de calcul a alors une probabilité d'échec inférieure à 1/2.
- La ligne *p Vérification* utilise les conclusions de la section 15.6 du chapitre 15 : nous vérifions la résolution candidate en une valeur aléatoire des paramètres, modulo un nombre premier p. Cette ligne donne le nombre de chiffres décimaux d'un nombre premier pour lequel ce test nous révèlera si la résolution candidate est incorrecte avec une probabilité supérieure à 999/1000.
- La ligne *Temps* donne le temps de calcul de notre implantation.
- La ligne *Gröbner 1* donne le temps le calcul d'une base de Gröbner lexicographique dans $\mathbb{Q}(p_1,\dots,p_m)[x_1,\dots,x_n]$, et la ligne *Gröbner 2* indique le temps de calcul d'une base de Gröbner lexicographique dans l'anneau $\mathbb{Q}[p_1,\dots,p_m,x_1,\dots,x_n]$.
- Le signe ∞ indique l'arrêt des processus quand la consommation de mémoire a étouffé les calculs.

FIG. 16.1 – *Résolution sur les rationnels*

Système	Birkhoff-3	Birkhoff-10	$j_1j_2j_3$	Hawes4
Variables n	3	3	4	6
Paramètres m	1	1	2	2
Degré Max. $d_{\mathbf{p},\mathbf{x}}$	6	6	10	9
Multiplications L	90	80	41	55
Hauteur h	7	7	31	12
Degré générique $\mathcal{D}_{\mathbf{x}}$	34	136	30	120
Degré paramètres $\mathcal{D}(t)$	5	19	9	46
Hauteur de la sortie $\mathcal{H}(t)$	56	123	89	167
(p_1,p_2)	(17,13)	(17,13)	(23,18)	(22,17)
p Vérification	10	12	14	18
Temps	15 s.	20 min.	7 min.	15 h.
Gröbner 1	5 min.	$> 10h$.	∞	∞
Gröbner 2	17 min.	∞	∞	∞

Conclusion La mise en œuvre sur les rationnels a le même bon comportement que l'implantation sur les corps finis; davantage d'expériences montrent que l'on retrouve une dépendance sensiblement linéaire en la complexité d'évaluation. Une fois de plus, n'oublions pas que l'on compare un algorithme probabiliste à un algorithme déterministe, même si les bornes sur les probabilités d'erreurs permettent de rester serein sur l'éventualité d'un résultat erroné.

L'approche par réduction modulaire puis remontée de Newton permettant d'éviter l'explosion des résultats intermédiaires, le facteur limitant devient la taille de la sortie: la résolution de l'exemple Hawes4 occupe plusieurs dizaines de Mo.

Septième partie

Vers une extension aux représentations triangulaires

Chapitre 17

Vue d'ensemble

17.1 Hypothèses et notations

Nous nous plaçons dans cette partie sous les hypothèses 1 du chapitre 7, mais abordons la question d'une représentation des zéros génériques par des ensembles triangulaires, dépendant des paramètres $(p_1, \ldots, p_m)$. Dans le chapitre 8, nous avons décrit les conséquences géométriques des hypothèses du chapitre 7 : la variété des points simples notée $\mathcal{U}$ est de dimension m, l'image de sa projection sur l'espace des paramètres $\mathbb{A}^m(\overline{k})$ est dense, et les variables $\mathbf{x}$ sont algébriques sur les variables $\mathbf{p}$. Nous avons baptisé zéros simples génériques les points décrits par l'idéal $\mathcal{I}(\mathcal{U})$ étendu dans $k(p_1, \ldots, p_m)[x_1, \ldots, x_n]$, pour donner des bornes sur les degrés en $(p_1, \ldots, p_m)$ apparaissant dans une résolution de Kronecker de ces zéros.

Désormais, nous nous intéressons à des *ensembles triangulaires* qui décriront les zéros simples génériques du système $(f_1, \ldots, f_n)$. La description qui suit est conséquence des résultats de la section 1.2, et en particulier la proposition, 4 du chapitre 1, et du résultat de séparabilité obtenu dans le chapitre 8.

Proposition 61 *Il existe une famille finie d'ensembles triangulaires $\mathbf{t}^{(j)} = (t_1^{(j)}, \ldots, t_n^{(j)})$, à coefficients dans $k(p_1, \ldots, p_m)$, qui décrivent les zéros simples génériques de $\mathbf{f}$. Ces ensembles satisfont les propriétés suivantes :*

- *Pour tous j et k, le polynôme $t_k^{(j)}$ est unitaire en x_k, ne dépend que des variables $(x_k, \ldots, x_n)$ et est réduit par rapport aux polynômes $(t_{k+1}^{(j)}, \ldots, t_n^{(j)})$.*
- *Pour tout j, $\mathbf{t}^{(j)}$ engendre un idéal premier de $k(p_1, \ldots, p_m)[x_1, \ldots, x_n]$.*
- *Pour tous j et k, la dérivée du polynôme $t_k^{(j)}$ par rapport à x_k est inversible par rapport aux polynômes $(t_k^{(j)}, \ldots, t_n^{(j)})$.*
- *Pour tout k entre 1 et n, si $j_1 \neq j_2$, alors*
  - *soit $(t_k^{(j_1)}, \ldots, t_n^{(j_1)}) = (t_k^{(j_2)}, \ldots, t_n^{(j_2)})$,*
  - *soit $(t_k^{(j_1)}, \ldots, t_n^{(j_1)}) + (t_k^{(j_2)}, \ldots, t_n^{(j_2)}) = (1)$.*

En particulier, le polynôme minimal de x_n est le produit des $t_n^{(j)}$ distincts.

Les notations de complexités employées dans cette partie sont en partie rappelées dans le chapitre 7. Nous supposerons en outre que l'algorithme de multiplication de polynômes univariés utilisé a une complexité en $O_{\log}(d^\alpha)$, pour des polynômes de degré d. La notation $\mathcal{M}(d_1, \ldots, d_n)$ désigne le coût de la multiplication modulo un ensemble triangulaire de polynômes de degrés $(d_1, \ldots, d_n)$.

Enfin, la constante universelle C désigne le facteur exponentiel lié au coût de la multiplication dans un empilement de quotients univariés. Ces grandeurs sont présentées avec plus de détails dans le chapitre 5.

Nous dirons qu'une valeur des paramètres $\mathbf{p}$ *satisfait toutes les hypothèses de généricité* si les solutions du système $\mathbf{f}(\mathbf{p},.)$ sont en nombre fini, maximal et sont des points simples de $\mathcal{V}(f_1,\ldots,f_n)$, si tous les ensembles $\mathbf{t}^{(j)}$ se spécialisent en $\mathbf{p}$ et si les polynômes minimaux de chaque variable se spécialisent en $\mathbf{p}$.

Les références sur d'autres travaux dans ce domaine sont données dans le chapitre 7.

17.2 Résultats principaux et plan des chapitres suivants

Cette partie s'organise comme les deux précédentes. Dans un premier temps, nous étudions les conséquences de nos hypothèses d'un point de vue géométrique ; nous en déduisons ensuite un algorithme de calcul, ici du *polynôme minimal* d'une variable. Nous ne distinguerons pas explicitement le cas des rationnels : la démarche serait la même que dans la partie précédente, et semée de détails techniques. Nous nous conterons d'énoncer un résultat à ce sujet dans le chapitre 19, et de décrire le comportement de notre implantation dans ce cas.

Le chapitre 18 pose la question d'une *majoration des degrés* en les paramètres intervenant dans un ensemble triangulaire codant une composante des zéros simples. Nous montrons tout d'abord un résultat de spécialisation : à nouveau, nous introduisons un lieu discriminant, qu'il suffit d'éviter pour pouvoir spécialiser la résolution générique.

Théorème 19 *Il existe un polynôme $\mathcal{D}isc$ non nul de $k[p_1,\ldots,p_m]$ tel que toute valeur $\mathbf{p}$ n'annulant pas $\mathcal{D}isc$ satisfait toutes les hypothèses de généricité. Le polynôme $\mathcal{D}isc$ peut être pris de degré inférieur à*

$$d_{\mathbf{p},\mathbf{x}}^n(2(n+1)d_{\mathbf{p},\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}}).$$

Nous obtenons ensuite une borne pour les degrés en $(p_1,\ldots,p_m)$ des coefficients des $\mathbf{t}^{(j)}$, analogue à celle donnée par Gallo et Mishra dans [72] pour un ensemble caractéristique, mais ici pour *chacune* des composantes des zéros simples génériques.

Théorème 20 *Pour tout j, les coefficients des polynômes de $\mathbf{t}^{(j)}$ ont des numérateurs et dénominateurs de degrés en $(p_1,\ldots,p_m)$ inférieurs à*

$$d_{\mathbf{p},\mathbf{x}}^n(2((n+1)^2+1)d_{\mathbf{x}}^{2n} + n + 1)^{n+1}.$$

Cette borne est d'ordre $d_{\mathbf{p},\mathbf{x}}^{O(n^2)}$, et donc de peu d'utilité pratique. Dans la fin du chapitre, nous posons donc la question de l'introduction d'initiaux bien choisis qui pourraient faire tomber le degré en $(p_1,\ldots,p_m)$. Dans un cas simplifié à deux variables, nous comparons une proposition théorique à des considérations expérimentales.

L'absence d'une bonne borne sur les degrés apparaissant dans les polynômes des $\mathbf{t}^{(j)}$ limite nos ambitions. Dans le chapitre 19, nous ne nous intéressons qu'au calcul du *polynôme minimal générique* d'une variable, par des techniques de remontée de Newton analogues à celles présentées dans les parties précédentes. Notons cette fois $\mathcal{C}_{\text{Zéro}}(\mathbf{f},\mathbf{p},k)$ le nombre d'opérations de k nécessaires pour obtenir une description des zéros simples du système spécialisé $\mathbf{f}(\mathbf{p},.)$ en une famille d'ensembles triangulaires irréductibles. Nous obtenons alors le résultat suivant :

Théorème 21 (Complexité du calcul du polynôme minimal) *Soit $m_{x_n} \in k(p_1,\dots,p_m)[t]$ le polynôme minimal générique de x_n. Notons d_n son degré en t et $\mathcal{D}(x_n)$ le maximum des degrés en $(p_1,\dots,p_m)$ des numérateurs et dénominateurs de ses coefficients.*
On peut calculer le polynôme m_{x_n} par un algorithme probabiliste en

$$\mathcal{C}_{Zéro}(\mathbf{f},\mathbf{p}_1,k) + \mathcal{C}_{Zéro}(\mathbf{f},\mathbf{p}_2,k) + O_{log}\big((n^\Omega + nL)\mathcal{M}_s(4\mathcal{D}(x_n),m)(C^n\mathcal{D}_{\mathbf{x}}^\alpha + md_n)\big)$$

opérations de k, où $\mathbf{p}_1$ et $\mathbf{p}_2$ sont deux valeurs des paramètres. Les choix des valeurs aléatoires intervenant dans l'algorithme qui mènent à l'échec sont enfermés dans une hypersurface algébrique de $\mathbb{A}^{3m-1}(\overline{k})$ de degré au plus $80nd_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^{3n}$.
En clair, le coût du calcul est polynomial en la taille du résultat, le degré des expressions intermédiaires étant bien contrôlé ; notons également la précesence d'un facteur exponential en n lié au calcul modulo un ensemble triangulaire.

Le dernier chapitre de cette section décrit l'implantation en Magma de l'algorithme mentionné ci-dessus, et montre qu'elle permet de faire aboutir des calculs où échouent les fonctions de calculs de bases de Gröbner proposées par Magma. Une comparaison poussée avec les implantations d'autres algorithmes reste à faire (implantations de P. Aubry, A. Dellière et M. Moreno-Maza en Axiom,...)

Pourquoi uniquement le polynôme minimal? Ce choix est dicté par les bornes dont nous disposons sur les degrés des coefficients des éléments des ensembles triangulaires $\mathbf{t}^{(j)}$. Les coefficients du polynôme minimal m_{x_n} ont des degrés bornés par des grandeurs d'origine géométrique, d'après le chapitre 8. Par contre, nous disposons uniquement de la borne en $d_{\mathbf{p},\mathbf{x}}^{O(n^2)}$ pour majorer les degrés des coefficients des polynômes $(t_1^{(j)},\dots,t_{n-1}^{(j)})$.

Notre algorithme de remontée calcule bien les développements de tous les éléments des ensembles triangulaires étudiés, mais en l'absence d'une meilleure borne sur leurs degrés, nous ne chercherons pas à reconstruire les polynômes autres que le dernier — nous ferons une exception à cette règle lors d'une application, dans le chapitre 25.

Il ne s'agit pas d'un problème uniquement théorique : dans bien des cas, les calculs mettant en jeu des ensembles d'une taille conséquente font ressortir que les polynômes intermédiaires sont d'une taille bien supérieure à celle du dernier d'entre eux. Pour résoudre ce problème, la réponse à la question d'*initiaux bien choisis* est cruciale.

Motivations Cette partie se présente comme une extension de la partie V au cas d'une représentation par des ensembles triangulaires. Notre première motivation est la même : nous voulons calculer des résolutions paramétrées. Comme le souligne le dernier point ci-dessus, nous nous limitons pour l'instant au calcul de polynômes minimaux, ce qui fournit matière à des applications décrites dans les chapitres 24, 25 et 27.

La donnée d'une résolution de Kronecker générique permet d'obtenir ce même résultat, au prix d'une élimination supplémentaire. On veut que le coût du calcul prenne en compte la taille de la sortie, c'est-à-dire du degré du polynôme minimal de x_n en $(p_1,\dots,p_m)$. La résolution de Kronecker calcule le polynôme minimal d'un élément séparant combinaison linéaire des variables, dont le degré en $(p_1,\dots,p_m)$ risque d'être supérieur.

Une dernière motivation à l'utilisation des méthodes de remontée dans une représentation triangulaire sera l'extension de l'algorithme de résolution géométrique complet à ce nouveau cadre. L'algorithme commence par effectuer un changement linéaire de variables générique, ce qui a pour effet de mettre toutes les variétés considérées au cours du calcul en position de Noether projective, et rend toute variable séparante. Si nous voulons éviter le changement de variables, il faut accepter que les variables de l'espace ambiant ne soient pas séparantes, et que des projections dominantes ne soient pas finies. Comme premier pas, cette partie présente la généralisation de l'étape de remontée de Newton à ce cadre plus large.

Chapitre 18

Estimations de degré

Résumé

Dans ce chapitre, nous considérons une représentation de l'ensemble des zéros simples du système $(f_1,\ldots,f_n)$ par des ensembles triangulaires $\mathbf{t}^{(j)}$ à coefficients dans $k(p_1,\ldots,p_m)$, pour aborder les questions ci-dessous.

- Dans la première section, nous donnons une borne de nature géométrique sur le degré d'un *lieu discriminant* contenant l'ensemble des valeurs des paramètres en lesquels les coefficients des $\mathbf{t}^{(j)}$ ne se spécialisent pas. Ce résultat sera utilisé dans le chapitre suivant, pour borner des probabilités d'erreur.
- Dans la deuxième section, nous donnons une majoration sur les degrés en $(p_1,\ldots,p_m)$ des coefficients de chaque ensemble $\mathbf{t}^{(j)}$, obtenue par des considérations d'algèbre linéaire et inspirée par celle de Gallo et Mishra [72].
- Dans la troisième section, nous cherchons des bornes intrinsèques sur ces degrés, en introduisant des initiaux bien choisis, inspirés par le cas d'une représentation univariée. Les pistes théoriques sont confrontées à des exemples pratiques.

Sommaire

Introduction

Comme nous l'avons fait pour une résolution de Kronecker, nous cherchons à dominer les degrés en les paramètres apparaissant dans les coefficients des ensembles $\mathbf{t}^{(j)}$. Nous n'avons pas apporté de réponse aussi nette que dans le cas d'une résolution univariée.

Dans la première section, nous montrons que le lieu des points $\mathbf{p}$ pour lesquels la spécialisation de l'un des ensembles $\mathbf{t}^{(j)}$ n'est pas possible est contenu dans la réunion de "lieux discriminants" associés à chaque variable. La preuve exhibe un algorithme de calcul d'un ensemble $\mathbf{t}^{(j)}$, dont on montre qu'il se spécialise en toute valeur des paramètres en dehors des lieux discriminants mentionnés ci-dessus. L'idée est simple : si la degré d'aucune projection ne chute, alors les calculs se spécialisent ; la mise en place demande un travail plus technique.

Dans la deuxième section, nous donnons une borne explicite sur les degrés des fractions rationnelles en $(p_1, \ldots, p_m)$ apparaissant dans *chacun* des ensembles $\mathbf{t}^{(j)}$, sous les conditions de normalisation données dans le chapitre 17. Cette borne est d'ordre de grandeur $d_{\mathbf{p},\mathbf{x}}^{O(n^2)}$, car elle repose sur des outils d'algèbre linéaire, inspirés du résultat de Gallo et Mishra [72]. La preuve utilise le résultat division qui a été rappelé dans le chapitre 3 de la partie 4 : la nature "intrinsèque" de cette proposition permet d'obtenir des coefficients de Bézout dont le degré est borné par un degré géométrique, et non pas par le degré de générateurs.

Finalement, dans la dernière section, nous proposons une piste, dans un cas simplifié à deux variables (x_1, x_2), pour l'introduction d'*initiaux* qui permettraient de faire chuter le degré des coefficients en $(p_1, \ldots, p_m)$, et de pouvoir le contrôler par des bornes d'origine géométrique.
Nous proposons pour cela de multiplier les coefficients de ce polynôme par des puissances bien choisies de la dérivée du polynôme minimal de x_2. Des exemples concrets montrent l'effet de cette manipulation, et indiquent que notre proposition peut certainement être améliorée : dans tous ces exemples, il est préferable de multiplier chaque coefficient par la dérivée du polynôme minimal de x_2, sans l'élever à quelque puissance que ce soit. Je n'ai pas d'explication à ce phénomène.
La question d'une majoration des degrés des ensembles triangulaires $\mathbf{t}^{(j)}$ par l'introduction d'initiaux n'est pas anecdotique : dans le cas simplifié de variables satisfaisant au "Shape Lemma", ces idées ont permis d'obtenir une borne fine, proche de la géométrie et directement applicable.

18.1 Spécialisation de la résolution générique

Dans cette première section, nous décrivons un lieu contenant l'ensemble des valeurs des paramètres pour lesquelles la spécialisation des polynômes formant les ensembles triangulaires $\mathbf{t}^{(j)}$ n'est pas possible.

18.1.1 Description du lieu discriminant

Notons $\mathcal{R}(t) = (t, q(t), \mathbf{w}(t))$ une représentation sous forme "Shape Lemma" des zéros simples de $(f_1, \ldots, f_n)$. D'après les résultats du chapitre 8, on sait enfermer les points où $\mathcal{R}(t)$ ne se spécialise pas à partir du lieu discriminant du polynôme minimal q.

Dans le cas d'une représentation par ensembles triangulaires, il est moins immédiat de donner des conditions assurant la spécialisation. La proposition suivante propose de telles conditions ; elles reviennent à demander que le nombre de valeurs distinctes que prend *chaque variable* ne chute pas à la spécialisation. Au cours de la preuve, nous aurons besoin d'un lemme de spécialisation du calcul de pgcd dans un empilement de quotients univariés, qui est donné dans la sous-section suivante.

Proposition 62 *Soit* $\mathbf{p}$ *une valeur des paramètres. Nous notons* $\mathcal{U}_{\mathbf{p}}$ *l'ensemble des zéros simples du système* $\mathbf{f}(\mathbf{p},.)$*, et nous supposons que :*

- *la fibre* $\mathcal{U}_{\mathbf{p}}$ *est finie, a degré générique et ses points sont des zéros simples du système* $(f_1,\ldots,f_n)$.
- *la résolution* $\mathcal{R}(t)$ *se spécialise en* $\mathbf{p}$ *en une description de* $\mathcal{U}_{\mathbf{p}}$ *;*
- *pour tout* k*, le polynôme minimal de* x_k *dans* $\mathcal{U}_{\mathbf{p}}$ *a degré maximal.*

Alors les zéros simples de $\mathbf{f}(\mathbf{p},.)$ *sont décrits par la spécialisation des* $\mathbf{t}^{(j)}$ *en* $\mathbf{p}$.

Nous proposons tout d'abord un calcul des polynômes de l'ensemble $\mathbf{t}^{(j)}$, inspiré de l'algorithme `LexTriangular` de Lazard [138], où les calculs sont effectués avec des coefficients dans des extensions finies du corps $k(p_1,\ldots,p_m)$. Nous montrerons ensuite que sous les hypothèses de la proposition, ces calculs se spécialisent en $\mathbf{p}$.

Pour chaque ensemble triangulaire $\mathbf{t}^{(j)} = (T_1^{(j)},\ldots,T_n^{(j)})$, nous notons $\mathcal{U}_{\mathbf{p}}^{(j)}$ la composante de $\mathcal{U}_{\mathbf{p}}$ décrite par $\mathbf{t}^{(j)}$, puis $\mathcal{R}(t)^{(j)} = (t,q^{(j)},\mathbf{w}^{(j)})$ le facteur de la résolution $\mathcal{R}(t)$ correspondant à cette composante : ces points sont donc décrits par $q^{(j)}(t) = 0$ et $x_i = w_i^{(j)}(t))$, pour $i = 1,\ldots,n$. Enfin, $\mathrm{m}_{x_k}^{(j)}$ désigne le polynôme minimal de x_k dans $\mathbf{t}^{(j)}$. Tous ces polynômes ont des coefficients dans $k(p_1,\ldots,p_m)$.

Pour simplifier les notations, nous omettons dans la suite l'indice j attaché à chacun de ces objets, étant entendu que nous travaillons toujours dans la même composante.

Lemme 16 *L'algorithme suivant calcule les polynômes* T_k.

- *Si* $k = n$*, posons*
$$T_n(x_n) = m_{x_n},$$
- *sinon, nous effectuons les calculs suivants dans* $Q_{k+1}[x_k,t]$*, avec*
$$Q_{k+1} = k(p_1,\ldots,p_m)[x_{k+1},\ldots,x_n]/(T_{k+1},\ldots,T_n) :$$

$$\left\{\begin{array}{rcl} p_{n,k}(t) & = & \mathrm{pgcd}(x_n - w_n(t),q(t)), \\ p_{n-1,k}(t) & = & \mathrm{pgcd}(x_{n-1} - w_{n-1}(t),p_{n-1,k}(t)), \\ & \vdots & \\ p_{k+1,k}(t) & = & \mathrm{pgcd}(x_{k+1} - w_{k+1}(t),p_{k+2,k}(t)), \\ p_k(x_k) & = & \mathrm{résultant}_t(x_k - w_k(t),p_{k+1,k}(t)), \\ T_k(x_k) & = & \mathrm{pgcd}(p_k(x_k),m_{x_k}(x_k)). \end{array}\right.$$

Démonstration. La preuve est une récurrence décroissante, pour $k = n,\ldots,1$. Pour $k = n$, la propriété est vraie, montrons le passage de $k+1$ à k.

Fixons un plongement de Q_{k+1} dans une clôture algébrique de $k(p_1,\ldots,p_m)$. Nous savons que T_k est un polynôme à racines simples, dont les racines sont les valeurs que prend x_k au-dessus du point de coordonnées $(x_{k+1},\ldots,x_n)$.

À l'issue de la première série de calculs de pgcd, le polynôme $p_{k+1,k}(t)$ a pour racines les valeurs que prend t parmi les points décrits par $\mathcal{R}(t)$ dont les dernières coordonnées sont $(x_{k+1},\ldots,x_n)$.

Le résultant $p_k(t)$ est un polynôme dont les racines sont les valeurs que prend x_k sur ces points. La partie sans carré de ce polynôme est la réduction du polynôme T_k recherché. Nous l'obtenons par le dernier calcul de pgcd : les racines de m_{x_k} sont simples, et parmi elles on trouve les valeurs qui nous intéressent (les autres racines de m_{x_k} sont associées aux points conjugués de $(x_{k+1},\ldots,x_n)$). □

Démonstration de la proposition. Soit $\mathbf{p}$ une valeur des paramètres satisfaisant les hypothèses de l'énoncé de la proposition. Par hypothèse sur $\mathbf{p}$, toutes les opérandes (w_k,q et m_{x_k} se spécialisent en $\mathbf{p}$, et q et les m_{x_k} sont sans racine multiple. Sous ces conditions, nous allons montrer que l'on peut suivre la trace du calcul précédent en spécialisant les valeurs des paramètres en $\mathbf{p}$.
Nous faisons une récurrence décroissante, pour $k = n,\ldots,1$; l'ensemble de définitions-propriétés que nous montrons est :

- La $(n-k+1)^{\mathrm{e}}$ étape du calcul présenté dans le lemme précédent, spécialisé en $\mathbf{p}$, est la trace du calcul sur $k(p_1,\ldots,p_m)$. En particulier, on ne crée pas de scindage. Cela nous permet de définir le polynôme t_k.
- Nous notons q_k (l'analogue de Q_k) le quotient de $k[x_k,\ldots,x_n]$ par les polynômes $(t_k,\ldots,t_n)$. Alors q_k est un produit de corps dont la somme des degrés sur k est égale au degré de Q_k sur $k(p_1,\ldots,p_m)$. Le quotient q_k décrit la projection de la fibre spécialisée $\mathcal{U}_{\mathbf{p}}$ sur le plan des $n-k+1$ dernières variables.

Pour $k = n$, la propriété est vraie : puisque l'on suppose que le polynôme minimal de x_n dans la fibre spécialisée a degré maximal, c'est la spécialisation du polynôme minimal générique.

Étudions le passage de $k+1$ à k. Le quotient q_{k+1} est un produit de corps. Il nous suffit de montrer que dans chacun de ces corps, le degré de chacun des calculs de pgcd est le degré du calcul générique. Sous ces conditions, puisque les autres opérandes se spécialisent, nous pourrons appliquer pour chacun des pgcd le lemme de spécialisation donné dans la sous-section suivante.

Nous montrons donc par récurrence décroissante sur $j = n,\ldots,k+1$ que le degré de chacun des pgcd $(p_{n,k},\ldots,p_{k+1,k})$ ne change pas, et qu'en conséquence leur calcul se spécialise. Pour montrer que les degrés des pgcd restent les mêmes à la spécialisation, nous factorisons *complètement* l'ensemble des générateurs du quotient q_{k+1} dans une clôture algébrique de k. Il suffit de montrer que dans chacun des corps correspondants, le degré du pgcd de change pas : cette factorisation rend le raisonnement géométrique plus commode. Nous noterons $(q_{k+1,l})_{l\in\mathcal{L}}$ les corps en question, auxquels sont associés des points de $\overline{k}^{n-k}$. Par hypothèse de récurrence, le cardinal de $\mathcal{L}$ est égal au degré de Q_{k+1} sur $k(p_1,\ldots,p_m)$.

- *Initialisation à $j = n$.* Le premier pgcd $p_{n,k}$ n'est fonction que de la valeur de x_n dans le quotient $q_{k+1,l}$. Pour chacune de ces valeurs, choisissons un représentant parmi les $(q_{k+1,l})_{l\in\mathcal{L}}$ et notons $(q_{k+1,l})_{l\in\mathcal{L}_n}$ leur réunion.

  Le cardinal de $\mathcal{L}_n$ est égal au degré du polynôme minimal de x_n dans le quotient q_{k+1}, donc au degré du polynôme minimal de x_n dans la fibre spécialisée. D'après nos hypothèses sur $\mathbf{p}$, ce degré est égal au degré du polynôme minimal de x_n sur $k(p_1,\ldots,p_m)$, c'est-à-dire $\deg_{k(p_1,\ldots,p_m)} Q_n$.

  Notons $\mathcal{D}_{n,l}$ le degré du pgcd calculé dans le corps $q_{k+1,l}$; il est égal au nombre de points dans la fibre spécialisée qui ont pour dernière coordonnée la valeur de x_n dans $q_{k+1,l}$. Notons ensuite $\mathcal{D}_n$ le degré du pgcd calculé dans le cas générique : le nombre de points de la fibre générique est égal à $\mathcal{D}_n \deg_{k(p_1,\ldots,p_m)} Q_n$.

Les deux opérandes du pgcd se spécialisent, et la seconde garde le même degré ; en conséquence, le degré du pgcd calculé dans chacun des corps ne peut que croître. Pour tout l nous avons donc l'inégalité $\mathcal{D}_{n,l} \geq \mathcal{D}_n$ et nous pouvons en déduire la suite d'inégalités :

$$\begin{aligned}\sum_{l\in\mathcal{L}_n} \mathcal{D}_{n,l} &\geq \mathcal{D}_n \# \mathcal{L}_n \\ &\geq \mathcal{D}_n \deg_{k(p_1,\ldots,p_m)} Q_n \\ &\geq \deg q.\end{aligned}$$

La somme en question est égale au nombre de points dans la fibre spécialisée, donc à $\deg q$. Aussi, chacune des inégalités $\mathcal{D}_{n,l} \geq \mathcal{D}_n$ doit être une égalité. En conséquence du lemme de spécialisation donné dans la sous-section suivante, le calcul se spécialise.

– *Passage de* $j+1$ *à* j. Dans le corps $q_{k+1,l}$, le pgcd $p_{j,k}$ est le produit des valeurs que prend t sur les points décrits par $\mathcal{R}(t)$ dont les dernières coordonnées sont les valeurs de $(x_j,\ldots,x_n)$ dans $q_{k+1,l}$. Notons $\mathcal{D}_{j,l}$ le degré du pgcd calculé dans le corps $q_{k+1,l}$.

  Choisissons pour chacun des jeux de valeurs de $(x_j,\ldots,x_n)$ possibles un représentant parmi les $(q_{k+1,l})_{l\in\mathcal{L}}$ et notons $(q_{k+1,l})_{l\in\mathcal{L}_j}$ leur réunion. Alors le cardinal de $\mathcal{L}_j$ est le cardinal de la projection de la fibre spécialisée sur le plan des variables $(x_j,\ldots,x_n)$. Puisque j est strictement plus grand que k, l'hypothèse de récurrence sur k assure que ce cardinal est égal au cardinal de la projection de la fibre générique sur le plan des variables $(x_j,\ldots,x_n)$, donc au degré $\deg_{k(p_1,\ldots,p_m)} Q_j$.

  Notons ensuite $\mathcal{D}_j$ le degré du pgcd $p_{j,k}$ calculé dans le cas générique. Par hypothèse de récurrence sur j, les deux opérandes du pgcd se spécialisent, et la seconde opérande garde un degré constant à la spécialisation. Le degré de chaque pgcd ne peut que croître : $\mathcal{D}_{j,l} \geq \mathcal{D}_j$ pour tout l. Enfin, le nombre de points de la fibre générique est égal à $\mathcal{D}_j \deg_{k(p_1,\ldots,p_m)} Q_j$. Nous pouvons alors écrire les égalités :

$$\begin{aligned}\sum_{l\in\mathcal{L}_j} \mathcal{D}_{j,l} &\geq \mathcal{D}_j \# \mathcal{L}_j \\ &\geq \mathcal{D}_j \deg_{k(p_1,\ldots,p_m)} Q_j \\ &\geq \deg q.\end{aligned}$$

  La somme est par ailleurs égale au degré de la fibre spécialisée, donc chaque inégalité $\mathcal{D}_{j,l} \geq \mathcal{D}_j$ est une égalité, ce que l'on voulait prouver.

Le calcul du résultant se spécialise, même si le degré en t de $w_k(t)$ chute. Regardons enfin le dernier calcul de pgcd. Dans chacun des corps $q_{k+1,l}$, le calcul produit un polynôme dont les racines sont les valeurs que prend x_k au-dessus du point de $\overline{k}^{n-k}$ associé à $q_{k+1,l}$. Ses racines sont simples, puisque par hypothèse sur $\mathbf{p}$, celles de m_{x_k} le sont.

Nous notons $\mathcal{D}'_{k,l}$ le degré du pgcd calculé dans le corps $q_{k+1,l}$, et $\mathcal{D}'_k$ le degré du pgcd calculé dans le cas générique. Les deux opérandes du pgcd se spécialisent en $\mathbf{p}$, et gardent même degré, nous avons donc pour tout l : $\mathcal{D}'_{k,l} \geq \mathcal{D}'_k$.

Le cardinal de la projection de la fibre générique sur le plan des coordonnées $(x_k,\ldots,x_n)$ est égal à $\deg_{k(p_1,\ldots,p_m)} Q_k$, donc au produit $\mathcal{D}'_k \deg_{k(p_1,\ldots,p_m)} Q_{k+1}$, et le cardinal de la projection de la fibre spécialisée sur le plan des coordonnées $(x_k,\ldots,x_n)$ lui est inférieur. Nous pouvons alors écrire les inégalités :

$$\begin{aligned}\sum_{l\in\mathcal{L}} \mathcal{D}'_{k,l} &\geq \mathcal{D}'_k \#\mathcal{L}\\ &\geq \mathcal{D}'_k \deg_{k(p_1,\ldots,p_m)} Q_{k+1}.\end{aligned}$$

La somme est égale au cardinal de la projection de la fibre spécialisée sur le plan des coordonnées $(x_k,\ldots,x_n)$, toutes les inégalités $\mathcal{D}'_{k,l} \geq \mathcal{D}'_k$ sont donc des égalités. Nous pouvons de nouveau appliquer le lemme de spécialisation du pgcd, et obtenons le polynôme t_k.
Le calcul spécialisé est la trace du calcul générique. La réduction du polynôme t_k est sans racine multiple dans chacun des corps $q_{k+1,l}$. Le quotient $q_{k+1}[x_k]/t_k$ est un produit de corps, le nombre de points décrits par ce quotient est égal à $\deg_{k(p_1,\ldots,p_m)} Q_k$ et ceux-ci sont les projections des points de la fibre spécialisée. □

Corollaire 9 *Il existe un polynôme $\mathcal{D}isc$ de $k(p_1,\ldots,p_m)$ tel que toute valeur $\mathbf{p}$ n'annulant pas $\mathcal{D}isc$ satisfait toutes les conditions de généricité. Le polynôme $\mathcal{D}isc$ peut être pris de degré inférieur à*

$$d^n_{\mathbf{p},\mathbf{x}}(2(n+1)d^n_{\mathbf{p},\mathbf{x}} + nd_{\mathbf{p},\mathbf{x}}).$$

Démonstration. Les conditions données ci-dessus sur $\mathbf{p}$ se traduisent par la non-annulation en du résultant du polynôme éliminant générique avec sa dérivée, du résultant du polynôme éliminant de chaque variable avec sa dérivée, et du déterminant jacobien. D'après les bornes énoncées dans la section 8.4 de la section 8, le produit de ces polynômes est de degré inférieur à

$$d^n_{\mathbf{p},\mathbf{x}}(2(n+1)d^n_{\mathbf{x}} + nd_{\mathbf{p},\mathbf{x}}).$$

□

Remarque On constate un phénomène analogue lors de la mise d'une représentation univariée sous forme “Shape Lemma” : les dénominateurs peuvent avoir des degrés importants, mais le lieu de leurs zéros est relativement bien contrôlé, à partir du discriminant du polynôme minimal.

18.1.2 Lemmes de spécialisation

Lemme 17 (Spécialisation de l'inverse) *Soient k un corps et $(F_1,\ldots,F_n)$ des polynômes de l'anneau $k(p_1,\ldots,p_m)[x_1,\ldots,x_n]$, tels que F_k est unitaire en x_k, ne dépend que des variables $(x_k,\ldots,x_n)$ et est réduit par rapport aux polynômes $F_{k+1},\ldots,F_n$. Soit Q le quotient*

$$k(p_1,\ldots,p_m)[x_1,\ldots,x_n]/(F_1,\ldots,F_n),$$

que l'on suppose être un produit de corps Q_i. Soit ensuite A un élément de $k(p_1,\ldots,p_m)[x_1,\ldots,x_n]$, on suppose que A est inversible dans Q, et on note B l'inverse de A, écrit dans la base canonique de Q.
Soit $\mathbf{p}$ un élément de k^m ; on suppose que $\mathbf{p}$ n'annule aucun des dénominateurs des coefficients des polynômes $(F_1,\ldots,F_n,A)$ et on note $(f_1,\ldots,f_n,a)$ leur spécialisation en $\mathbf{p}$.
Si a est inversible dans q, d'inverse b, alors la spécialisation des coefficients de B en $\mathbf{p}$ est possible, et coïncide avec b.

Démonstration. On démontre la propriété par récurrence sur n :

- Si $n = 1$, l'inverse de A est donné par un calcul de pgcd étendu, comme solution d'un système linéaire dont la matrice $\mathbf{S}$ est la matrice de Sylvester associée à F_1 et A. De même, l'inverse de a est donné comme solution d'un système linéaire dont la matrice $\mathbf{s}$ est la matrice de Sylvester associée à f_1 et a. Le polynôme F_1 est unitaire, donc la réduction en $\mathbf{p}$ du déterminant de $\mathbf{S}$ est le déterminant de $\mathbf{s}$, éventuellement au signe près. Celui-ci est non nul par hypothèse, et c'est le seul nouveau dénominateur qui peut apparaître. Dans les deux cas, les cofacteurs sont donnés par les formules de Cramer, et on vérifie que même si le degré de A chute à la spécialisation, ces calculs se spécialisent, au signe mentionné précédemment près.
- Supposons la propriété vraie au rang n, passons au rang $n+1$. On considère Q comme le quotient univarié

$$\big(k(p_1,\dots,p_m)[x_1,\dots,x_{n-1}]/(f_1,\dots,f_{n-1})\big)[x_n]/(f_n).$$

  La démarche est la même que dans le cas $n = 1$: on construit les deux matrices de Sylvester $\mathbf{S}$ et $\mathbf{s}$, qui permettent de calculer les inverses B et b par résolution d'un système linéaire. Les calculs des cofacteurs qui donnent le numérateur de l'inverse se font par les formules de Cramer et ne font pas intervenir de nouveau dénominateur en $(p_1,\dots,p_m)$; à nouveau, ils se spécialisent bien, même si le degré de A chute à la spécialisation. Pour conclure, il suffit d'appliquer l'hypothèse de récurrence au déterminant de $\mathbf{S}$.

□

Corollaire 10 (Spécialisation du pgcd) *Sous les mêmes hypothèses que précédemment, soient A et B des éléments de $Q[t]$, avec A unitaire. Supposons qu'il existe un entier d tel que dans chacun des $Q_i[t]$, le degré du pgcd de A et B est égal à d. Notons R le pgcd de A et B dans $Q[t]$, où les coefficients sont pris dans la base canoniquement associée à $(F_1,\dots,F_n)$.*
Soit $\mathbf{p}$ un élément de k^m ; on suppose que $\mathbf{p}$ n'annule aucun des dénominateurs des coefficients des polynômes $(A,B,F_1,\dots,F_n)$, et on note $(a,b,f_1,\dots,f_n)$ leur spécialisation en $\mathbf{p}$.
Supposons que $q = k[x_1,\dots,x_n]/(f_1,\dots,f_n)$ est un produit de corps q_i et que dans chacun des q_i, le pgcd de a et b a degré d. Alors le point $\mathbf{p}$ n'annule le dénominateur d'aucun coefficient de R, et le pgcd de a et b est la spécialisation de R en $\mathbf{p}$.

Démonstration. Le pgcd R de A et B dans $Q[t]$ est donné par la formule

$$R = \mathrm{Cof}_a A + \mathrm{Cof}_b B.$$

Notons d_A et d_B les degrés de A et B, et appelons $\mathbf{S}_d$ la $d$ième matrice de sous-résultants associée à A et B (voir l'annexe C). Les coefficients des cofacteurs $\mathrm{Cof}_{a,b}$ sont les solutions du système

$$\mathbf{S}_d \begin{bmatrix} c_1 \\ \vdots \\ c_{d_A+d_B-2d} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}.$$

Notons maintenant d_a et d_b le degré de a et b, et spécialisons en $\mathbf{p}$ les coefficients de $\mathbf{S}_d$. Le polynôme A est unitaire, donc la sous-matrice inférieure gauche de taille $d_a + d_b - 2d$ extraite de cette spécialisation est la $d$ième matrice des sous-résultants $\mathbf{s}_d$ associée à a et b. De plus, les $m - d$ premiers coefficients de la diagonale de $\mathbf{S}_d$ sont des 1. Ainsi, le déterminant de $\mathbf{s}_d$ est, à un signe près, la spécialisation de celui de $\mathbf{S}_d$.

Ces deux déterminants sont inversibles, et on peut leur appliquer le lemme précédent. Les numérateurs des cofacteurs sont obtenus par les formules de Cramer appliquées au système précédent. Ces calculs se spécialisent, au signe mentionné ci-dessus près. Ceci donne le résultat. □

Remarque Il n'apparaît que partiellement dans les preuves ci-dessus que les questions d'inverses et de pgcd dans une tour d'extensions sont fortement liées. On pourra consulter [154] pour une présentation plus détaillée de cette question.

18.2 Borne en degré

Dans [72], Gallo et Mishra donnent une borne sur le degré des polynômes formant un *ensemble caractéristique* d'un idéal en fonction d'une borne sur les degrés des générateurs. Ils se basent sur des résultats de division, à partir du Nullstellensatz effectif de Kollár [122], et du théorème de Bézout de Heintz, le dernier ingrédient étant de l'algèbre linéaire. Une argumentation similaire à la leur, mais basée sur le lemme de division plus "intrinsèque" 12 du chapitre 4, permet de conserver des bornes du même ordre de grandeur pour les degrés des expressions en $(p_1,\dots,p_m)$ apparaissant dans *chacun* des $\mathbf{t}^{(j)}$.

Théorème 22 (Borne en degré d'après Gallo et Mishra) *Pour tout j, les coefficients des polynômes de $\mathbf{t}^{(j)}$ ont des numérateurs et dénominateurs de degrés inférieurs à*

$$d_{\mathbf{p},\mathbf{x}}^n(2((n+1)^2+1)d_{\mathbf{x}}^{2n}+n+1)^{n+1}.$$

Démonstration. Nous oublions les indices (j), puisque nous allons travailler dans une seule composante. Soit $\mathcal{R}(t)=(t,q,\mathbf{v})$ une résolution de Kronecker de la composante décrite par $\mathbf{t}$. Après en avoir chassé tous les dénominateurs, nous obtenons des polynômes de $k[p_1,\dots,p_m,x_1,\dots,x_n,t]$ de degré au plus $\mathcal{D}_{\mathbf{p},\mathbf{x}}$, d'après les résultats du chapitre 8.
Travaillons sur le corps des fractions rationnelles en $(p_1,\dots,p_m)$. Soit $\mathcal{I}$ l'idéal

$$\big(q(t),(q'(t)x_i-v_i(t))_{i=1,\dots,n}\big)$$

de $k(p_1,\dots,p_m)[x_n,\dots,x_1,t]$. Dans cet ordre, la suite des générateurs forme une suite régulière de $k(p_1,\dots,p_m)[x_n,\dots,x_1,t]$ satisfaisant les hypothèses de la proposition 12 du chapitre 4 ; les degrés des variétés intermédiaires

$$\big(\mathcal{V}(q),\dots,\mathcal{V}(q,\dots q'x_n-v_n)\big)$$

sont tous bornés par $\mathcal{D}_{\mathbf{x}}$, donc par $d_{\mathbf{x}}^n$. Chaque polynôme t_k est dans l'idéal $\mathcal{I}$, il existe donc des coefficients de Bézout $p_{i,k}$ tels que

$$t_k=p_{0,k}q+\sum_{i=1}^n p_{i,k}(q'x_i-v_i).$$

D'après la proposition 12 du chapitre 4 et les bornes sur le degré en $\mathbf{x}$ des t_k, le degré en $(x_1,\dots,x_n,t)$ de chaque terme de la somme peut être pris inférieur à

$$2((n+1)^2+1)d_{\mathbf{x}}^{2n}.$$

Dès lors, on peut reprendre exactement l'argumentation de [72, section 3.2.2]. Écrivons le système $k(p_1,\dots,p_m)$-linéaire satisfait par les coefficients des polynômes t_j et $p_{i,j}$ et appelons Γ le nombre de monômes de degré au plus $2((n+1)^2+1)d_{\mathbf{x}}^{2n}$ en $n+1$ variables : $\Gamma=\mathcal{T}(2((n+1)^2+1)d_{\mathbf{x}}^{2n},n+1)$.

Alors les coefficients des polynômes t_j s'écrivent comme des quotients de déterminants de matrices dont les entrées sont des polynômes de degrés au plus $\mathcal{D}_{\mathbf{p},\mathbf{x}}$, et qui ont une taille au plus Γ. Ces coefficients sont donc des fractions rationnelles ayant des numérateurs et dénominateurs de degrés au plus

$$\mathcal{D}_{\mathbf{p},\mathbf{x}}\Gamma \leq d_{\mathbf{p},\mathbf{x}}^n(2((n+1)^2+1)d_{\mathbf{x}}^{2n}+n+1)^{n+1}.$$

$\square$

Remarque On pourrait de même donner une borne sur les hauteurs des éléments de $\mathbf{t}$, qui serait elle aussi de l'ordre de $hd_{\mathbf{p},\mathbf{x}}^{n^2}$.

18.3 On recherche une borne intrinsèque

À la suite de la section 8.3 du chapitre 8, nous voudrions pouvoir donner une borne plus intrinsèque sur les degrés en $(p_1,\ldots,p_m)$ des numérateurs et dénominateurs des coefficients des éléments des $\mathbf{t}^{(j)}$, quitte à multiplier les termes de tête par des unités. Ainsi, le passage d'une représentation "Shape Lemma" à une représentation de Kronecker se fait en multipliant par un initial qui s'avère être naturel, la dérivée du polynôme minimal. La suite est un premier (petit) pas dans cette direction, dans le domaine des ensembles triangulaires.

18.3.1 Proposition pour $n = 2$

Hypothèse 4 *Nous travaillerons sous les hypothèses simplificatrices suivantes :*

- *Nous nous intéressons uniquement aux deux dernières variables x_{n-1} et x_n ; on peut donc supposer que $n = 2$, et considérer les polynômes $t_1(x_1,x_2)$ et $t_2(x_2)$ à coefficients dans $k(p_1,\ldots,p_m)$, de degrés en leur variable principale d_1 et d_2.*
- *Nous supposons que la fibre générique peut être décrite en entier par un seul ensemble triangulaire. C'est une hypothèse de commodité, on pourrait s'en passer, en travaillant uniquement dans une des composantes de la fibre.*

Notons $(x_2^{(1)},\ldots,x_2^{(d_2)})$ les racines de t_2 dans une clôture algébrique de $k(p_1,\ldots,p_m)$. Au-dessus de la racine $x_2^{(i)}$, nous notons $(x_1^{(i,1)},\ldots,x_1^{(i,d_1)})$ les racines de $t_1(x_1,x_2^{(i)})$ dans la même clôture. Les $x_1^{(i,j)}$ ne sont pas des fonctions algébriques des $x_2^{(i)}$.
Notons S_k la $k$ème somme de Newton du polynôme t_1, et $S_k(i) = \sum_{j=1}^{d_1}(x_1^{(i,j)})^k$ son évaluation en $x_2^{(i)}$. Nous allons donner une expression des S_k comme fonction de x_2, avec une borne sur le degré des coefficients, dont on tirera des informations sur le degré des fonctions symétriques élémentaires de t_1.

Notations Soit ℓ une nouvelle variable et étendons k en $k(\ell)$. Nous noterons :

- $\mathrm{m}_{x_2}(t) = t_2(t)$ le polynôme minimal de la multiplication par x_2 dans la fibre générique,
- $\mathrm{m}_{x_2+\ell x_1^k}(t)$ le polynôme minimal de la multiplication par $x_2+\ell x_1^k$ dans la fibre générique,
- $\mathrm{m}_{x_2+\ell S_k}(t)$ le polynôme minimal de la multiplication par $x_2+\ell S_k$ dans la fibre générique.

À l'instar de la démonstration de 8.3 du chapitre 8, la donnée de $\mathrm{m}_{x_2+\ell S_k}$ modulo ℓ^2 permet de donner S_k en fonction de x_2 :

Proposition 63 *La k^{e} somme de Newton S_k est donnée par :*

$$\frac{\partial m_{x_2}}{\partial t}S_k = -\mathrm{Coeff}(m_{x_2+\ell S_k},\ell,1).$$

Démonstration. Pour tout ℓ, $\mathrm{m}_{x_2+\ell S_k}(t)$ est nul dans la fibre générique, donc sa dérivée par rapport à ℓ l'est également. Cette dérivée est $\frac{\partial \mathrm{m}_{x_2+\ell S_k}}{\partial t} S_k + \frac{\partial \mathrm{m}_{x_2+\ell S_k}}{\partial \ell}$. Spécialisons en $\ell = 0$; $\mathrm{m}_{x_2+\ell S_k}$ se spécialise en m_{x_2} : puisque S_k est fonction de x_2, le polynôme minimal de $x_2 + \ell S_k$ a le même degré que celui de x_2. On trouve donc :

$$\frac{\partial \mathrm{m}_{x_2}}{\partial t} S_k = -\mathrm{Coeff}(\mathrm{m}_{x_2+\ell S_k},\ell,1).$$

□

On peut calculer $\mathrm{m}_{x_2+\ell S_k}$ à précision 2 à partir des polynômes minimaux que l'on a introduits :

Proposition 64 *On a l'égalité*

$$m_{x_2+\ell x_1^k} = m_{x_2}^{d_1-1} m_{x_2+\ell S_k} \mod \ell^2.$$

Démonstration. Posons

$$\vartheta_k^{(i)} = \prod_{j=1}^{d_1} (t - x_2^{(i)} - \ell(x_1^{(i,j)})^k).$$

Alors le polynôme $\mathrm{m}_{x_2+\ell x_1^k}$ est égal à

$$\prod_{i=1}^{d_2} \vartheta_k^{(i)}.$$

Au deuxième ordre en ℓ, $\vartheta_k^{(i)}$ est égal à $(t - x_2^{(i)})^{d_1-1}(t - x_2^{(i)} - \ell S_k(i))$ et le produit des $\vartheta_k^{(i)}$ est égal à $\mathrm{m}_2(t)^{d_1-1} \prod (t - x_2^{(i)} - \ell S_k(i))$, ce qui est l'égalité voulue. □

Corollaire 11 *Le coefficient de degré 1 en ℓ de $\mathrm{m}_{x_2+\ell S_k}$ peut s'écrire comme un polynôme de $k(p_1,\dots,p_m)[t]$, de degré $d_2 - 1$ en t, dont les coefficients ont des numérateurs de degré au plus $k\mathcal{D}_{\mathbf{p},\mathbf{x}}$, et pour dénominateur un polynôme d_k de degré au plus $k\mathcal{D}_{\mathbf{p},\mathbf{x}}$.*

Démonstration. D'après la proposition 27 du chapitre refchapter:degres, les coefficients de $\mathrm{m}_{x_2+\ell x_1^k}$ sont des fractions rationnelles dont les numérateurs et dénominateurs ont un degré au plus $k\mathcal{D}_{\mathbf{p},\mathbf{x}}$. Les dénominateurs sont indépendants de ℓ, d'après la section 8.2.2 du chapitre 8. En faisant $\ell = 0$ dans l'égalité de la proposition précédente, on voit que le dénominateur de m_{x_2} à la puissance $d_1 - 1$ divise le dénominateur de $\mathrm{m}_{x_2+\ell x_1^k}$. On divise ensuite le coefficient de degré 1 en ℓ de $\mathrm{m}_{x_2+\ell x_1^k}$ par le numérateur de m_{x_2} pour obtenir le résultat. □

Conclusion Le résultat précédent suggère qu'un "bon" initial pour la $k$ème somme de Newton S_k est la dérivée du polynôme minimal m_{x_2}, puisque le degré en $(p_1,\dots,p_m)$ des coefficients de $\mathrm{Coeff}(\mathrm{m}_{x_2+\ell S_k},\ell,1)$ est borné par des grandeurs d'origine géométrique.

On peut écrire les coefficients de t_1 comme des sommes de produits des sommes de Newton. Rappelons que d_1 désigne le degré de t_1 ; alors le coefficient de degré $d_1 - k$ de t_1 s'exprime en particulier à l'aide de la puissance $k$ème de la première somme de Newton. Cette écriture donne donc comme dénominateur pour ce coefficient la *puissance $k$ème de la dérivée du polynôme minimal* m_{x_2}. Cette construction généralise l'emploi du dénominateur q' dans une résolution univariée sous forme de Kronecker.

18.3.2 Considérations empiriques

L'expérience semble indiquer que le résultat précédent peut être amélioré. Nous regroupons ici les résultats de divers calculs faisant intervenir des ensembles triangulaires. Ici, on ne s'intéresse pas aux temps de calcul ni aux méthodes employées, mais uniquement à la taille de la sortie pour différents choix des initiaux.

Les systèmes évoqués ci-dessous seront donnés dans les annexes E.8, E.5 et E.9. Ce sont des systèmes dépendant de paramètres, pour lesquels une variable n'est pas élément séparant. Dans chaque cas, nous n'avons retenu que le comportement des deux dernières variables. Pour ces petites expériences, il est plus commode de travailler avec des nombres rationnels que dans un corps de paramètres, aussi avons-nous spécialisé les paramètres en une valeur entière ; la tableau ci-dessous indique la *hauteur* des coefficients des polynômes qui apparaissent après mise sous forme triangulaire, pour différents choix des initiaux. Comme l'ont montré les parties précédentes, la hauteur des coefficients ou leur degré en les paramètres suivent sensiblement les mêmes lois.

Les colonnes désignent respectivement

- les degrés des polynômes $(t_1(x_1,x_2),t_2(x_2))$ en leur variable principale ;
- la hauteur (décimale) des numérateurs et des dénominateurs du coefficient constant du polynôme t_1, pour une représentation sans initiaux : le polynôme est unitaire, tel qu'il sort par exemple d'un calcul de base de Gröbner;
- la hauteur du même coefficient, multiplié par le *carré* de la dérivée du polynôme t_2. Nous suivons ici les résultats de la partie précédente, puisqu'ici la variable x_1 est toujours de degré 2 sur la variable x_1.
- la hauteur (décimale) des numérateurs et des dénominateurs du coefficient constant du polynôme t_1 multiplié par dérivée du polynôme t_2.

Système	Degrés (d_1,d_2)	Hauteur Sans initial	Hauteur $(\min'_{x_2})^2$	Hauteur $\min'_{x_2}$
Bershenko	(2,12)	(97,95)	(53,7)	(27,5)
$j_1j_2j_3$	(2,7)	(47,39)	(68,11)	(35,7)
Hawes4	(2,60)	(2192,2192)	(383,218)	(121,50)

Conclusion Le résultat indiqué par la section précédente permet de réduire la hauteur des coefficients. Cependant, l'idée plus simple qui consiste à multiplier par la dérivée du polynôme minimal de x_2, sans élévation de puissance, fournit des résultats plus probants sur ces exemples. Je ne sais pas expliquer pourquoi c'est le cas.

Chapitre 19

Calcul d'un polynôme minimal

Résumé

Dans ce chapitre, nous nous intéressons au calcul du *polynôme minimal* de la variable x_n, par des calculs mettant en jeu des ensembles triangulaires à coefficients dans $k(p_1,\ldots,p_m)$ qui représentent les zéros simples génériques de $(f_1,\ldots,f_n)$. Nous proposons un algorithme probabiliste qui calcule ce polynôme, à partir de la donnée d'une résolution spécialisée en un point chanceux.

La complexité du processus de remontée est linéaire en la complexité d'évaluation du système $\mathbf{f}$, polynomiale en m et en la taille de la sortie, et dépend d'un facteur exponentiel en n lié au coût du calcul modulo un ensemble triangulaire. Les valeurs des paramètres qui mènent à l'échec sont incluses dans une hypersurface algébrique dont le degré est borné à partir des résultats du chapitre précédent.

Sommaire

Hypothèses et résultats

Nous cherchons désormais à calculer le polynôme minimal de la variable x_n, par l'intermédiaire d'ensembles triangulaires $\mathbf{t}^{(j)} = (t_1^{(j)},\ldots,t_n^{(j)})$ représentant les zéros simples génériques de $(f_1,\ldots,f_n)$; nous supposons en particulier que ces ensembles satisfont les conditions données dans le chapitre 17. Le produit sur j des polynômes $t_n^{(j)}$ distincts est alors le polynôme minimal "générique" de x_n. C'est le seul de ces polynômes pour lequel on dispose d'une borne "raisonnable" sur le degré de ses coefficients, c'est donc le seul que l'on va chercher à calculer.

Hypothèse 5 *Nous supposons en outre que pour toute valeur* $\mathbf{p}$ *des paramètres satisfaisant toutes les hypothèses de généricité, il est possible de calculer une représentation des zéros simples du système* $\mathbf{f}(\mathbf{p},.)$ *par un nombre fini d'ensembles triangulaires, décrivant des idéaux premiers, en* $\mathcal{C}_{Zéro}(\mathbf{f},\mathbf{p},k)$ *opérations de* k.

Pour énoncer le résultat de ce chapitre, il faut une notation supplémentaire : nous supposerons que l'algorithme de multiplication de polynômes univariés sur lequel sont basés les calculs a une complexité en $O_{\log}(d^\alpha)$ pour des polynômes de degré d. D'après le chapitre 5, on peut prendre $\alpha = 1$. La constante universelle C a été définie dans le du chapitre 5, elle désigne un facteur exponentiel lié au calcul dans un empilement de quotients univariés.

Théorème 23 (Complexité du calcul du polynôme minimal) *Soit* $m_{x_n} \in k(p_1,\ldots,p_m)[t]$ *le polynôme minimal générique de* x_n. *Notons* d_n *son degré en* t *et* $\mathcal{D}(x_n)$ *le maximum des degrés en* $(p_1,\ldots,p_m)$ *des numérateurs et dénominateurs de ses coefficients.*
On peut calculer le polynôme m_{x_n} *par un algorithme probabiliste en*

$$\mathcal{C}_{Zéro}(\mathbf{f},\mathbf{p}_1,k) + \mathcal{C}_{Zéro}(\mathbf{f},\mathbf{p}_2,k) + O_{log}\big((n^\Omega + nL)\mathcal{M}_s(4\mathcal{D}(x_n),m)(C^n\mathcal{D}_{\mathbf{x}}^\alpha + md_n)\big)$$

opérations de k, *où* $\mathbf{p}_1$ *et* $\mathbf{p}_2$ *sont deux valeurs des paramètres. Les choix des valeurs aléatoires intervenant dans l'algorithme qui mènent à l'échec sont enfermés dans une hypersurface algébrique de* $\mathbb{A}^{3m-1}(\overline{k})$ *de degré au plus* $80nd_{\mathbf{x}}^n d_{\mathbf{p},\mathbf{x}}^{3n}$.

Cela signifie que le nombre d'opérations de la phase de remontée est en particulier "polynomial en la taille de la sortie" ; notons également la présence du terme exponentiel C^n, lié au choix du calcul modulo un "empilement" de quotients univariés.

Une version spécialisée pour les rationnels. Supposons que k est le corps des rationnels, et que les polynômes $(f_1,\ldots,f_n)$ sont à coefficients entiers de hauteur bornée par h. Notons $\mathcal{H}(x_n)$ la hauteur des coefficients du polynôme minimal de x_n ; elle est majorée dans la proposition 43 du chapitre 13, par une borne de Bézout arithmétique.

Par une démarche analogue à celle du chapitre 15, nous pouvons énoncer le théorème suivant, donné sans démonstration. À la différence du chapitre 15, il faut demander que le polynôme minimal de *chaque* variable reste sans carré lors des diverses réductions. C'est la seule différence notable, elle induit une modification des valeurs explicites des premiers de réduction, mais pas de leur comportement asymptotique. Le résultat global de complexité reste le même :

Théorème 24 *On peut calculer le polynôme minimal générique de* x_n *par un algorithme probabiliste basé sur un processus de remontée de Newton, où le coût de la phase de remontée est*

$$O_{log}\big((n^\Omega + nL)\mathcal{M}_s(4\mathcal{D}(x_n),m)(C^n\mathcal{D}_{\mathbf{x}}^\alpha + md_n)\mathcal{M}_{\mathbb{N}}(\mathcal{H}(x_n) + \log h + mn\log d_{\mathbf{p},\mathbf{x}})\big)$$

opérations binaires avec une probabilité d'erreur inférieure à $1/2$.

En clair, le temps de calcul devient quasiment linéaire en la taille des entiers de la sortie.

Calcul local. Tout comme lors du calcul d'une résolution de Kronecker paramétrée, il est possible de ne travailler que sur une composante de la fibre générique : il suffit de la sélectionner lors de la résolution spécialisée et du test d'arrêt. L'analyse de complexité et la probabilité d'erreur ne sont pas modifiées.

Vérification. Enfin, le principe de vérification probabiliste des chapitres 10 et 15 s'appliquerait à cette nouvelle situation, puisqu'il repose uniquement sur des tests d'égalité ou de divisibilité de polynômes minimaux. Nous n'avons pas encore mis en œuvre cette possibilité, qui ne pose cependant pas de difficulté particulière.

19.1 Description de l'algorithme

L'algorithme suit le principe général déjà rencontré dans les chapitres 10 et 15.

- On calcule une résolution des zéros simples au-dessus d'une valeur aléatoire des paramètres $\mathbf{p}_1 = (p_{1,1}, \dots, p_{1,m})$ par une famille d'ensembles triangulaires irréductibles.
- Après avoir effectué le changement de variables $p'_k = p_k - p_{1,k}$, on applique l'opérateur de remontée de Newton modulo les puissances successives de l'idéal $(p'_1, \dots, p'_m)$ de $k[[p'_1, \dots, p'_m]]$, à partir d'une sous-famille $(\mathbf{t}^{(j)} = (t_1^{(j)}, \dots, t_n^{(j)}))_{j \in J}$ qui sert de valeur initiale ;
- Au terme de chaque étape de remontée de Newton, le produit des polynômes $t_n^{(j)}$ fournit une approximation du polynôme minimal de x_n à coefficients dans les séries $k[[p'_1, \dots, p'_m]]$. On tente de reconstruire ses coefficients ; le polynôme minimal de x_2 au-dessus d'un paramètre $\mathbf{p}_2$ sert alors de témoin pour décider de l'arrêt du processus de remontée.

Nous donnons tout d'abord le schéma d'ensemble de l'algorithme, les sections suivantes décrivent les sous-routines :

- `ResolutionTriangulaire` (résolution de la fibre spécialisée) ;
- `RemonteeNewton` ;
- `Arret` (test d'arrêt de la remontée).

19.2 Résolution de la fibre spécialisée

D'après la première section du chapitre précédent, les points $\mathbf{p}_1$ ne satisfaisant pas toutes les hypothèses de généricité sont enfermés dans une hypersurface de $\mathbb{A}^m(\overline{k})$ de degré inférieur à $d_{\mathbf{p},\mathbf{x}}^n(2(n+1)d_{\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}})$.

Supposons que $\mathbf{p}_1$ est un point chanceux. Nous utilisons une boîte noire notée `Resolution-Triangulaire`$(\mathbf{f}(\mathbf{p}_1,.))$ pour calculer une représentation de l'ensemble des zéros simples du système $\mathbf{f}(\mathbf{p}_1,.)$, par un ensemble d'ensembles triangulaires irréductibles. Pour y parvenir, une possibilité est de calculer une résolution de Kronecker de la fibre spécialisée et factoriser son polynôme minimal, pour effectuer l'algorithme `Triangular` [138] à partir de chacun des facteurs.

Les ensembles triangulaires dont on dispose à l'issue de ce calcul peuvent être davantage factorisés que les ensembles $\mathbf{t}^{(j)}$ génériques, et donc ne pas en être les spécialisations. Puisque la spécialisation des $\mathbf{t}^{(j)}$ est possible en $\mathbf{p}_1$, cette factorisation supplémentaire n'empêchera pas le processus de remontée de Newton. Nous ne retenons qu'une sous-famille $(\mathbf{t}_0^{(j)})_{j\in J}$ de ces ensembles, telle que le produit des derniers polynômes $(t_{0,n}^{(j)})_{j\in J}$ est égal au polynôme minimal de x_n au point $\mathbf{p}_1$: cela

FIG. 19.1 – *Polynôme minimal dans la fibre générique*

procedure `ResolutionTriangulaireGenerique`($\mathbf{f}$)

\# **Entree**
\# le système $\mathbf{f}$
\# **Sortie**
\# m_{x_n} polynôme minimal générique de x_n

$\mathbf{p}_1$, $\mathbf{p}_2$ $\leftarrow$ Paramètres aléatoires de Ω^m;
$\gamma \leftarrow$ Changement de variables aléatoire de Γ^{m-1};
b_1,$\mathbf{t}_0^{(j)}$ $\leftarrow$ `ResolutionTriangulaire`($\mathbf{f}(\mathbf{p}_1,.)$);
b_2,$\mathbf{t}_{\text{temoin}}^{(j)}$ $\leftarrow$ `ResolutionTriangulaire`($\mathbf{f}(\mathbf{p}_2,.)$);
if not (b_1 **and** b_2) **then return** "echec" **fi**;
$m_{\text{temoin}} \leftarrow$ Polynôme minimal de x_n dans $\mathbf{t}_{\text{temoin}}^{(j)}$;
$\mathcal{I} \leftarrow (p_1 - p_{1,1}, \ldots ,p_m - p_{1,m})$;
return `RemonteeNewton`($\mathbf{f}$,$\mathcal{I}$,($\mathbf{t}_0^{(j)}$),`Arret`(m_{temoin},$\mathbf{p}_2$,γ),$4d^m_{\mathbf{p},\mathbf{x}}$);
end;

suffira pour reconstruire le polynôme minimal générique. Nous noterons enfin $d_k^{(j)}$ le degré en x_k du $k$ème polynôme de $\mathbf{t}_0^{(j)}$.

Remarque Si l'on cherchait à calculer *tous les polynômes* des ensembles $\mathbf{t}^{(j)}$, il faudrait conserver tous les ensembles triangulaires spécialisés, et surtout les recombiner, afin d'éviter les problèmes de mauvaise factorisation, typiques des algorithmes de remontée à la Newton-Hensel [210]. Nous ne détaillerons pas la manière d'effectuer une telle recombinaison, qui suppose par exemple l'introduction d'une représentation canonique des zéros : pour reconstituer le polynôme minimal de la dernière variable, ce n'est pas nécéssaire.

19.3 Remontée des séries par Newton-Hensel

La proposition suivante assure que les conditions d'application de l'opérateur de Newton donné dans le chapitre 6 sont remplies :

Proposition 65 *Pour chacun des ensembles $\mathbf{t}^{(j)}$, il existe une matrice $\mathbf{A}^{(j)}$ à coefficients dans $k(p_1, \ldots ,p_m)$ satisfaisant l'égalité*

$$\mathbf{f} = \mathbf{A}^{(j)}\mathbf{t}^{(j)},$$

et dont les dénominateurs sont des produits des dénominateurs de $\mathbf{t}^{(j)}$.

Démonstration. Dans $k(p_1, \ldots ,p_m)$, le système $\mathbf{f}$ est réduit à zéro par chaque ensemble $\mathbf{t}^{(j)}$. Comme les polynômes formant $\mathbf{t}^{(j)}$ sont unitaires en leur variable principale, la réduction ne fait pas apparaître de nouveau dénominateur. □

Sous l'hypothèse d'une spécialisation chanceuse de $\mathbf{p}_1$, nous pouvons donc appliquer l'algorithme `RemonteeNewton` présenté dans le chapitre 6.

– Après avoir effectué le changement de variables $p'_k = p_k - p_{1,k}$, on travaille modulo les puis-

sances successives de l'idéal $\mathcal{I} = (p'_1, \dots ,p'_m)$ de l'anneau $\mathcal{A} = k[[p'_1, \dots ,p'_m]]$.
- On remonte en parallèle l'ensemble des $\mathbf{t}_i^{(j)}$, pour j dans J.
- Pour tout j, le point initial est l'ensemble $\mathbf{t}_0^{(j)}$ calculé au-dessus de $\mathbf{p}_1$; les approximations suivantes $\mathbf{t}_i^{(j)}$ sont à coefficients dans les séries $k[[p'_1, \dots ,p'_m]]$ à précision 2^i.
- Le critère d'arrêt est présenté dans la section suivante.
- Le nombre maximal d'étapes est fixé par la borne de Bézout sur le degré des coefficients du polynôme minimal de x_n.

Nous notons abusivement `RemonteeNewton`$(.,(\mathbf{t}_0^{(j)}),.)$ l'application de la procédure `RemonteeNewton` à l'ensemble des $(\mathbf{t}_0^{(j)})$.

Cette fois, le coût de la multiplication modulo l'ensemble $\mathbf{t}_i^{(j)}$ est $O(\mathcal{M}(d_1^{(j)}, \dots ,d_n^{(j)}))$. Nous en déduisons le coût total d'une étape de remontée.

Proposition 66 (Complexité d'une étape de Newton-Hensel) *Le coût de la ième étape de remontée est*

$$O\big((n^\Omega + nL)\mathcal{M}_s(2^i,m)\sum_{j\in J}\mathcal{M}(d_1^{(j)}, \dots ,d_n^{(j)})\big)$$

opérations de k.

Les remarques de la section 10.3 du chapitre 10 concernant les possibles accélérations de calcul restent valables.

Proposition 67 (Nombre d'étapes) *Si $\mathbf{p}_1$ satisfait toutes les hypothèses de généricité alors l'algorithme s'arrête après au plus $K = \lceil \log_2 \mathcal{D}(x_n)\rceil + 1$ étapes.*

Démonstration. Sous cette hypothèse, après k passages dans la boucle principale, nous avons une approximation du polynôme minimal de x_n à précision 2^k. Le degré des numérateurs et dénominateurs des coefficients que l'on recherche est borné par $\mathcal{D}(x_n)$, ce qui donne le résultat. □

19.4 Fonction d'arrêt

Le principe du test d'arrêt est calqué sur celui des fonctions analogues déjà rencontrées. Au terme de la $i$ème étape de remontée de Newton, le produit des polynômes $t_{i,n}^{(j)}$ donne une approximation du polynôme minimal de x_n à coefficients dans les séries formelles. Ce premier pas est suivi par la procédure de reconstruction rationnelle de ses coefficients puis par la comparaison au polynôme minimal témoin au-dessus de $\mathbf{p}_2$.

Reconstruction rationnelle de la résolution.

Proposition 68 *À la ième étape, le coût total de raconstruction rationnelle des coefficients du polynôme minimal de x_n est*

$$O_{log}(m\mathcal{M}_s(2^i,m)\sum_{j\in J}d_n^{(j)})$$

Si les reconstructions de tous les coefficients de m_{x_n} sont possibles, alors les changements de variables γ qui ne permettent pas ces reconstructions sont inclus dans une hypersurface de $\mathbb{A}^{m-1}(\overline{k})$ de degré inférieur à

$$d_n 2^{2i}(2^i + 1).$$

Démonstration. On reprend la preuve de la proposition 9.1 du chapitre 9, et cette fois on ne s'intéresse qu'au polynôme minimal m_{x_n}.

FIG. 19.2 – *Critère d'arrêt et reconstruction*

procedure `Arret`($\mathrm{m}_{\mathrm{Temoin}}$,$\mathbf{p}_2$,$\gamma$,$(\mathbf{t}_i^{(j)})$)

\# **Entree**
\# $\mathrm{m}_{\mathrm{Temoin}}$ polynôme minimal de x_n en $\mathbf{p}_2$
\# $\mathbf{p}_2 = (p_{2,1}, \ldots ,p_{2,m})$ valeur des paramètres
\# γ changement de variables
\# $(\mathbf{t}_i^{(j)})$: ensemble de résolutions à coefficients dans $k[[p'_1, \ldots ,p'_m]]$
\# **Sortie**
\# b,m_{x_n} : b booléen ; si b est vrai, m_{x_n} est le polynôme minimal de x_n

$\mathrm{m}_{x_n} \leftarrow \prod_{j \in J} t_n^{(j)}$;
b,$\mathrm{m}_{x_n} \leftarrow$ `ReconstructionRationnelle`(m_{x_n},γ);
if not b **then return false** ; **fi** ;
$\mathcal{J} \leftarrow (p_1 - p_{2,1}, \ldots ,p_m - p_{2,m})$;
$b \leftarrow$ `Temoin`($\mathrm{m}_{\mathrm{Temoin}}$,$\mathcal{J}$,$\mathrm{m}_{x_n}$);
if b **then return true**,m_{x_n}; **else return false** ; **fi** ;
end;

Comparaison à la résolution témoin. Nous utilisons l'analogue de la fonction `Temoin` du chapitre 9. Si le point $\mathbf{p}_2$ satisfait toutes les hypothèses de généricité, nous savons calculer le polynôme minimal de x_n en $\mathbf{p}_2$, et il suffit de comparer ce polynôme à la spécialisation du polynôme candidat.

Le test d'arrêt déclare que la résolution est la bonne si l'appel à la fonction `Temoin` répond “vrai”. Selon l'exemple de la section 10.4 du chapitre 10, on peut quantifier le risque d'erreur provoqué par ce test probabiliste : le test échoue si la reconstruction d'une des premières étapes est possible et si on décrète que la résolution est la bonne avant terme. C'est le cas si le point $\mathbf{p}_2$ est un *mauvais témoin* pour l'une des reconstructions intermédiaires. Pour chaque reconstruction intermédiaire, ces mauvais points sont quantifiés de manière analogue à la proposition 33 du chapitre 9 ; on peut les regrouper dans la proposition suivante :

Proposition 69 *Les points* $\mathbf{p}_2$ *satisfaisant toutes les hypothèses de généricité et mauvais témoins pour les reconstructions des étapes* $\{1, \ldots ,K-1\}$ *sont inclus dans une hypersurface de* $\mathbb{A}^m(\overline{k})$ *de degré au plus*

$$d_{\mathbf{p},\mathbf{x}}^n(\lceil \log_2(d_{\mathbf{p},\mathbf{x}}^n) \rceil + 2).$$

19.5 Complexité et probabilité d'erreur

Nous appellerons $M = M(n,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}})$ la somme des entiers :

- $16(4d_{\mathbf{p},\mathbf{x}}^n + 1)d_{\mathbf{p},\mathbf{x}}^{2n} d_{\mathbf{x}}^n$ (erreur de la reconstruction rationnelle de la dernière étape),
- $2d_{\mathbf{p},\mathbf{x}}^n(2(n+1)d_{\mathbf{p},\mathbf{x}}^n + nd_{\mathbf{p},\mathbf{x}})$ (contrôle des points non génériques),
- $d_{\mathbf{p},\mathbf{x}}^n(\lceil \log_2(d_{\mathbf{p},\mathbf{x}}^n) \rceil + 2)$ (contrôle des mauvais témoins).

FIG. 19.3 – *Comparaison aux valeurs témoins, cas triangulaire*

```
procedure Temoin(m_Temoin, J, t_n)

# Entree
# m_Temoin polynôme minimal de x_n en J
# J idéal de A
# t_n polynôme candidat, à coefficients dans fr(A)
# Sortie
# true si t_n = m_Temoin mod J
# false sinon
end;
```

L'entier M est un majorant de la borne de degré donnée du théorème donné dans l'introduction.

Théorème 25 *Les valeurs des paramètres* $(\mathbf{p}_1,\mathbf{p}_2,\gamma)$ *qui font échouer l'algorithme sont incluses dans une hypersurface de* $\mathbb{A}^{3m-1}(\overline{k})$ *de degré au plus* M.

Démonstration. L'algorithme réussit si $\mathbf{p}_1$ et $\mathbf{p}_2$ satisfont toutes les hypothéses de généricité, si γ est un changement de variables chanceux pour la dernière reconstruction rationnelle et si $\mathbf{p}_2$ est un bon témoin pour toutes les étapes intermédiaires. Ces conditions se traduisent par la non-annulation d'un polynôme non nul de degré inférieur à M. □

Finalement, voici la complexité totale de l'algorithme. Les constantes α et C ont été (re)-définies dans l'introduction de ce chapitre.

Théorème 26 *En cas de succès, la complexité en opérations de* k *de l'algorithme 19.1 est*

$$2\mathcal{C}_{Z\acute{e}ro}(\mathbf{f},k) + O_{log}\big((n^{\Omega}+nL)\mathcal{M}_s(4\mathcal{D}(x_n),m)(C^n\mathcal{D}_{\mathbf{x}}^{\alpha}+md_n)\big).$$

Démonstration. La complexité totale de la remontée est dominée par celle de la dernière étape ; on arrive donc à l'estimation :

$$2\mathcal{C}_{\text{Zéro}}(\mathbf{f}) + O_{\log}\big((n^{\Omega}+nL)\mathcal{M}_s(4\mathcal{D}(x_n),m)\sum_j(\mathcal{M}(d_1^{(j)},\dots,d_n^{(j)})+md_n^{(j)})\big).$$

D'après les estimations du chapitre 5, chaque terme $\mathcal{M}(d_1^{(j)},\dots,d_n^{(j)})$ est dans

$$O_{\log}\big(C^n(d_1^{(j)}\dots d_n^{(j)})^{\alpha}\big).$$

La somme des produits des $d_i^{(j)}$ est inférieure au degré $\mathcal{D}_{\mathbf{x}}$, et la somme des $d_n^{(j)}$ est égale au degré d_n du polynôme minimal de x_n, ce qui donne le résultat. □

Chapitre 20

Mise en pratique

Résumé

Nous montrons le comportement de l'implantation de l'algorithme du chapitre précédent sur une série d'exemples, pour la plupart tirés des parties d'applications qui suivent. Les conclusions sont les mêmes que celles des parties VI et VII, nous menons à bien des calculs que ne permettent pas d'effectuer les fonctions proposées par Magma version 2.6.

Sommaire

20.1 Implantation

La première phase des calculs est de trouver une représentation par ensembles triangulaires d'une fibre spécialisée. L'algorithme `Triangular` [138] permet d'obtenir les ensembles initiaux, par exemple à partir de la factorisation d'une résolution de Kronecker. Le logiciel Magma dispose également de fonctions de décomposition d'un idéal en composantes primaires.

L'algorithme principal de cette partie repose sur l'opérateur de Newton pour les ensembles triangulaires. Nous avons implanté cet algorithme en Magma, et comparé ses performances à celles de la version univariée issue de `kronecker` dans le chapitre 6. Cette version est légèrement plus lente, la différence provenant du coût de l'arithmétique sous-jacente, puisqu'un ensemble triangulaire est représenté comme un empilement de quotients univariés.

Dans ce chapitre, nous ne distinguons plus explicitement le cas des calculs sur les rationnels des calculs sur les corps finis. Nous avons mentionné dans le chapitre 19 une version de l'algorithme théorique utilisant des techniques de calcul modulaire, qui suit exactement celle de la partie VI et qui est utilisée en pratique.

20.2 Résultats des calculs

20.2.1 Comparaison

Dans cette partie, nous nous sommes imposés d'utiliser les ensembles triangulaires uniquement pour calculer le polynôme minimal de la dernière variable. En conséquence, nous avons décidé de comparer notre implantation à un calcul de base de Gröbner pour un ordre d'élimination, dans le logiciel Magma. À nouveau, nous distinguons deux possibilités de calcul :

- Le calcul sur le corps $k(p_1,\dots,p_m)$, pour un ordre qui sépare les blocs de variables $(x_1,\dots,x_{n_1})$ et (x_n) donne exactement le même résultat que nos algorithmes.
- Il peut être préférable de conserver les paramètres dans l'anneau $k[p_1,\dots,p_m,x_1,\dots,x_n]$: nous calculons pour un ordre qui sépare les blocs de variables $(x_1,\dots,x_{n_1})$ et $(x_n,p_1,\dots,p_m)$. On bénéficie d'une arithmétique certainement plus efficace ; en contrepartie, une telle base décrit complètement la projection sur l'espace des variables $(x_n,p_1,\dots,p_m)$: la sortie peut comporter davantage de polynômes, et être moins exploitable.

Notons que l'implantation FGb [1] de l'algorithme F4 par J.-C. Faugère [70] offre des performances de haut niveau : le système Hawes4, présenté ci-dessous, est résolu en 12 h.

D'autres implantations existent qui permettent d'effectuer ces calculs (implantations de P. Aubry, M. Moreno Maza [13] ou S. Dellière [57]). Il faut donc accorder une valeur indicative aux tables de temps, sachant qu'en outre, comme précédemment, on compare un algorithme probabiliste à un algorithme déterministe.

20.2.2 Systèmes utilisés

Nous avons retenu les systèmes suivants, qui sont donnés dans les annexes, et pour la plupart détaillés dans les chapitres d'application.

- $\mathbf{c_0}$: dans la section 25.3 du chapitre 25, nous résolvons une suite de 5 systèmes de 3 équations à 3 variables et 2 paramètres. Nous donnons ici le temps nécessaire à la résolution de l'un d'entre eux par notre méthode génerale ; le chapitre 25 montre une manière d'accélérer ces calculs. Ces systèmes sont donnés dans l'annexe E.6.

- **Bersenko** : ce système est présenté dans le chapitre 27, et donné dans l'annexe E.8.
- $\mathbf{j_1j_2j_3}$: ce système est présenté dans la section 25.2 du chapitre 25. Nous calculons ici uniquement le polynôme minimal de la dernière variable, ce qui n'est pas la réponse à la question posée dans le chapitre 25, mais nous sert à étalonner notre algorithme. Le système est donné dans l'annexe E.5.
- **EqMod1** : ce système calcule l'équation modulaire en genre 1, pour $\ell = 13$, d'après la section 24.3 du chapitre 24 ; il est donné dans l'annexe E.3.
- **EqMod3** : ce système donne l'équation modulaire en genre 2, pour $\ell = 3$, d'après la section 24.4 du chapitre 24. Il est donné dans l'annexe E.4.
- **Hawes4** : système-test, où l'on demande le polynôme minimal d'une variable non séparante. Ce système est donné dans l'annexe E.9.

À nouveau, ces systèmes s'évaluent bien : le nombre de multiplications est faible par rapport rapport au nombre de monômes de leurs formes développées.

20.2.3 Résultats

FIG. 20.1 – *Calcul de polynôme minimal par ensembles triangulaires*

Système	c_0	Bershenko	$j_1j_2j_3$	EqMod1	EqMod3	Hawes4
Variables n	3	3	4	3	3	6
Paramètres m	2	2	2	1	3	2
Degré Max. $d_{\mathbf{p},\mathbf{x}}$	24	12	10	89	32	9
Multiplications	77	32	41	130	50	55
Hauteur h	36	7	31	64	23	12
Degré en t d_n	1	12	5	14	40	120
Degré en $(p_1, \ldots, p_m)$ $\mathcal{D}(x_n)$	6	7	7	4	13	44
Hauteur $\mathcal{H}(x_n)$	19	27	90	25	49	170
Temps	4 min.	22 s.	21 s.	20 s.	4.25 h.	30 h.
Gröbner 1	> 5 h.	$>$ 5h.	> 5 h.	∞	∞	∞
Gröbner 2	> 5 h.	7 min.	> 5 h.	∞	∞	∞

- La ligne *Temps* donne le temps de calcul de notre implantation.

– La ligne *Gröbner 1* donne le temps le calcul d'une base de Gröbner dans l'anneau de polynômes $\mathbb{Q}(p_1,\ldots,p_m)[x_1,\ldots,x_n]$, et la ligne *Gröbner 2* indique le temps de calcul d'une base de Gröbner dans l'anneau $\mathbb{Q}[p_1,\ldots,p_m,x_1,\ldots,x_n]$, toutes deux pour un ordre d'élimination.
– Le signe ∞ indique l'arrêt des processus quand la consommation de mémoire a étouffé les calculs.

Conclusion Les conclusions à tirer de ces expériences sont semblables à celles des parties précédentes : notre modèle de calcul, bien que probabiliste, permet de résoudre des problèmes autrement inaccessibles aux fonctions de calculs de bases de Gröbner proposées par Magma ; ces bonnes performances découlent de la “non-explosion” des données intermédiaires. Enfin, l'influence de la complexité d'évaluation est manifeste sur tous ces exemples : si on dégrade les calculs d'évaluation à l'entrée, par exemple en développant les polynômes, le coût pratique s'en ressent d'autant.

Huitième partie

Systèmes surdéterminés à paramètres

Chapitre 21

Systèmes surdéterminés à paramètres

Résumé

Ce chapitre reprend l'essentiel de l'article [85], écrit avec M. Giusti. Nous considérons la question d'un système $(f_1,\ldots,f_{n+1})$ de polynômes de $\mathbb{Q}[p_1,\ldots,p_m,x_1,\ldots,x_n]$: les variables $\mathbf{p}$ sont toujours considérées comme des paramètres, et les variables $\mathbf{x}$ comme de véritables inconnues.

Les systèmes de ce type sont fréquemment rencontrés comme modélisations de phénomènes physiques, où un grand nombre de mesures sont possibles. Sous des hypothèses géométriques vraisemblablement rencontrées en pratique, nous montrons que l'ensemble des valeurs des paramètres pour lesquels le système est consistant forme une hypersurface, au-dessus de laquelle les valeurs des inconnues sont uniquement déterminées.

Dans ces conditions, pour une valeur générique des paramètres, le système $(f_1,\ldots,f_{n+1})$ est inconsistant. Nous proposons un algorithme qui, à partir d'un jeu de paramètres inconsistant, permet d'en déterminer des raffinements successifs qui approchent le lieu de consistance.

Sommaire

Introduction

Nous enrichissons les hypothèses de la partie VI, par la présence d'une équation supplémentaire f_{n+1}. Nous notons $\mathbf{f}$ le système partiel $(f_1,\ldots,f_n)$ et $\mathbf{F}$ le système complet $(f_1,\ldots,f_{n+1})$, tous ces polynômes étant à coefficients rationnels. L'étude qui suit reprend l'essentiel de l'article [85], écrit avec M. Giusti.

Tant que le nombre d'équations était égal au nombre d'inconnues, nous avions génériquement un nombre fini de solutions simples. Désormais, si l'équation f_{n+1} est suffisamment générique, le système $(f_1,\ldots,f_{n+1})$ devient inconsistant pour une valeur générique des paramètres ; et quand il est consistant, il n'admet génériquement qu'une solution. C'est le cas dans l'exemple suivant, qui est un calcul de pgcd dépendant de paramètres.

Exemple Dans $\mathbb{Q}[p_1,p_2,x]$, considérons les polynômes $f_1 = (x-p_1)(x-p_1-1)$ et $f_2 = (x-p_2)$. Le système $(f_1,f_2)(p_1,p_2,.)$ n'admet de solution que pour $p_1 = p_2$ ou $p_1+1 = p_2$; dans ce cas, cette solution est égale à p_2.

Cette situation est fréquemment rencontrée quand des questions de robotique [197, 64] ou de vision artificielle [128, 152] sont mises en équations sous forme polynomiale. Les équations dépendent de diverses mesures physiques, que l'on peut multiplier à loisir, et la solution physique est unique. Le bruit dans les mesures rend ces systèmes génériquement inconsistants, de sorte que ces questions sont difficiles à traiter par des méthodes formelles. La question de la résolution d'un système surdéterminé se pose ici dans ces termes :

On dispose d'une valeur initiale des paramètres $\mathbf{p}$ *pour laquelle le système* $\mathbf{F}(\mathbf{p},.)$ *n'est pas consistant ; on cherche un système consistant dans le voisinage de cette situation bruitée, et sa solution.*

Dans ce chapitre, nous allons mettre en forme quelques remarques à ce sujet, et proposer un algorithme de résolution. Nous nous baserons sur les hypothèses ci-dessous.

Hypothèse 6 *Nous supposons que :*

- *pour une valeur générique des paramètres* $\mathbf{p}$*, le système* $\mathbf{f}(\mathbf{p},.)$ *admet un nombre fini de solutions et ces solutions sont simples ; par contre le système* $\mathbf{F}(\mathbf{p},.)$ *n'admet pas de solution ;*
- *il existe une valeur* $\mathbf{p}_0$ *des paramètres telle que le système* $\mathbf{f}(\mathbf{p},.)$ *admet un nombre fini de solutions, ces solutions sont simples, et le système* $\mathbf{F}(\mathbf{p}_0,.)$ *est consistant.*

Dans la section 21.2, ces hypothèses vont permettre de définir une hypersurface de consistance, incluse dans l'espace des paramètres, au-dessus de laquelle le système $\mathbf{F}$ n'admet génériquement un nombre fini de solutions. Nous nous placerons dès lors sous une hypothèse d'unicité :

Hypothèse 7 *Nous supposerons que pour presque tout point* $\mathbf{p}$ *pour lequel le système* $\mathbf{F}(\mathbf{p},.)$ *est consistant, il n'admet qu'une solution.*

Dans la section 21.3, nous verrons que cette hypothèse permet de décrire les variables comme des fractions rationnelles en les paramètres, cette description étant valable au-dessus du lieu de consistance.

Notre méthode de résolution s'appuie sur cette description géométrique : à partir d'un jeu de données "bruitées", donc menant à un système inconsistant, nous allons approcher l'hypersurface de consistance, en utilisant un opérateur de Newton valable en dimension positive qui est décrit dans la section 21.4. Nous présentons dans la dernière section deux possibilités de mises en pratique, utilisant une méthode formelle basée sur les algorithmes des parties précédentes, ou numérique.

Ce chapitre se termine sur deux exemples. Le premier consiste en la détermination d'un point singulier sur une courbe plane ; il reprend les constructions de ce chapitre pas à pas. Le second est l'étude d'un robot manipulateur "surdéterminé", où notre processus de raffinement des données initiales permet d'obtenir des solutions numériques de bonne qualité à partir d'une situation bruitée.

21.1 Références

Les systèmes surdéterminés à coefficients imparfaitement connus ont récemment été l'objet de nombreux travaux. Une première instance de cette question est le calcul d'un *pgcd approché* de polynômes univariés, qui peut se formuler ainsi :

Soient f et g deux polynômes de $\mathbb{C}[x]$, et une tolérance $\varepsilon > 0$; on cherche des polynômes δf et δg de degrés inférieurs aux degrés de respectivement f et g, tels que la norme $||(\delta f,\delta g)||_2$ est inférieure à ε et pour lesquels le degré du pgcd du $f + \delta f$ et $g + \delta g$ est maximal.

Sous cette forme, le problème est abordé dans les travaux de Corless, Gianni, Trager et Watt [53], Emiris, Galligo et Lombardi [67, 68], Rupprecht [189] ; la situation est bien maîtrisée, via l'étude des valeurs propres et singulières de matrices de Sylvester associées au problème. Citons également l'approche de Schönhage [191], où l'on suppose qu'un oracle permet de raffiner les coefficients des polynômes d'entrée, et le travail de Pan [170] reposant sur une approximation numérique des racines des polynômes. Enfin, les calculs de pgcd paramétrés [6, 58] peuvent s'inscrire dans une veine analogue.

Dans le domaine des systèmes surdéterminés plus généraux, Bondyfalat, Mourrain et Pan [31] proposent un algorithme permettant de déterminer la racine d'un système de n équations à n inconnues qui minimise une équation supplémentaire. Dedieu et Shub [56] font dans ce cadre l'étude d'un opérateur de Gauss-Newton (qui détermine les solutions aux moindres carrés), dans le vocabulaire de l'α-théorie. Enfin, l'article de Huang, Stetter, Wu et Zhi [106] aborde la question de la factorisation approchée dans un contexte numérique, dans le même esprit que notre article [85].

21.2 Le lieu de consistance

Dans cette première section, nous montrons l'existence d'une hypersurface caractérisant localement les valeurs des paramètres pour lesquelles le système d'équations $\mathbf{F}(\mathbf{p},.)$ est consistant. D'après les résultats du chapitre 8, sous l'hypothèse 6, il existe une résolution des zéros de $\mathbf{f}$, valable au-dessus de presque toute valeur de $\mathbf{p}$, de la forme

$$q(t) = 0, \quad \left\{ \begin{array}{ccc} \frac{dq(t)}{dt} x_1 & = & v_1(t), \\ & \vdots & \\ \frac{dq(t)}{dt} x_n & = & v_n(t), \end{array} \right.$$

où t est une combinaison linéaire des variables $\mathbf{x}$, et q et les v_i sont des polynômes de $\mathbb{Q}(p_1, \ldots ,p_m)[t]$. Notons $\mathcal{F}$ une hypersurface de $\mathbb{C}^m$ contenant les valeurs des paramètres ne satisfaisant pas toutes les hypothèses de généricité pour t, puis $\mathcal{O}$ le complémentaire de $\mathcal{F}$ dans $\mathbb{C}^m$.

Nous allons décrire le lieu de $\mathcal{O}$ dans lequel le système $\mathbf{F}$ est consistant. Pour cela, nous appellerons π la projection

$$\begin{array}{cccc} \pi : & \mathbb{C}^{m+n} & \to & \mathbb{C}^m \\ & (\mathbf{p},\mathbf{x}) & \mapsto & \mathbf{p}. \end{array}$$

Théorème 27 *L'ensemble des paramètres* $\mathbf{p}$ *de* $\mathcal{O}$ *pour lesquels le système* $\mathbf{F}(\mathbf{p},.)$ *est consistant est de dimension* $m-1$. *Pour toute valeur* $\mathbf{p}$ *de* $\mathcal{O}$, *le système* $\mathbf{F}(\mathbf{p},.)$, *s'il est consistant, n'a qu'un nombre fini de solutions.*

Démonstration. La première hypothèse implique que $\overline{\pi(\mathcal{V})}$ est de dimension au plus $m-1$, montrons que sa dimension est au moins $m-1$.
Soit $\mathcal{C}$ une composante de $\mathcal{V}(f_1,\dots,f_n)$ telle qu'au-dessus de l'ouvert $\mathcal{O}$, les fibres de π restreinte à $\mathcal{C}$ sont de dimension zéro. Par hypothèse, il existe une telle composante pour laquelle $\mathcal{C}\cap\mathcal{V}(f_{n+1})$ est non vide. Dans ce cas, les fibres de π restreinte à $\mathcal{C}\cap\mathcal{V}(f_{n+1})$ ont donc aussi dimension zéro. D'un autre côté, d'après le théorème de Krull, toutes les composantes de $\mathcal{V}$ ont une dimension au moins égale à $m-1$. Le théorème de dimension des fibres [164, Theorem 3.13] implique que la dimension de $\mathcal{C}\cap\mathcal{V}(f_{n+1})$ et celle de son image par π sont au moins égales à $m-1$, ce qui donne la première partie du résultat.

Ensuite, pour toute valeur de $\mathbf{p}$ de $\mathcal{O}$, le système $\mathbf{f}(\mathbf{p},.)$ n'a qu'un nombre fini de solutions, il en est donc de même pour le système $\mathbf{F}(\mathbf{p},.)$ s'il est consistant. □

Définition 17 *Nous appellerons désormais* lieu de consistance *l'adhérence de l'ensemble des points de* $\mathcal{O}$ *qui sont dans l'image* $\pi(\mathcal{V})$. *C'est une hypersurface de* $\mathbb{C}^m$, *qui sera notée* Δ.

Dans l'ouvert $\mathcal{O}$, les points $\mathbf{p}$ de l'hypersurface Δ sont caractérisés par la consistance du système

$$\{f_1(\mathbf{p},.) \;=\; \cdots = f_n(\mathbf{p},.) = f_{n+1}(\mathbf{p},.) = 0\}. \tag{21.1}$$

Les solutions des n premières équations y sont décrites par la résolution donnée précédemment. Mettons cette résolution sous forme “Shape Lemma”, et supprimons les dénominateurs, pour l'écrire sous la forme :

$$\tilde{q}(t,p_1,\dots,p_m)=0,\quad \left\{\begin{array}{rcl} \rho x_1 &=& \widetilde{w_1}(t,p_1,\dots,p_m)\\ &\vdots& \\ \rho x_n &=& \widetilde{w_n}(t,p_1,\dots,p_m),\end{array}\right.$$

où t est une combinaison linéaire des variables x_i, ρ un polynôme de $\mathbb{Q}[p_1,\dots,p_m]$ qui ne s'annule pas dans $\mathcal{O}$, et $\tilde{q}$ et les $\widetilde{\mathbf{w}}$ sont des polynômes de $\mathbb{Q}[p_1,\dots,p_m][t]$; le polynôme $\tilde{q}$ n'est pas nécessairement unitaire. Il existe un entier d tel que l'équation $\rho^d f_{n+1}$ se réécrit par calculs de pseudo-restes sous la forme

$$r_{n+1}(t,p_1,\dots,p_m)\in\mathbb{Q}[p_1,\dots,p_m][t].$$

Alors le système 21.1 est consistant en une valeur $\mathbf{p}$ des paramètres de $\mathcal{O}$ si et seulement si les équations $\tilde{q}(t,p_1,\dots,p_m)$ et $r_{n+1}(t,p_1,\dots,p_m)$ spécialisées en $\mathbf{p}$ ont une racine commune. Leur résultant en t fournit donc une équation de Δ valable dans l'ouvert $\mathcal{O}$.

Lien avec les pgcd approchés. Nous nous sommes ramenés à la question du calcul du pcgd de deux polynômes univariés dépendant de paramètres. Les coefficients de ces polynômes ne sont pas algébriquement indépendants, les résultats de [67, 67, 189] ne s'appliquent donc pas directement.

21.3 Unicité de la solution

Nous étudions maintenant les conséquences de l'hypothèse 7 d'unicité de la solution. Pour commencer, notons que certaines situations assurent que cette hypothèse est vérifiée localement :

Proposition 70 *Supposons qu'il existe une valeur des paramètres* $\mathbf{p}_0$ *telle que :*
- *le système* $\mathbf{f}(\mathbf{p}_0,.)$ *a un nombre fini de solutions, qui sont simples et en nombre maximal ;*

– *le système* $\mathbf{F}(\mathbf{p}_0,.)$ *est consistant et admet une seule solution.*

Alors dans toute composante de $\pi(\mathcal{V})$ *contenant* $\mathbf{p}_0$*, pour presque tout* $\mathbf{p}$*, le système* $\mathbf{F}(\mathbf{p},.)$ *n'admet qu'une solution.*

Démonstration. Toutes les racines de $\mathbf{f}(\mathbf{p}_0,.)$ sont simples, donc d'après le théorème d'inversion locale, il existe une boule $B_0 \subset \mathbb{C}^m$ centrée en $\mathbf{p}_0$ et des boules $B_i \subset \mathbb{C}^{m+n}$ disjointes, centrées en chacune de ces racines, telles qu'au-dessus de chaque valeur de $\mathbf{p}$ de B_0 il existe un unique zéro de $\mathbf{f}(\mathbf{p},.)$ dans B_i ; ce sont en outre les seuls zéros de $\mathbf{f}(\mathbf{p},.)$ puisque nous avons supposé que les racines de $\mathbf{f}(\mathbf{p}_0,.)$ sont en nombre maximal.
Soit $\mathcal{C}$ une composante de $\pi(\mathcal{V})$ passant par $\mathbf{p}_0$ et $\mathcal{C}'$ une composante de sa préimage par π restreinte à $\mathcal{V}$. Au-dessus du point $\mathbf{p}_0$, cette dernière ne rencontre les zéros de $\mathbf{f}$ que dans l'une des boules B_i mentionnées ci-dessus. Pour tout i, la fermeture de la projection $\pi(B_i \cap \mathcal{C}')$ est un fermé de $\mathbb{C}^m$; par la remarque précédente, il existe un unique i pour lequel ce fermé contient $\mathbf{p}_0$.
On en déduit que dans un voisinage (pour la topologie classique) de $\mathbf{p}_0$ dans $\mathcal{C}$, le système $\mathbf{F}(\mathbf{p},.)$ n'admet qu'une seule solution dans $\mathcal{C}'$. Ceci prouve que pour presque tout $\mathbf{p}$ de $\mathcal{C}$ le système $\mathbf{F}(\mathbf{p},.)$ n'admet qu'une solution dans $\mathcal{C}'$. Faisant de même pour toutes les composantes de $\pi^{-1}(\mathcal{C})$ au-dessus de $\mathbf{p}$, on obtient le résultat demandé. □

Il peut exister des valeurs des paramètres $\mathbf{p}$ pour lesquelles le nombre de solutions de $\mathbf{F}(\mathbf{p},.)$ est supérieur à 1. Dans le cas où Δ est lisse, on peut assurer que le cardinal maximal d'une fibre est 1, d'après [98, Proposition 1], mais cette hypothèse de lissité de Δ est peu vraisemblable : Δ est un objet éliminant, donc certainement singulier. De manière générale, le cardinal d'une fibre finie de π ne peut être supérieur à 1 qu'au-dessus d'un point singulier de Δ, d'après le Main Theorem de Zariski [164, Theorem 3.20].

Exemple Considérons à nouveau une situation de pgcd : on pose $f_1 = (x - p_1)(x - p_2 + p_1)$ et $f_2 = (x - p_2)(x + p_1 - 1)$. Ce système est consistant si $p_1(p_1 - p_2)(p_1 - p_2 + 2)(p_1 - 1/2) = 0$, et n'a alors génériquement qu'une solution. Le point (2,2) est singulier sur cette courbe et le système $(f_1,f_2)(1,1,.)$ a 2 solutions distinctes.

Il est possible d'exprimer la valeur de l'unique solution $\mathbf{x}$ du système $\mathbf{f}(\mathbf{p},.)$ par des fractions rationnelles en $(p_1,\ldots,p_m)$:

Proposition 71 *Il existe un jeu de variables* $(p'_1,\ldots,p'_m)$ *de* $\mathbb{C}^m$ *et une description de l'hypersurface* Δ *et de sa préimage par* π *de la forme :*

$$Q = 0, \quad \left\{ \begin{array}{rcl} Q'x_1 & = & V_1, \\ & \vdots & \\ Q'x_n & = & V_n, \end{array} \right.$$

où Q *et les* V_i *sont des polynômes de* $\mathbb{Q}(p'_1,\ldots,p'_{m-1})[p'_m]$.

Démonstration. Le lieu de consistance Δ est une hypersurface de l'espace des paramètres $\mathbb{C}^m$. Soient $\mathcal{H}$ un hyperplan de $\mathbb{C}^m$ tel que la projection de Δ sur $\mathcal{H}$ est surjective, et $(p'_1,\ldots,p'_m)$ un nouveau jeu de coordonnées, dans lesquelles $\mathcal{H}$ est défini par $p'_m = 0$. Alors la projection de $\mathcal{V}$ sur l'hyperplan $\mathcal{H}$ est d'image dense, et génériquement finie. D'après notre hypothèse d'unicité, la variable p'_m est élément séparant de la fibre générique au-dessus de $\mathcal{H}$. Nous pouvons donc appliquer les résultats du chapitre 8. □

La paramétrisation n'est unique que modulo une équation définissant Δ. Reprenons la construction d'une équation caractérisant localement Δ faite dans la section précédente. Tout comme le résultant des polynômes $q(t)$ et $r_{n+1}(t)$ fournit une équation Q de Δ valable dans $\mathcal{O}$, le dernier sous-résultant de ces polynômes donne t en fonction des paramètres $(p_1,\ldots,p_m)$. Cette construction permet par la suite d'obtenir des paramétrisations des variables x_i au-dessus de Δ, valables dans un ouvert de $\mathcal{O}$.

Lien avec l'algorithme de résolution géométrique. La résolution écrite ci-dessus est obtenue par un procédé de *relaxation* : nous faisons passer un paramètre du côté des variables, il devient alors lié sur l'ensemble des paramètres restés libres. Cela nous ramène à une situation avec autant d'équations que d'inconnues.

Le chemin suivi est celui d'une étape de l'algorithme de résolution géométrique. Partant de la représentation des zéros du système $(f_1,\ldots,f_n)$ pour un élément séparant t, nous obtenons une représentation des zéros du système $(f_1,\ldots,f_{n+1})$, par le calcul d'un résultant avec la nouvelle équation f_{n+1}.

21.4 Vers un algorithme de résolution

Dans cette section, nous proposons une réponse à la question suivante : on disposons d'une valeur initiale des paramètres $\mathbf{p}_0$ en dehors du lieu de consistance, et on cherche un système consistant dans son voisinage, pour en donner la solution.

21.4.1 Description

Le principe de l'algorithme est de calculer des raffinements successifs du système initial, qui le font "tendre" vers un système consistant. Je ne donnerai pas de résultat de complexité concernant cet algorithme, considérant que son étude reste en cours.

L'algorithme prend en entrée :

- le système $\mathbf{F}$;
- un jeu de paramètres $\mathbf{p}_0$ pour lequel le système $\mathbf{F}(\mathbf{p}_0,.)$ est inconsistant ;
- un critère d'arrêt `Arret` pour arrêter l'algorithme, quand on estime posséder une solution satisfaisante.

À partir du point $\mathbf{p}_0$, on construit la suite $\mathbf{p}_i$ qui va tendre vers sur l'hypersurface Δ. Le calcul repose sur l'itération de Newton pour les systèmes de dimension positive détaillée ci-dessous, que nous notons `NewtonGeneralise`. Nous notons `SolutionApprochee`$(\mathbf{p})$ une fonction donnant une solution approchée $\mathbf{x}_{\text{app}}$ du système $\mathbf{F}(\mathbf{p},.)$.

Remarque Plus généralement, si on ne fait pas l'hypothèse d'unicité de la solution, on peut obtenir une description des variables x_i au-dessus des composantes de dimension $m-1$ de $\pi(\mathcal{V})$ par un ou plusieurs ensembles triangulaires à coefficients dans $\mathbb{Q}(p_1,\ldots,p_m)$. Le principe d'approximation de l'hypersurface de consistance s'applique toujours, mais on obtient plusieurs solutions au-dessus d'un point de Δ.

21.4.2 L'opérateur de Newton en dimension positive

Dans les parties précédentes, nous avons utilisé un opérateur de Newton classique, mais dans un cadre non archimédien. Ici, nous utilisons au contraire un opérateur de Newton archimédien, mais appliqué à la recherche d'un point sur l'hypersurface Δ. Pour cela, nous supposons pouvoir évaluer une équation Q donnant Δ, ainsi que ses dérivées ; nous supposons également que le gradient de Q, noté $\mathbf{Jac}(Q)$, est non nul en les points $\mathbf{p}_i$ successifs.

FIG. 21.1 – *Systèmes polynomiaux surdéterminés*

procedure `Resolution`($\mathbf{F}$,$\mathbf{p}_0$,`Arret`)

\# **Entree**
\# le système $\mathbf{F}$
\# $\mathbf{p}_0$: valeur initiale des paramètres
\# `Arret`(.,.) : fonction à valeur booléene
\# **Sortie**
\# $\mathbf{p}$,$\mathbf{x}_{\text{app}}$: une valeur des paramètres et des inconnues satisfaisant `Arret`

$\mathbf{p} \leftarrow \mathbf{p}_0$;
$\mathbf{x}_{\text{app}} \leftarrow$ `SolutionApprochee`($\mathbf{p}$);
while not `Arret`($\mathbf{p}$,$\mathbf{x}_{\text{app}}$) **do**
  $\mathbf{p} \leftarrow$ `NewtonGeneralise`($\mathbf{p}$);
  $\mathbf{x}_{\text{app}} \leftarrow$ `SolutionApprochee`($\mathbf{p}$);
od ;
return $\mathbf{p}$,$\mathbf{x}_{\text{app}}$;
end;

On peut étendre la méthode de Newton aux systèmes dont la jacobienne est de rang constant, en utilisant son *inverse de Moore-Penrose.* Nous reprenons ici la définition donnée dans [55], pour le cas particulier de la matrice d'une application linéaire surjective.

Définition 18 (Inverse de Moore-Penrose) *Soit* $\mathbf{A}$ *la matrice d'une application linéaire surjective ; on appelle inverse de Moore-Penrose de* $\mathbf{A}$ *et on note* $\mathbf{A}^\dagger$ *la matrice* $\mathbf{A}^*(\mathbf{A}\mathbf{A}^*)^{-1}$.

Cette définition permet d'étendre l'opérateur de Newton à la résolution de systèmes de dimension positive : l'opérateur de Newton généralisé est défini par

$$N_Q \ : \mathbf{p} \mapsto \mathbf{p} - \mathbf{Jac}(Q)(\mathbf{p})^\dagger Q(\mathbf{p}).$$

Nous appellerons *zéro approché* de Q un point $\mathbf{p}$ dont les itérés par N_Q convergent quadratiquement vers un point de Δ.

Cet opérateur est étudié par Allgower et Georg [8], Beyn [27], Tran [201] ; Dedieu et Kim [55] offrent une présentation synthétique des résultats d'α-théorie dans ce cadre, à la suite des travaux initiés par Kim, Shub, Smale,... [118, 196, 56].

Remarque Il est possible d'utiliser une autre procédure produisant une suite de points $\mathbf{p}_n$ qui tend vers le lieu de consistance. Par exemple, on pourra chercher un point $\mathbf{p}_\infty$ de Δ qui rende extrémale la distance à $\mathbf{p}_0$, c'est-à-dire un point critique de cette fonction sur Δ. Ces points annulent le système

$$Q(\mathbf{p}) = 0, \quad \mathbf{grad}\, Q(\mathbf{p}) // \mathbf{p}\mathbf{p_0}.$$

21.4.3 Critère d'arrêt du calcul

La fonction `Arret` décide de l'arrêt du processus d'approximation. Dans le cadre de ce mémoire, nous nous sommes limités à des mises en pratique expérimentales, en décidant de l'arrêt quand la valeur $f_{n+1}(\mathbf{p},\mathbf{x}_{\mathrm{app}})$ devient inférieur à une valeur ε fixée a priori.

L'α-théorie [196] donne un critère numérique local, estimé au point $\mathbf{p}_i$ à partir de toutes les dérivées de Q, dont la vérification implique que $\mathbf{p}_i$ est un zéro approché de Q. À terme, il sera certainement préférable de choisir un tel test, pour arrêter les calculs quand $\mathbf{p}_i$ devient un zéro approché de Q.

21.5 Mise en place

Pour mettre en pratique l'algorithme ci-dessus, il faut pouvoir évaluer une équation caractérisant Δ, et donner une valeur approchée aux inconnues $\mathbf{x}$. Nous donnons dans cette section deux réponses à ces questions : dans un premier temps, nous utilisons des méthodes formelles reposant sur les parties précédentes de ce mémoire ; nous proposons ensuite une version purement numérique.

21.5.1 Méthode formelle

La première solution est d'écrire complètement une résolution paramétrée du système $\mathbf{F}$, par exemple celle de la proposition 71. Le coût du calcul est détaillé dans les chapitres 10 et 15. Une fois connue la résolution, il suffit d'injecter les coordonnées des points $\mathbf{p}_i$ successifs dans l'équation de Q et de ses dérivées pour suivre l'itération de Newton généralisée. Les paramétrisations fournissent des estimations des valeurs des inconnues $\mathbf{x}$.

Cette méthode rejoint celle de l'article [85], où nous proposons un algorithme où les données sont codées par des *calculs d'évaluation*. Cet article suppose que le lieu les zéros $\mathcal{V}(f_1,\ldots,f_{n+1})$ est en position de Noether par rapport à l'espace des paramètres, ce qui permet de représenter cette résolution par un programme sans test ni division ; dans le cas général, ce n'est pas le cas.

21.5.2 Une version purement numérique

Il est possible d'éviter le calcul d'une résolution générique. Au point courant $\mathbf{p}_i$, pour passer à l'étape $i+1$, il suffit de connaître l'*évaluation* d'une équation donnant Δ et de ses dérivées.

Supposons donnés les zéros simples du système des n premières équations $\mathbf{f}(\mathbf{p}_i,.)$ par des approximations numériques. L'algorithme de Newton classique fournit les développements en séries de ces solutions par rapport aux variables $(p_1,\ldots,p_m)$ à un ordre arbitraire. Localement, Δ est définie par l'annulation d'un résultant, calculé à partir de la dernière équation. On peut calculer ce résultant et ses dérivées à un ordre arbitraire en $(p_1,\ldots,p_m)$, comme le produit

$$\prod_{\mathbf{x}_i^{(j)}} f_{n+1}(\mathbf{p}_i,\mathbf{x}_i^{(j)}),$$

où les $\mathbf{x}_i^{(j)}$ sont les développements en séries des zéros de $\mathbf{f}(\mathbf{p}_i,.)$, à l'ordre recherché. Pour estimer les valeurs approchées des inconnues $\mathbf{x}$, nous retenons des racines $\mathbf{x}_i^{(j)}$ celle qui minimise la dernière équation f_{n+1}.

Il suffit de calculer une approximation des solutions des n premières équations au-dessus du point $\mathbf{p}_{i+1}$ pour poursuivre le processus : on utilise un procédé d'homotopie [29, 45] pour obtenir les zéros de $\mathbf{f}(\mathbf{p}_{i+1},.)$ à partir des zéros du système $\mathbf{f}(\mathbf{p}_i,.)$.

21.5.3 Conclusion, travaux en cours

Nous avons proposé une étude géométrique d'une situation surdéterminée, pour en déduire le schéma d'un algorithme de résolution. Le problème est découplé en la recherche d'un point sur le lieu de consistance d'un côté, et l'estimation d'une solution approchée de l'autre.

L'étude n'est pas close, de nombreuses questions sont en suspens :

- L'approche formelle fournit une paramétrisation des variables valable au-dessus du lieu de consistance. Le choix de cette paramétrisation n'est pas unique, et en-dehors du lieu de consistance, les diverses solutions n'ont pas le même comportement. Y a-t-il des choix meilleurs que d'autres ?
- La solution numérique remplace l'utilisation d'une paramétrisation par des méthodes d'homotopie. L'étude théorique de cette méthode est à faire, notamment pour estimer le nombre de chiffres significatifs nécessaires.

Dans ce contexte, citons l'article de Castro, Hägele, Morais et Pardo [45], qui est un premier pas visant à rapprocher les méthodes formelles ("Kronecker solving") de méthodes numériques ("Newton solving"), dans le cadre des systèmes de dimension zéro.

21.6 Exemple : point singulier d'une courbe plane

Considérons le polynôme de $\mathbb{Q}[a,b,c,d][x,y]$

$$f = 1 + ax + bxy + cy^2 + dx^4;$$

nous cherchons à quelles conditions la courbe $\mathcal{V}(f)$ est singulière. On identifie ces courbes aux points de $\mathbb{C}^4$, de coordonnées (a,b,c,d), les points singuliers de $\mathcal{V}(f)$ sont les solutions du système :

$$\begin{array}{lrcl} f: & 1 + ax + bxy + cy^2 + dx^4 & = & 0, \\ f_x: & 4dx^3 + by + a & = & 0, \\ f_y: & bx + 2cy & = & 0, \end{array}$$

où les "inconnues"sont les variables (x,y) et les paramètres les variables (a,b,c,d). Ce premier exemple va illustrer les constructions précédentes. Nous suivons le plan de ce chapitre, en détaillant les calculs pour chaque étape.

Le lieu de consistance. Dans un premier temps, isolons deux équations de ce système, admettant un nombre fini de solutions pour un choix générique des paramètres : on choisit (f_x,f_y). Une représentation des zéros du système (f_x,f_y), sous forme "Shape Lemma", avec pour éléments séparant $t = x + 2y$, est :

$$\left\{ \begin{array}{rcl} (b-c)x & = & -ct, \\ 2(b-c)y & = & bt, \end{array} \right.$$

$$q: \quad 8c^3dt^3 + (-b^4 + 2b^3c - b^2c^2)t + 2(ab^3 + 3ac^2c - 3abc^2 + ac^3) = 0.$$

Nous travaillons dans le complémentaire de l'hypersurface définie par le résultant de q avec sa dérivée :

$$\mathcal{O} = \left\{(a,b,c,d) \in \mathbb{C}^m, cd(c-b)(216a^2dc^3 - b^6) \neq 0\right\}.$$

Ajoutons la dernière équation, le polynôme f lui-même, dans lequel on injecte les paramétrisations précédentes. Le lieu de consistance est l'ensemble des points (a,b,c,d) de $\mathcal{O}$ pour lesquels il existe t tel que

$$\begin{array}{rrcl} q(t,d): & 8c^3dt^3 + (-b^4 + 2b^3c - b^2c^2)t + 2(ab^3 + 3ab^2c - 3abc^2 + ac^3) & = & 0, \\ r_3(t,d): & 2c^4dt^4 - b^2c(b-c)^2t^2 - 2ac(b-c)^3t + 2(b-c)^4 & = & 0 \end{array}$$

Une fois supprimés les parasites (le discriminant et le terme de tête de q), le résultant en t de ces deux polynômes donne l'équation de Δ recherchée :

$$\Delta = \{(a,b,c,d) \in \mathcal{O}, Q(a,b,c,d) = -432c^4a^4d + 4096c^4d^2 - 576b^2da^2c^3 - 128b^4dc^2 + b^6a^2c + b^8 = 0\}.$$

Si on garde q unitaire, le résultant est multiplié par une puissance du dénominateur de q, qui est non nul dans $\mathcal{O}$.

Calcul des paramétrisations. Au-dessus des zéros de Q, les coordonnées des points singuliers sont données par les paramétrisations ci-dessous. Nous choisissons le dernier sous-résultant de $q(t)$ et $r_3(t)$, ce qui donne t, puis successivement :

$$\begin{array}{rcl} t & = & -2\dfrac{a(c-b)(192dc^2+b^4)}{288a^2dc^3+64b^2dc^2-b^6}, \\ x & = & -2\dfrac{ac(192dc^2+b^4)}{288a^2dc^3+64b^2dc^2-b^6}, \\ y & = & \dfrac{ba(192dc^2+b^4)}{288a^2dc^3+64b^2dc^2-b^6}. \end{array}$$

Les points de Δ de l'ouvert $\mathcal{O}$ en lesquels cette paramétrisation est valable sont des courbes singulières à un seul point singulier. En regardant dans les fermés de $\mathbb{C}^4$ que l'on a rejetés, on peut trouver des composantes de dimensions plus petites, décrivant des courbes à singularités de moins en moins génériques.

Exemple numérique. Considérons la courbe

$$f_0 : 1 - 3x - 3.61xy + 2.67y^2 + 5x^4.$$

Cette courbe n'est pas singulière, et on cherche une courbe singulière de type "générique" dans son voisinage, ainsi que les coordonnées du point singulier.

La tableau ci-dessous donne les résultats de l'application de l'algorithme 21.4.1 à cette situation. Le test d'arrêt demandait que l'estimation des coordonnées du point P singulier annulent les trois équations à 10^{-2} près. Les lignes donnent respectivement la courbe initiale f_0, la courbe limite f_∞ et les coordonnées de son point singulier P_{sing}, obtenues par les paramétrisations ci-dessus.

	`NewtonGeneralise`
f_0	$1 - 3x - 3.61xy + 2.67y^2 + 5x^4$
f_∞	$1 - 1.94x - 3.27xy + 2.07y^2 + 4.98x^4$
P_{sing}	[.55,.44]

Remarque Dans ce cadre, on peut tout aussi bien considérer les courbes f comme des points de $\mathbb{P}^4(\mathbb{C})$, et utiliser un algorithme de Newton homogène.

Choix de la paramétrisation. Une question non éclaircie est le choix d'une "bonne" paramétrisation pour estimer une valeur approchée des solutions. Nous avons donné une possibilité ci-dessus, issue d'un calcul de résultant par l'algorithme d'Euclide. D'autres paramétrisations sont possibles, par exemple celle proposée dans la proposition 71. Ces deux possibilités donnent deux jeux de paramétrisations, qui sont égales modulo l'équation Q :

$$\begin{aligned} x_1 &= -2\frac{ac(192dc^2+b^4)}{288a^2dc^3+64b^2dc^2-b^6}, \\ y_1 &= \frac{ba(192dc^2+b^4)}{288a^2dc^3+64b^2dc^2-b^6} \end{aligned}$$

$$\begin{aligned} \text{et} \quad x_2 &= 2\frac{-256c^2d+27a^4c^2+30a^2b^2c+4b^4}{ab^2(9a^2c+8b^2)}, \\ y_2 &= -\frac{-256c^2d+27a^4c^2+30a^2b^2c+4b^4}{abc(9a^2c+8b^2)}. \end{aligned}$$

Estimons le gradient de ces paramétrisations au-dessus d'un point du lieu de consistance : plus il est petit, moins l'incertitude sur les paramètres a d'influence sur l'estimation des solutions. Le tableau suivant donne le gradient de (x_1,y_1), (x_2,y_2) et (x_3,y_3) par rapport à (a,b,c,d), évalué au point $(1,2,3,0.34460589128545012430071957856\,6)$, où Q vaut approximativement 10^{-25}.

FIG. 21.2 – *Gradient de différentes paramétrisations*

	$/\partial a$	$/\partial b$	$/\partial c$	$/\partial d$
$\partial(x_1,y_1)$	$(0.6,-0.2)$	$(0.1,0.1)$	$(-0.04,-0.1)$	$(0.1,-0.04)$
$\partial(x_2,y_2)$	$(16,-5.4)$	$(5.8,-1.7)$	$(-2,0.5)$	$(19,6.5)$

Le premier choix apparaît bien plus stable. J'ignore si cette propriété est propre à cet exemple, ou s'il s'agit d'un phénomène plus général.

21.7 Le robot parallèle

En 1965, D. Stewart [197] propose la construction d'un robot manipulateur parallèle, en vue de construire des simulateurs de vol. Le robot parallèle se distingue du robot série par la présence de plusieurs bras articulés qui relient la base au même objet. Notre robot parallèle consistera en :

- deux corps solides (les plateformes) A et B ; nous travaillerons dans un repère lié à la plateforme A.
- deux jeux de points $(A_1,\dots,A_6)$ (resp. $(B_1,\dots,B_6)$), liés à la plateformes A (resp. B) ; les points (A_i,B_i) sont reliés par des bras rectilignes, dont la longueur peut varier.

Nous nous intéressons au problème *direct* : étant données la géométrie des deux plateformes A et B, et les longueurs des bras (A_iB_i), il s'agit de déterminer la position de la plateforme B par rapport à la plateforme A. Cette question peut se réécrire comme un système d'équations polynomiales, dont les inconnues sont le déplacement menant du repère lié à A au repère lié à B. Dans le cas général, six mesures sont nécessaires pour se ramener à un système de dimension zéro : le déplacement est formé d'une translation et une rotation de l'espace, soit 6 inconnues à déterminer ; il faut donc spécifier 6 paramètres pour pouvoir se ramener en dimension zéro.

Ce système a fait l'objet d'un grand nombre d'études : pour diverses formes de la plateforme, de nombreux travaux visent à déterminer le nombre maximal de solutions admissibles, réelles ou complexes [139, 183, 162] . J.P. Merlet [157] fournit une description technique et historique détaillée de ce sujet. La question d'une plateforme de Stewart "surdéterminée", dépendant d'une mesure supplémentaire, a été abordée dans [96, 199].

Dans [85], nous étudions l'exemple d'une plateforme de Stewart selon le modèle donné par S. Egner dans [64], à laquelle nous adjoignons une septième patte. Les calculs sont rendus délicats par la présence de nombres algébriques dans les paramètres de construction de la plateforme, aussi nous prenons ici un autre exemple, dont les équations sont données dans l'annexe E.1.

La situation est modélisée par un système de 7+1 équations à 7 inconnues, qui sont les coordonnées d'un déplacement paramétré par un quaternion et un vecteur de translation, et 7 paramètres. Les 6 premières mesures sont les longueurs des bras, la 7ième est l'abscisse x_1 du point B_1. Cette surdétermination préserve la symétrie des solutions par rapport au plan $z = 0$, l'hypothèse d'unicité 7 n'est pas satisfsfaite. Le principe d'approximation de l'hypersurface de consistance s'applique toujours, mais on obtient deux solutions au-dessus d'un point de Δ, symétriques par rapport au plan $z = 0$.

Nous partons de la situation idéale suivante : les coordonnées des points B_i sont

Point	x	y	z
B_1	12.31	4.11	14.7
B_2	16.33	-2.67	17.1
B_3	5.60	6.72	15
B_4	8.00	-7.79	21.6
B_5	-2.74	1.6	19.5
B_6	-0.6	-5.5	22.5

pour lesquelles longueurs des bras sont

$$L_0 = [15.9982, 19.0701, 16.0606, 23.8027, 20.9447, 24.4307]$$

Perturbons les longueurs : nous partons de l'estimation

$$L_0 = [15.998, 19.07, 16.06, 23.802, 20.944, 24.43],\ x_1 = 12.31.$$

On détermine l'ensemble des solutions des 7 premières équations, par un processus quelconque de résolution en dimension zéro. Parmi les solutions des 7 premières équations, celle qui s'approche le plus de la solution non perturbée est :

Point	x	y	z
B_1	12.17	4.21	14.75
B_2	16.2	−2.57	17.15
B_3	5.46	6.81	15.06
B_4	7.87	−7.70	21.65
B_5	−2.88	1.69	19.56
B_6	−0.73	−5.41	22.56

Pour raffiner ces solutions, nous utilisons la version numérique du processus d'approximation, vue dans la section 21.5, avec une précision fixée empiriquement à 40 chiffres significatifs. On choisit pour solution approchée $\mathbf{x}_{\mathrm{app}}$ la solution des 7 premières équations qui minimise la dernière équation ; le processus de raffinement des paramètres est arrêté lorsque la valeur de la dernière équation y devient inférieure à 10^{-30}. Au terme du processus de raffinement des paramètres, on obtient la solution suivante, tronquée à deux chiffres après la virgule ; on peut vérifier sa proximité à la solution idéale.

Point	x	y	z
B_1	12.3	4.14	−14.7
B_2	16.33	−2.65	−17.1
B_3	5.58	6.74	−15.0
B_4	7.99	−7.77	−21.6
B_5	−2.75	1.62	−19.51
B_6	−0.61	−5.48	−22.5

Le processus de correction d'une estimation initiale bruitée donne des résultats encourageants, et l'utilisation de méthodes numériques n'a pas posé de difficultés. L'étape suivante sera de mettre en œuvre cet algorithme avec des données réelles.

Neuvième partie

Applications : calculs dans les courbes

Chapitre 22

Vue d'ensemble

Résumé

Cette partie décrit des questions liées aux courbes hyperelliptiques, en genre 2 principalement. Les sujets abordés ici sont le fruit d'une collaboration avec P. Gaudry, du Laboratoire d'Informatique de l'École polytechnique.

Les questions de nature notamment arithmétique qui se posent en théorie des courbes elliptiques et hyperelliptiques dépassent le cadre de la géométrie effective. Malgré tout, certaines d'entre elles se traduisent dans ce vocabulaire, et nous avons pu y apporter des réponses. Ces questions ont également été l'occasion de mettre au point une grande partie des algorithmes présentés dans ce mémoire.

Ce chapitre introductif présente brièvement les points sur lesquels nous avons travaillé et donne un ensemble de définitions suffisantes à la compréhension des chapitres qui suivent.

Sommaire

22.1 Résultats des chapitres suivants

Nous nous intéressons dans cette partie à trois questions issues de l'étude des courbes elliptiques et hyperelliptiques. La première d'entre elles a pour but de faire aboutir le calcul du cardinal de la Jacobienne d'une courbe de genre 2 donnée . Les deux points suivants abordent des questions plus générales, en traitant des courbes vues dans leur ensemble ; nous avons obtenu ou retrouvé des résultats généraux, et cette étude n'est pas achevée. Vues sous l'angle des systèmes polynomiaux, toutes ces questions rentrent dans le cadre des parties précédentes : nous les avons interpretées comme des systèmes à paramètres.

Les définitions nécessaires à la compréhension des lignes ci-dessous ont été reportées à la section suivante.

Division par deux dans une Jacobienne. Le chapitre 23 présente une application de l'algorithme de calcul d'une résolution de Kronecker générique à une question de *division par deux* dans la Jacobienne d'une courbe définie sur un corps fini de cardinal de l'ordre de 10^{19}. La division se fait par la résolution d'un système polynomial, paramétré par les "coordonnées" du dividende. Ce calcul a été effectué en réponse à une question de P. Gaudry et R. Harley (Projet Cristal, INRIA), et a contribué à l'aboutissement du calcul du cardinal de cette Jacobienne (voir leur article [74]).

Construction d'équations modulaires. Le chapitre 24 propose une définition d'*équations modulaires*, sous un angle géométrique utilisant le vocabulaire de ce mémoire. Pour les courbes de genre 1, nous retrouvons une construction due à Charlap, Coley et Robbins [46] ; en genre supérieur, la construction semble nouvelle. Dans tous les cas, nous les obtenons comme des polynômes minimaux d'éléments bien choisis ; ces polynômes dépendent des paramètres nécessaires pour décrire les courbes d'un genre donné.

La spécialisation de ces polynômes permet d'obtenir rapidement des informations sur le cardinal d'une courbe : un débouché possible de cette construction serait d'inclure ces polynômes dans des bases de données. Nous donnons à la fin du chapitre 24 le premier exemple pour les courbes de genre 2, avec une application au calcul de la cardinalité d'une Jacobienne modulo 3.

Les courbes à Jacobienne décomposable. Le chapitre 25 est une étude des courbes de genre 2 à Jacobienne (2,2)-décomposable ; ces courbes sont l'objet de nombreux travaux [108, 135, 129, 146, 105, 130]. Nous calculons ici le polynôme minimal des j-invariants des courbes elliptiques quotients d'une courbe à Jacobienne décomposable : c'est un polynôme de degré 2, qui dépend des paramètres décrivant l'ensemble des courbes de genre 2.

Pour une courbe donnée, cette question se résoud par un calcul d'élimination. L'intérêt d'un précalcul est de fournir des formules prêtes à l'emploi, où il suffit de spécialiser les paramètres pour obtenir le polynôme minimal recherché : elles peuvent être utilisées indépendamment d'un système de résolution d'équations polynomiales. Nous avons étudié tous les cas de dégénérescence possibles, suivant les travaux d'Igusa [108] dans ce domaine. Dans ce mémoire, nous présentons le cas "générique", pour lequel les calculs sont les plus ardus ; l'article [75] donne davantage de détails.

22.2 Courbes hyperelliptiques

Nous donnons dans cette section la définition d'une courbe hyperelliptique et des isomorphismes entre courbes ; la loi de groupe n'est définie que dans la section suivante. La présentation est minimale, un exposé détaillé sortirait du cadre de ce mémoire. Les définitions et propriétés sont extraites de [97, 121], que l'on peut consulter pour plus de détails.

22.2.1 Définitions

Soit k un corps de caractéristique différente de 2, 3 et 5. Les courbes mentionnées dans la suite sont a priori définies sur k ; nous serons également amenés à considérer des courbes isomorphes définies sur une extension algébrique de k.

Définition 19 (Courbe hyperelliptique) *Une courbe hyperelliptique $\mathcal{C}$ est une courbe projective non singulière, telle qu'il existe un morphisme fini de degré 2 de $\mathcal{C}$ vers $\mathbb{P}^1(k)$.*

Si $\mathcal{C}$ est une courbe hyperelliptique de genre g, elle est birationnellement équivalente à une courbe plane de partie affine

$$y^2 = f(x),$$

où f est un polynôme de degré $2g + 1$ ou $2g + 2$ sans racine multiple. Dans la suite, nous nous intéresserons essentiellement aux cas particuliers suivants :

- Toute courbe de genre 1 est hyperelliptique, mais on dit qu'elle est *elliptique*. Sous nos hypothèses sur la caractéristique de k, toute courbe elliptique est birationnellement équivalente à une courbe plane non singulière donnée par une équation de la forme :

$$y^2 = x^3 + x + b.$$

- Toute courbe de genre 2 est hyperelliptique. Sous nos hypothèses sur la caractéristique de k, toute courbe hyperelliptique de genre 2 est birationnellement équivalente à une courbe plane non singulière donnée par une équation de la forme :

$$y^2 = x^5 + x^3 + bx^2 + cx + d.$$

On munit une courbe hyperelliptique d'un point privilégié noté P_∞, pour lequel on choisit usuellement le point à l'infini de la courbe.

Définition 20 (Involution hyperelliptique) *Soit $\mathcal{C}$ une courbe hyperelliptique donnée dans un modèle affine plan. Si $P = (x,y)$ est un point de $\mathcal{C}$, nous appellerons opposé de P le point $-P = (x, -y)$. L'application $P \mapsto -P$ est appelée involution hyperelliptique et notée ι.*

22.2.2 Isomorphismes entre courbes hyperelliptiques

Les courbes hyperelliptiques peuvent être classifiées à isomorphisme près par des *invariants* bien choisis ; puisque ces courbes sont non singulières, cette classification se ramène à la classification des modèles plans affines à équivalence birationnelle près.

Le j-invariant des courbes elliptiques. Les courbes elliptiques sont classifiées par leur invariant modulaire noté j. Si la courbe est donnée par l'équation $y^2 = x^3 + Ax + B$, l'invariant j s'écrit

$$j = 1728 \frac{4A^3}{4A^3 + 27B^2}.$$

Les modules d'une courbe de genre 2. Les courbes de genre 2 sont classifiées par 3 invariants, appelés *modules*. Leur construction se base sur des formes covariantes, notées (A,B,C,D) depuis Igusa [108] ; les modules notés (j_1,j_2,j_3) sont construits comme des quotients de ces covariants, valables quand le covariant A est non nul. Dans le chapitre 25, nous les expliciterons dans un cas particulier.

Formes des isomorphismes. Soit $\mathcal{C}$ une courbe hyperelliptique de genre $g \geq 2$ donnée dans un modèle plan affine. Une courbe plane $\mathcal{C}'$ est isomorphe à $\mathcal{C}$ si et seulement s'il existe des éléments (a,b,c,d,w) de k tels que $ad - bc$ et w sont non nuls, et que l'on passe de $\mathcal{C}$ à $\mathcal{C}'$ par l'équation

$$(x',y') = \left(\frac{ax+b}{cx+d}, \frac{wy}{(cx+d)^{g+1}}\right),$$

d'après par exemple Mestre [158].

22.3 Jacobienne d'une courbe

Nous définissons ici la Jacobienne d'une courbe, qui est un groupe abélien construit à partir de sommes formelles de points. Cette construction mène à de nombreux développements, en particulier sur les corps finis, où la question du cardinal d'une Jacobienne est de première importance (voir le livre de Koblitz [121]).

22.3.1 Diviseurs et Jacobienne

Définition 21 *Soit $\mathcal{C}$ une courbe hyperelliptique.*

- *Nous appellerons* diviseur, *et noterons souvent D, une somme finie de points de $\mathcal{C}$, à coefficients dans $\mathbb{Z}$:*
$$D = \sum n_i P_i.$$
Nous appellerons degré *d'un diviseur la somme des n_i. L'ensemble des diviseurs est naturellement muni d'une structure de groupe.*
- *Nous appellerons* diviseur principal *un diviseur de la forme $\sum P_i - \sum Q_i$, tel qu'il existe une fraction rationnelle f définie sur $\mathcal{C}$ admettant les P_i pour zéros et les Q_i pour pôles. Un tel diviseur a pour degré zéro, et l'ensemble des diviseurs principaux forme un sous-groupe de l'ensemble des diviseurs.*
- *Nous appellerons* Jacobienne *de la courbe $\mathcal{C}$ et noterons* **Jac**$(\mathcal{C})$ *le quotient du sous-groupe des diviseurs de degré zéro par le sous-groupe des diviseurs principaux.*

Si la courbe $\mathcal{C}$ est de genre 1, elle s'identifie à sa Jacobienne.

Proposition 72 (Diviseurs réduits) *Soit $\mathcal{C}$ une courbe hyperelliptique de genre g. On peut écrire tout élément de la Jacobienne comme une somme de points de la forme*

$$D = \sum m_i P_i - (\sum m_i) P_\infty,$$

où tous les m_i sont positifs, les P_i sont des points affines, on ne trouve pas simultanément P_i et $-P_i$, et enfin $\sum m_i$ est inférieur à g. Nous appellerons une telle somme de points un diviseur réduit, *et poids de D l'entier $\sum m_i$.*

Dans la suite, de manière abusive, nous appellerons souvent diviseur un élément de la Jacobienne, et nous omettrons le terme d'équilibre $-\sum m_i P_\infty$.

Théorème 28 *Soit $\mathcal{C}$ une courbe hyperelliptique. On peut munir la Jacobienne de $\mathcal{C}$ d'une structure de groupe algébrique.*

En d'autres termes, la loi de groupe est donnée par des fractions rationnelles sur la Jacobienne de $\mathcal{C}$. La sous-section suivante fournit une représentation commode des diviseurs réduits, la *représentation de Mumford*. Les calculs des chapitres 23 et 24 adoptent cette représentation pour manipuler les diviseurs.

Notations La somme des diviseurs D_1 et D_2 est notée $D_1 + D_2$, et la somme $D + \cdots + D$ (ℓ fois) est notée $[\ell]D$. Quand le contexte ne permet pas la confusion, nous noterons l'élément neutre 0.

22.3.2 Représentation de Mumford d'un diviseur

Soit $\mathcal{C}$ une courbe hyperellitique de genre g donnée par l'équation

$$y^2 = f(x) = x^{2g+1} + f_{2g-1}x^{2g-1} + \cdots + f_1 x + f_0.$$

Nous adopterons la représentation dite de Mumford [165] des diviseurs réduits de $\mathcal{C}$ (Mumford attribue cette représentation à Jacobi) :

$$D = \langle u(x), v(x) \rangle = \langle x^g + u_{g-1}x^{g-1} + \cdots + u_0, v_{g-1}x^{g-1} + v_{g-2}x^{g-2} + \cdots + v_0 \rangle.$$

Si $D = \sum P_i$, avec $P_i = (x_i, y_i)$ dans $\mathcal{C}$, le polynôme u est $\prod(x - x_i)$ et le polynôme v interpole les valeurs des y_i aux points x_i, en respectant les multiplicités — autrement dit, on demande que u divise $v^2 - f$.

Exemple :

- Un diviseur $P_1 = (x_1, y_1)$ de poids 1 est représenté par $\langle x - x_1, y_1 \rangle$.
- Dans le cas d'une courbe $\mathcal{C}$ de genre 2, la représentation de Mumford des diviseurs de la Jacobienne de $\mathcal{C}$ est :

$$D = \langle u(x), v(x) \rangle = \langle x^2 + u_1 x + u_0, v_1 x + v_0 \rangle.$$

Remarque Cette représentation peut être vue comme la mise sous forme "Shape Lemma" de l'idéal de dimension zéro des points P_i "comptés avec multiplicités". Il pourrait être intéressant pour les algorithmes manipulant ces diviseurs d'utiliser d'autres descriptions de cet idéal. On pense en particulier à une représentation de Kronecker, pour un diviseur formé de points simples :

$$D = \langle u(x), \frac{w(x)}{u'(x)} \rangle,$$

où $w = vu' \mod u$. Si u admet des facteurs multiples, on peut garder trace des multiplicités et conserver une représentation similaire, voir [9, 185].

22.3.3 Isogénies

La notion d'isogénie est définie dans le cadre de l'étude des variétés abéliennes ; le théorème 28 nous dit précisément qu'une courbe hyperellitique est une variété abélienne. Nous ne nous intéresserons dans la suite qu'à des isogénies de formes très particulières, mais donnons la définition dans le cadre général.

Définition 22 (Isogénies) *Soient $\mathcal{C}$ et $\mathcal{C}'$ deux variétés algébriques abéliennes. On appelle isogénie de $\mathcal{C}$ vers $\mathcal{C}'$ un morphisme φ de $\mathcal{C}$ dans $\mathcal{C}'$ compatible avec la loi de groupe, surjectif et à noyau fini. On appelle degré d'une isogénie le cardinal de son noyau.*

Multiplication. Le premier type d'isogénie que avons rencontré est la multiplication par un entier ℓ dans la Jacobienne d'une courbe hyperellitique $\mathcal{C}$, notée $[\ell]$. Nous appellerons *éléments de ℓ-torsion* les éléments du noyau de cette application ; ils forment un sous-groupe de $\mathbf{Jac}(\mathcal{C})$, noté $\mathbf{Jac}[\ell]$:

$$\begin{aligned}\mathbf{Jac}[\ell] &= \{D \in \mathbf{Jac}(\mathcal{C}), D + \cdots + D(\ell\,\text{fois}) = 0\} \\ &= \{D \in \mathbf{Jac}(\mathcal{C}), [\ell]D = 0\}.\end{aligned}$$

On peut montrer que si D est un diviseur de ℓ-torsion, alors son opposé par l'involution hyperelliptique est également un diviseur de ℓ-torsion.

Les diviseurs de torsion interviendront dans le chapitre 23, et surtout dans le chapitre 24, où ils sont à la base de la construction des équations modulaires. Dans la représentation de Mumford, les diviseurs de ℓ-torsion sont caractérisés par les idéaux de ℓ-torsion, dont on trouve la définition et les propriétés dans [177, 116].

Définition 23 (Idéal de ℓ-torsion) *L'idéal de ℓ-torsion de $\mathcal{C}$, noté $\mathcal{I}_\ell$ est l'ensemble des polynômes P de $K[u_0, u_1, \ldots, u_{g-1}, v_0, v_1, \ldots, v_{g-1}]$ tels que pour tout diviseur $D = \langle u(x), v(x)\rangle$ de poids g et de ℓ-torsion, donné dans sa représentation de Mumford, on a*

$$P(u_0, u_1, \ldots, u_{g-1}, v_0, v_1, \ldots, v_{g-1}) = 0.$$

Proposition 73 *Pour une courbe $\mathcal{C}$ de genre g, pour tout nombre premier impair ℓ, il existe en tout $\ell^{2g} - 1$ diviseurs de ℓ-torsion de poids g.*

Jacobiennes décomposables. Le chapitre 25 s'intéresse à des isogénies d'une forme bien particulière. Si $\mathcal{C}$ est une courbe hyperelliptique de genre 2, on dit que la Jacobienne de $\mathcal{C}$ est *(2,2)-décomposable* si elle est isogène à un produit de 2 courbes elliptiques $\mathcal{E}_1 \times \mathcal{E}_2$, et si le noyau de l'isogénie est isomorphe à $(\mathbb{Z}/2\mathbb{Z})^2$. On dit alors que $\mathcal{E}_1$ et $\mathcal{E}_2$ sont des quotients de degré 2 de $\mathcal{C}$.
D'après Igusa [108], il suffit d'exhiber un morphisme surjectif de degré 2 de la courbe $\mathcal{C}$ vers une courbe elliptique pour être assurés que la Jacobienne de $\mathcal{C}$ est (2,2)-décomposable. Dans le chapitre 25, nous utiliserons cette propriété.

Chapitre 23

Division par deux dans une Jacobienne

Résumé

Ce chapitre décrit la résolution d'un système paramétré en application de l'algorithme de la partie V. Ce système a été résolu en réponse à une question de P. Gaudry et R. Harley, dans le contexte de la détermination du cardinal de la Jacobienne d'une courbe de genre 2, décrit dans leur article [74].

Sommaire

Introduction

Ce chapitre présente le résultat d'une collaboration avec P. Gaudry du Laboratoire d'Informatique de l'École polytechnique et R. Harley du projet CRISTAL de l'INRIA Rocquencourt, pour leur article [74]. Ma contribution est la résolution explicite d'un système polynomial dépendant de paramètres, défini sur un corps fini de cardinal de l'ordre de 10^{19}. Des questions mises à jour lors de l'étude de ce système sont à la base des résultats du chapitre suivant.

La problématique générale est la détermination du cardinal de la Jacobienne d'une courbe de genre 2, définie sur un corps fini k. Les motivations sont nombreuses, en particulier dans le domaine de la cryptographie : le livre de Koblitz [121] donne davantage de détails sur ce sujet. L'algorithme de Schoof [193] décrit le calcul du cardinal (de la Jacobienne) d'une courbe *elliptique* $\mathcal{E}$ à partir de la connaissance de coordonnées de *points de torsion* de ce groupe. En genre 2, Gaudry et Harley proposent dans [74] un algorithme d'inspiration similaire, où la détermination du cardinal passe par la description de points de torsion de la Jacobienne.

Principe : Soient $\mathcal{C}$ une courbe hyperelliptique définie sur un corps fini k, ϕ l'endomorphisme de Frobenius sur la Jacobienne de $\mathcal{C}$, et $\chi(t)$ son polynôme caractéristique. Alors on a l'égalité

$$\#\mathbf{Jac}(\mathcal{C}) = \chi(1).$$

Soit ℓ un nombre premier, différent de la caractéristique de k. L'endomorphisme de Frobenius agit linéairement sur le groupe $\mathbf{Jac}[\ell]$ des diviseurs de ℓ-torsion, et un théorème de Tate affirme que le polynôme caractéristique de l'endomorphisme induit est exactement la réduction de χ modulo ℓ. Ainsi, la connaissance des éléments de $\mathbf{Jac}[\ell]$ et de l'action de l'endomorphisme de Frobenius permet de déterminer le cardinal de la Jacobienne modulo ℓ. Procédant pour plusieurs valeurs de ℓ, on peut reconstruire le cardinal de la Jacobienne de $\mathcal{C}$ par le théorème des restes chinois. En pratique, la connaissance explicite des coordonnées d'un point de ℓ-torsion suffit à éliminer des candidats pour le cardinal de la Jacobienne modulo ℓ. De manière analogue, la connaissance de diviseurs de ℓ^i-torsion permet de supprimer des candidats pour le cardinal de la Jacobienne modulo ℓ^i.

Dans l'exemple sur lequel je suis intervenu, les méthodes de la partie V ont été appliquées pour rechercher des points de 2^i-torsion, pour des valeurs de i allant jusqu'à 8, et ont contribué à l'aboutissement du calcul du cardinal.

23.1 Présentation du problème

Soit $\mathcal{C}$ une courbe de genre 2, définie sur un corps k et donnée par l'équation à coefficients dans k :

$$\mathcal{C} \ : y^2 = x^5 + p_1x^4 + p_2x^3 + p_3x^2 + p_4x + p_5.$$

Rappelons qu'en genre 2, la représentation de Mumford des diviseurs de la Jacobienne de $\mathcal{C}$ est :

$$D = \langle u(x), v(x)\rangle = \langle x^2 + u_1x + u_0, v_1x + v_0\rangle.$$

Les valeurs (u_0,u_1,v_0,v_1) apparaissent comme des coordonnées avec lesquelles calculer.

En particulier, les coefficients de la forme de Mumford de $[2]D$ s'écrivent comme des fractions rationnelles en les coefficients de D, à coefficients dans k, dépendant de l'équation de $\mathcal{C}$: ce sont les formules d'addition de Cantor [42], qui permettent de voir le problème de la division par deux comme un système polynomial dépendant de paramètres. Ainsi, il existe des polynômes $(f_1,\dots,f_4,g)$ tels que si D est un diviseur de coordonnées (u_0,u_1,v_0,v_1), le diviseur D' de coordonnées (U_0,U_1,V_0,V_1) est le double de D si et seulement si

$$\begin{array}{rcl} f_1(u_0,u_1,v_0,v_1,U_0,U_1) & = & 0, \\ f_2(u_0,u_1,v_0,v_1,U_0,U_1) & = & 0, \\ f_3(u_0,u_1,v_0,v_1,U_0,U_1,V_0,V_1) & = & 0, \\ f_4(u_0,u_1,v_0,v_1,U_0,U_1,V_0,V_1) & = & 0, \\ g(u_0,u_1,v_0,v_1) & \neq & 0. \end{array}$$

À l'aide de ces équations, la donnée des coordonnées d'un diviseur de 2^i-torsion D_i permet de calculer les coordonnées d'un diviseur de 2^{i+1}-torsion : on résout le système précédent en spécialisant les paramètres (U_0,U_1,V_0,V_1) en les coordonnées de D_i. Il y a en général $2^{2g} = 16$ solutions à ce système, qui diffèrent l'une de l'autre par un diviseur de 2-torsion. On trouve donc les coordonnées de D_{i+1} dans une extension du corps de définition de D_i, par la résolution d'un système de dimension zéro, à coefficients dans le corps de définition de D_i.
À partir de coordonnées d'un diviseur de 2-torsion, l'application itérée de ce procédé fournit les coordonnées de diviseurs de 2^i-torsion. Hélas, les spécialisations des paramètres se font sur des nombres algébriques de degrés de plus en plus élevés, et les calculs deviennent vite impossibles. Il a paru préférable de calculer une fois pour toutes la résolution générique, pour ensuite la spécialiser sur les coordonnées des points D_i.
Nous ne donnons pas ce premier jeu d'équations ici : elles sont complexes, et nous n'avons pas cherché à les résoudre telles qu'elles.

23.2 Des équations simplifiées

Les équations $(f_1,\dots,f_4)$ dépendent de 4 paramètres, ce qui n'est pas négligeable. Nous modifions le système, suivant une suggestion de R. Harley : nous allons résoudre le système ci-dessous, qui ne dépend plus que de deux paramètres. Les caractéristiques de ce système sont données dans le tableau de la sous-section suivante ; le système est écrit dans l'annexe E.2, et baptisé P19 .

$$\begin{array}{rcl} f_1(u_0,u_1,v_0,v_1,U_0,U_1) & = & 0, \\ f_2(u_0,u_1,v_0,v_1,U_0,U_1) & = & 0, \\ \tilde{f}_3(u_0,u_1,v_0,v_1) & = & 0, \\ \tilde{f}_4(u_0,u_1,v_0,v_1) & = & 0, \\ g(u_0,u_1,v_0,v_1) & \neq & 0. \end{array}$$

Les nouvelles équations $\tilde{f}_3$ et $\tilde{f}_4$ imposent que le point (u_0,u_1,v_0,v_1) est sur la courbe $\mathcal{C}$. En contrepartie, nous perdons la dépendance en (V_0,V_1) : ce nouveau système ne dépend plus que de l'abscisse du diviseur D_i. Celle-ci fixée, 4 ordonnées sont possibles : notre nouveau système va donc avoir génériquement $16 \times 4 = 64$ solutions.
Une fois cette résolution générique spécialisée en un point D_i, nous pouvons retrouver les 16 solutions qui nous intéressent, par un calcul de pgcd faisant intervenir les équations (f_3,f_4) qui avaient été écartées.

23.3 Mise en pratique

Le but de l'article [74] est de calculer le cardinal de la Jacobienne d'une courbe "tirée au hasard" sur le corps $k = \mathbb{F}_p$, où $p = 10^{19} + 51$ — le modulo choisi est le plus petit nombre premier supérieur à 10^{19}. La courbe retenue est

$$\mathcal{C} : y^2 = x^5 + p_1 x^4 + p_2 x^3 + p_3 x^2 + p_4 x + p_5, \text{ avec}$$

$$\begin{aligned} p_1 &= 3141592653589793238, \\ p_2 &= 4626433832795028841, \\ p_3 &= 9716939937510582097, \\ p_4 &= 4944592307816406286, \\ p_5 &= 2089986280348253421. \end{aligned}$$

Nous allons résoudre les équations $(f_1, f_2, \tilde{f}_3, \tilde{f}_4)$ associées à cette courbe, en fonction des paramètres (U_0, U_1).

FIG. 23.1 – *Division par deux dans une Jacobienne*

Variables	Paramètres	Multiplications	Degrés
4	2	253	(4,4,9,10)

Pour déterminer une résolution paramétrées des points simples génériques, nous obtenons les temps suivants, sur une machine Compaq XP/1000 - Alpha EV6 de l'UMS MEDICIS. Le système nous est donné sous une forme développée : même après passage par la fonction `optimize` de Maple, le nombre de multiplications est relativement élevé, ce qui a comme conséquence un temps de calcul assez important. Il existe certainement une forme qui s'évalue mieux, et pour la trouver, il faudrait remonter à la conception des polynômes de Cantor.

Temps, mémoire	Degré générique	Degré en (U_0, U_1)
2.6 h., 30 Mo.	64	44

Validité et utilisation du résultat. La résolution a le degré générique 64 prévu par la théorie : il ne manque aucun point. Nous pouvons appliquer les résultats de la section 10.6 du chapitre 10 pour se convaincre qu'elle est correcte : l'estimation des probabilités d'erreur du calcul et de la vérification a été donnée dans le chapitre 11.

En outre, cette résolution a été utilisée pour déterminer des points de 2^i-torsion successifs de la courbe $\mathcal{C}$, et nous n'avons pas décelé d'erreur dans les résultats qu'elle a fournis.
Le tableau suivant, tiré de [74], décrit les calculs des diviseurs de 2^i-torsion D_i. La première colonne indique l'information sur le cardinal de la Jacobienne de $\mathcal{C}$ que l'on peut tirer de la connaissance

de D_i. La colonne "degré" indique le degré de l'extension de k où vit le diviseur en question, et la colonne "temps" indique le temps nécéssaire pour spécialiser la résolution générique en D_i, en extraire les 16 solutions qui nous intéressent par un calcul de pgcd, et déterminer les candidats possibles pour le cardinal de la Jacobienne (ce qui passe par une étape de factorisation). Le diviseur de 2-torsion initial est obtenu à partir de la factorisation du polynôme $x^5 + p_1x^4 + p_2x^3 + p_3x^2 + p_4x + p_5$ (voir [74] pour plus de détails).

$\#\mathbf{Jac}(\mathcal{C})$	Degré	$\#\mathbf{Jac}(\mathcal{C})$	Degré	Temps
0 mod 2	1	16 mod 32	16	
0 mod 4	1	48 mod 64	32	
0 mod 8	4	48 mod 128	64	5000 s.
0 mod 16	8	176 mod 256	128	9 h.

Nous avions mentionné dans la première section une stratégie plus naturelle, qui consiste à itérer des calculs en dimension zéro, en spécialisant les paramètres sur les coordonnées des points de 2^i-torsion successifs. En Magma, les calculs deviennent impossibles pour l'ordre 64, où les paramètres sont algébriques de degré 32 sur le corps de base.

Chapitre 24

Calcul d'équations modulaires

Résumé

Nous proposons dans ce chapitre une définition d'*équations modulaires* pour une courbe hyperelliptique $\mathcal{C}$ de genre quelconque. En genre fixé, ces équations sont indicées par un nombre premier impair ℓ, et décrivent en partie les idéaux de ℓ-torsion de $\mathcal{C}$.

Notre approche repose sur une étude des idéaux de torsion par des considérations algébriques élémentaires : la connaissance d'actions de groupes explicites permet de décrire la structure des extensions associées à ces idéaux. Les équations modulaires sont les polynômes minimaux d'invariants bien choisis associés à ces groupes, dont le degré est inférieur au degré des extensions considérées.

La construction explicite de ces équations repose sur des calculs mettant en jeu des ensembles triangulaires ; c'est une application de l'algorithme de la partie VII. Les calculs ne sont pas simples, et nous nous limitons aux cas suivants.

- En genre 1, nous pouvons monter jusqu'à des valeurs de ℓ de l'ordre de quelques dizaines. Nous utilisons une technique d'approximation de Hermite-Padé qui n'a pas été encore été présentée dans ce mémoire, mise en œuvre dans la section 24.3.
- En genre 2, nous n'avons pour l'instant cherché à résoudre que le cas $\ell = 3$.

Sommaire

24.1 Introduction

Ce chapitre se place dans la continuité du chapitre 23 : la motivation est toujours de décrire les idéaux de torsion d'une courbe hyperelliptique sur un corps fini, pour obtenir des informations sur le cardinal de sa Jacobienne. Les résultats décrits ici sont le fruit d'une collaboration avec P. Gaudry, du Laboratoire d'Informatique de l'École polytechnique ; ce travail est toujours en cours.

Pour les courbes de genre 1, la littérature présente une variété d'équations dites "modulaires". Ces différentes équations ne sont pas fondamentalement différentes les unes des autres, on souvent passe de l'une à l'autre par un changement de variables. Typiquement, il s'agit de familles de polynômes univariés, indicées par l'ensemble des nombres premiers, à coefficients rationnels et dépendant d'un paramètre. Pour un nombre premier ℓ donné, si $\mathcal{E}$ est une courbe elliptique définie sur un corps fini, la spécialisation du paramètre en (une fonction du) j-invariant de $\mathcal{E}$ fournit un polynôme univarié à coefficients dans le corps de définition de $\mathcal{E}$, et ses *motifs de factorisation* fournissent en particulier des informations sur le cardinal de $\mathcal{E}$ modulo ℓ.

Dans ce chapitre, nous proposons une construction d'équations qui vont satisfaire des propriétés analogues, mais en genre arbitraire. La construction est élémentaire, et permet d'effectuer les calculs par les méthodes developpées jusqu'ici. Nos équations modulaires sont des polynômes minimaux d'éléments bien choisis, indicés par un nombre premier ℓ. Pour une courbe donnée, on pourra obtenir l'équation recherchée par des calculs d'élimination. Notre objectif est d'écrire ces polynômes minimaux pour une courbe à *coefficients génériques*, afin de ne pas avoir à refaire le calcul d'élimination pour chaque courbe qui se présente, et éventuellement les destiner à des bases de données, comme c'est le cas en genre 1.

Pour un genre g fixé, nous noterons Ξ_ℓ la ℓème équation modulaire ; elle sera définie dans la prochaine section. Ξ_ℓ est un polynôme univarié dont les coefficients sont des fractions rationnelles en les paramètres nécessaires pour décrire l'ensemble des courbes de genre g, à coefficients rationnels. Le théorème suivant, dû à P. Gaudry, est à la base d'une utilisation de ces équations.

Théorème 29 *Soient $\mathcal{C}$ une courbe de genre g définie sur le corps $\mathbb{F}_p$, et Ξ_ℓ la ℓième équation modulaire pour le genre g ; notons $\Xi_\ell(\mathcal{C})$ l'équation Ξ_ℓ où on a spécialisé les paramètres dont dépend Ξ_ℓ en les coefficients de $\mathcal{C}$, prise modulo p.*
Soit ϕ_ℓ l'endomorphisme de Frobenius de $\mathbf{Jac}(\mathcal{C})$ restreint à l'espace vectoriel des diviseurs de ℓ-torsion. La réduction de Jordan de cet endomorphisme détermine :

- *le cardinal de $\mathbf{Jac}(\mathcal{C})$ modulo ℓ,*
- *le motif de factorisation de $\Xi_\ell(\mathcal{C})$.*

La section 24.4 donne une application concrète de ce résultat en genre 2, pour $\ell = 3$.

Nous définirons les polynômes Ξ_ℓ, à partir des idéaux de ℓ-torsion, indépendamment du système d'équations utilisé pour les décrire. La construction repose sur une étude des quotients associés : des actions de groupes permettent d'en préciser la structure. Nous verrons apparaître l'intérêt de les décrire à l'aide d'ensembles triangulaires, qui mettent en jeu des invariants associés à ces groupes. Dans la section 24.3, nous appliquons la construction au cas des courbes de genre 1, et retrouvons un résultat obtenu par Charlap, Coley et Robbins dans [46]. Dans ce cas, les idéaux de ℓ-torsion sont décrits à l'aide des *polynômes de division*, qui sont définis par récurrence sur ℓ. Nous menons les calculs à partir de ce système d'équations ; les techniques utilisées sont dans la continuité des algorithmes présentés jusqu'ici, et reposent en outre sur des approximations de type Hermite-Padé. La construction est valable en genre supérieur à 1, et dans ce cas nous n'avons pas trouvé trace dans la littérature d'équations analogues. Pour une construction explicite, nous nous limitons au genre 2, où les idéaux de ℓ-torsion peuvent être décrits à partir des formules de Cantor. Pour le cas $\ell = 3$, nous calculons Ξ_3 dans la section 24.4, par une application de l'algorithme de la partie VII ; les temps de calculs et la taille de la sortie restent raisonnables. Le cas $\ell = 5$ est envisagé, mais n'a pas encore été traité.

24.2 Présentation générale

Soit $\mathcal{C}$ une courbe de genre g, définie sur un corps K, donnée dans un modèle affine plan ; nous supposons $\mathcal{C}$ "suffisamment générique". Nous nous intéressons aux diviseurs de ℓ-torsion de la Jacobienne de $\mathcal{C}$, pour un entier premier ℓ. Les diviseurs D de $\mathcal{C}$ seront représentés sous la forme de Mumford :

$$D = \langle u(x),v(x) \rangle = \langle x^g + u_{g-1}x^{g-1} + \cdots + u_0, v_{g-1}x^{g-1} + v_{g-2}x^{g-2} + \cdots + v_0 \rangle.$$

Comme dans le chapitre 23, les grandeurs (u_i,v_i) sont vues comme des coordonnées sur l'ensemble des diviseurs. Nous noterons A_g l'anneau de polynômes correspondant :

$$A_g = K[u_0, \ldots, u_{g-1}, v_0, \ldots, v_{g-1}].$$

Dans les coordonnées (u_i,v_i), l'idéal de ℓ-division $\mathcal{I}_\ell$ présenté dans la section 22.3.3 du chapitre 22 est l'idéal de A_g annulant l'ensemble des $\ell^{2g} - 1$ diviseurs de ℓ-torsion de poids g, et dans nos coordonnées, il est défini par les formules de Kampkötter [116].

Nous allons étudier l'extension

$$K \to A_g/\mathcal{I}_\ell,$$

en l'écrivant comme une succession d'extensions

$$K \to K_1 \to K_2 \to A_g/\mathcal{I}_\ell.$$

Pour cela, nous allons décrire l'action de deux groupes sur les points décrits par $\mathcal{I}_\ell$. Des invariants associés à ces actions de groupe vont fournir des éléments non séparants qui permettent de décrire ces extensions par des ensembles triangulaires.

24.2.1 L'action de $\{1,\iota\}$

Dans un premier temps, nous considérons l'action du groupe $\{1,\iota\}$, où ι est l'involution hyperelliptique $P \mapsto -P$ définie dans le chapitre 22.

Soit D un diviseur réduit, écrit sous la forme $\sum P_i - rP_\infty$. Une fois D mis sous forme de Mumford, u_{g-1} est la somme des abscisses des points P_i. Le diviseur opposé $-D$ a même abscisse, donc même coordonnée u_{g-1}, que D, et son ordonnée est l'opposée de celle de D : u_{g-1} est un invariant pour l'action du groupe $\{1,\iota\}$.

Si D est un diviseur de torsion, ιD l'est également. Les orbites de l'action de $\{1,\iota\}$ sur les diviseurs de torsion sont les couples $(D, -D)$, et si $\mathcal{C}$ est une courbe générique, la coordonnée u_{g-1} distingue ces couples. Nous en déduisons que u_{g-1} a un polynôme minimal de degré $\frac{\ell^{2g}-1}{2}$. L'idéal de ℓ-division peut donc être décrit par un ou plusieurs ensembles triangulaires, dont le dernier polynôme en u_{g-1} est de degré $\frac{\ell^{2g}-1}{2}$.

Définition 24 *Nous noterons ψ_ℓ le polynôme minimal de u_{g-1} dans le quotient $A_g/\mathcal{I}_\ell$. Pour toute valeur u_{g-1} racine de ψ_ℓ, nous noterons $\pm D_{u_{g-1}}$ le couple de diviseurs de ℓ-torsion correspondant. Les coefficients du polynôme ψ_ℓ sont des fractions rationnelles en les coefficients de la courbe $\mathcal{C}$.*

À ce stade, nous avons décrit l'extension $K \to A_g/\mathcal{I}_\ell$ sous la forme

$$K \to K_2 \to A_g/\mathcal{I}_\ell,$$

où les deux extensions sont de degrés respectifs $(\frac{\ell^{2g}-1}{2},2)$, et où l'extension $K \to K_2$ est engendrée par le polynôme ψ_ℓ. Le polynôme ψ_ℓ fournit des informations sur la structure de l'idéal de ℓ-torsion, mais il est de degré en u_{g-1} égal à $\frac{\ell^{2g}-1}{2}$; l'intérêt des équations modulaires sera de conserver de l'information sur les idéaux de torsion, avec un degré moindre.

24.2.2 L'action de $\mathbb{F}_\ell^*/\{\pm 1\}$

Si D est diviseur de ℓ-torsion, les diviseurs

$$\left\{-[\frac{\ell-1}{2}]D, \dots, -D,0,D,\dots,[\frac{\ell-1}{2}]D\right\}$$

forment un groupe cyclique. Les abscisses des diviseurs $-[k]D$ et $[k]D$ sont les mêmes, donc les abscisses de tous les multiples non nuls de D forment un ensemble de cardinal $\frac{\ell-1}{2}$. On peut munir l'ensemble des abscisses de ces diviseurs d'une action du groupe $\mathbb{F}_\ell^*/\{\pm 1\}$, correspondant la multiplication par un entier k de $\{1,\dots,\frac{\ell-1}{2}\}$, puisque l'addition commute avec l'involution ι. Cette action peut être rendue explicite :

Proposition 74 (Action sur les racines) *Pour tout entier k dans $\left[1,\frac{l-1}{2}\right]$, il existe un polynôme h_k de $K[t]$ tel que si u_{g-1} est une racine de ψ_ℓ correspondant aux diviseurs $\pm D_{u_{g-1}}$, $h_k(u_{g-1})$ est la racine correspondant au diviseur $[\pm k]D_{u_{g-1}}$. Les coefficients du polynôme h_k sont des fractions rationnelles en les coefficients de la courbe $\mathcal{C}$.*

Démonstration. Les formules d'addition dans une Jacobienne s'écrivent comme des fractions rationnelles. Modulo l'idéal de ℓ-torsion, elles s'écrivent sous forme polynomiale. Pour conclure, on remarque que l'abscisse du diviseur $[\pm k]D$ est fonction uniquement de l'abscisse de $\pm D$, donc de la variable u_{g-1}. □

De suite, si u_{g-1} est une racine de ψ_ℓ, la somme de ses $\frac{\ell-1}{2}$ images est invariante sous l'action de $\mathbb{F}_\ell^*/\{\pm 1\}$. Cela mène à la définition suivante.

Définition 25 (Équation modulaire) *Notons t_ℓ l'élément*

$$t_\ell = \sum_{1 \le k \le \frac{\ell-1}{2}} h_k(u_{g-1}).$$

Le polynôme minimal de t_ℓ dans le quotient $A_g/\mathcal{I}_\ell$ est appelé ℓème équation modulaire, et est noté Ξ_ℓ. Le polynôme Ξ_ℓ est de degré

$$\frac{\ell^{2g}-1}{\ell-1},$$

et ses coefficients sont des fractions rationnelles en les coefficients de $\mathcal{C}$.

Conclusion Nous avons décrit l'extension

$$K \to A_g/\mathcal{I}_\ell$$

comme

$$K \to K_1 \to K_2 \to A_g/\mathcal{I}_\ell,$$

où les extensions sont de degrés $(\frac{\ell^{2g}-1}{\ell-1},\frac{\ell-1}{2},2)$; l'équation modulaire est un polynôme engendrant la première extension $K \to K_1$, que l'on peut obtenir par un calcul d'élimination.

Les sections suivantes décrivent les calculs avec plus de détails, pour les courbes de genre 1 et 2. Notre objectif est d'obtenir ces polynômes minimaux pour des courbes à coefficients génériques, pour n'avoir ensuite qu'à les spécialiser quand une courbe particulière se présente.

Pour conclure cette présentation générale, faisons un peu de prospective. Le degré de l'extension $K \to K_1$ est $\frac{\ell^{2g}-1}{\ell-1}$. Pour tout g, ce polynôme se factorise : par exemple, il admet systématiquement le facteur $\ell+1$. Il est tentant de poursuivre le processus de "dévissage" en cherchant une décomposition de cette extension en rapport avec les degrés des facteurs qui apparaissent. C'est le sujet de travaux en cours.

24.3 Calculs en genre 1

Nous considérons dans cette section des courbes $\mathcal{E}(b)$ de genre 1, données par l'équation

$$\mathcal{E}(b) : y^2 = x^3 + x + b.$$

Dans ce cas, les équations modulaires Ξ_ℓ appartiennent à $\mathbb{Q}(b)[t_\ell]$, et sont de degré $\frac{\ell^2-1}{\ell-1} = \ell + 1$ en t_ℓ. Pour pouvoir les calculer, il faut décrire les idéaux de ℓ-torsion, ainsi que la multiplication d'un diviseur par un entier k. En genre 1, ces opérations sont effectuées à partir des *polynômes de division*, qui sont définis ci-dessous.

24.3.1 Polynômes de division

La courbe $\mathcal{E}(b)$ s'identifie à sa Jacobienne, et les diviseurs s'identifient aux points de la courbe, sous la forme

$$D : \langle x - x_0, y_0\rangle.$$

La coordonnée u_{g-1} que nous avons particularisée dans la présentation ci-dessus coïncide (au signe près) avec l'abscisse x_0 du point de la courbe correspondant.

Les polynômes de division ψ_ℓ sont définis par les relations de récurrence ci-dessous ; la définition construit des fractions rationnelles, qu'il faut considérer modulo l'équation définissant la courbe.

$$\begin{array}{rcl}
\psi_1 &=& 1,\\
\psi_2 &=& 2y,\\
\psi_3 &=& 3x^3 + 6x^2 + 12bx - 1,\\
\psi_4 &=& 4y(x^6 + 5x^4 + 20bx^3 - 5x^2 - 4bx - 8b^2 - 1),\\
\psi_{2\ell+1} &=& \psi_{\ell+2}\psi_\ell^3 - \psi_{\ell-1}\psi_{\ell+1}^3 \quad \text{si } \ell \geq 2,\\
\psi_{2\ell+2} &=& \psi_{\ell+1}(\psi_{\ell+3}\psi_\ell^2 - \psi_{\ell-1}\psi_{\ell+2}^2)/2y \quad \text{si } \ell \geq 2.
\end{array}$$

Pour tout ℓ impair, ψ_ℓ ne dépend que de x et a pour racines les abscisses des points de ℓ-torsion ; c'est un polynôme de degré $(\ell^2-1)/2$. L'idéal de ℓ-torsion est donc défini par $\big(\psi_\ell(x), y^2-(x^3+x+b)\big)$.

Enfin, pour tout k, la multiplication par k peut s'écrire comme :

$$[k](x,y) = (x - \frac{\psi_{k-1}\psi_{k+1}}{\psi_k^2}, \frac{\psi_{k+2}\psi_{k-1}^2 - \psi_{k-2}\psi_{k+1}^2}{4y\psi_k^3}).$$

24.3.2 Mise en équations

Nous travaillons avec 3 inconnues : (x,y) qui sont les coordonnées d'un point de $\mathcal{E}(b)$ et t_ℓ, qui est l'élément non séparant défini dans la section précédente. À ℓ fixé, ces grandeurs sont liées par le système d'équations ci-dessous, qui a ses coefficients dans $\mathbb{Q}(b)$.

$$\mathcal{F}_\ell \left\{ \begin{array}{rcl} y^2 - (x^3 + x + b) & = & 0, \\ \psi_\ell(x) & = & 0, \\ t_\ell - \sum_{1 \leq k \frac{k-1}{2}} x - \frac{\psi_{k-1}\psi_{k+1}}{\psi_k^2}(x,y) & = & 0. \end{array} \right.$$

Écrit ainsi, ce système est formé de fractions rationnelles. Pour se ramener à un système polynomial, il suffit de considérer les numérateurs de ces fractions ; nous donnons dans l'annexe E.3 le code Magma utilisé pour évaluer ces numérateurs. L'équation modulaire Ξ_ℓ est le polynôme minimal de t_ℓ dans ce système.

Exemple Pour $\ell = 5$, le système peut s'écrire comme suit ; l'objectif est de donner le polynôme minimal de t_5 en fonction de b.

$$\begin{array}{lcl} f_1 & : & -b - x^3 - x + y^2, \\ f_2 & : & -1728b^3x^3 - 1296b^2x^6 - 2592b^2x^4 + 432b^2x^2 - 256b^2y^4 - 324bx^9 - 1296bx^7 - 1080bx^5 + \\ & & 640bx^3y^4 + 432bx^3 - 128bxy^4 - 36bx - 27x^{12} - 162x^{10} - 297x^8 - 32x^6y^4 - 108x^6 + \\ & & 160x^4y^4 + 99x^4 - 160x^2y^4 - 18x^2 - 32y^4 + 1, \\ f_3 & : & 24bx + 6x^4 + 12x^2 - 16xy^2 + 8y^2t_5 - 2. \end{array}$$

La définition des polynômes de division par récurrence s'accomode bien d'une représentation par calcul d'évaluation : le nombre d'opérations effectuées est de l'ordre de 10ℓ, alors que le degré de ces polynômes croît de manière quadratique.

24.3.3 Résolution

Les zéros du système $\mathcal{F}_\ell$ peuvent être décrits par un ensemble triangulaire

$$\left\{ \begin{array}{l} T_1(y,x,t_\ell) \\ T_2(x,t_\ell) \\ \Xi_\ell(t_\ell) \end{array} \right.$$

où les polynômes (Ξ_ℓ,T_2,T_1) sont de degrés $(\ell + 1,\frac{\ell-1}{2},2)$ en leur variable principale, et ont leurs coefficients dans $\mathbb{Q}(b)$. Il est possible de résoudre ce système directement par les méthodes présentées dans la partie VII, mais ces calculs deviennent vite difficiles. Nous avons donné dans le chapitre 20 le temps nécessaire pour la résolution du cas $\ell = 13$.

Nous n'avons pas cherché à avancer davantage avec ces méthodes, mais proposons une technique d'inspiration analogue, qui donne des résultats prometteurs sur cette famille d'exemples, tout en restant à l'état expérimental ; nous n'avançons ici aucun résultat, nous nous contentons d'une présentation générale. Le principe est toujours d'utiliser une méthode de remontée de Newton, cette fois sur un ensemble triangulaire décrivant *un seul point* à coordonnées dans un quotient $\mathbb{Z}/p\mathbb{Z}$. La difficulté est de trouver un tel point : pour un système quelconque, rien ne dit que cela sera facile ; ici, la nature du problème suggère que cela pourrait l'être davantage.

Nous partons donc du postulat qu'il n'est pas "trop difficile" de déterminer explicitement une courbe $\mathcal{E}_0(b_0)$, un nombre premier p et un point de ℓ-torsion $P_0 = (x_0,y_0)$ pour la réduction de la courbe $\mathcal{E}_0(b_0)$ prise modulo p, *à coordonnées dans* $\mathbb{Z}/p\mathbb{Z}$. Cette question est certainement abordée dans la littérature ; pour l'instant, en pratique, nous avons cherché ces points au hasard, avec de bons résultats. On détermine la valeur $t_{\ell,0}$ de t_ℓ correspondant au point P_0, par la formule donnée dans la définition 25, pour ainsi disposer d'un point $(x_0,y_0,t_{\ell,0})$ à coordonnées dans $\mathbb{Z}/p\mathbb{Z}$ qui satisfait le système $\mathcal{F}_\ell$ pris modulo l'idéal $(p,b-b_0)$ de $\mathbb{Z}[b]$. Autrement dit, l'ensemble triangulaire

$$\begin{cases} y - y_0 \\ x - x_0 \\ t_\ell - t_{\ell,0} \end{cases}$$

apparaît dans la résolution du système $\mathcal{F}_\ell$ spécialisé en b_0, pris modulo p. Appliquons l'opérateur de Newton sur cet ensemble triangulaire, pour remonter selon les puissances du premier p et de l'idéal $(b - b_0)$, à une précision "suffisante". Nous obtenons un développement de t_ℓ en une série à coefficients dans $\mathbb{Z}/p^k\mathbb{Z}$, autour de b_0 et nous savons que cette série est l'amorce d'une solution du polynôme Ξ_ℓ de $\mathbb{Q}(b)[t_\ell]$

Arrivés à ce stade, la reconstruction de ce polynôme est une question d'algèbre linéaire, pour laquelle existent des méthodes adaptées, telles l'algorithme d'approximation de Hermite-Padé [21, 22, 59] ; l'article [45] propose un algorithme similaire pour le calcul d'une résolution géométrique sur les rationnels, en utilisant l'algorithme LLL de Lenstra, Lentra et Lovàcz [145]. Nous n'avons pas implanté de telle méthode, mais uniquement résolu un système linéaire dont les solutions sont les coefficients du polynôme Ξ_ℓ. Pour ℓ de l'ordre de quelques dizaines, la résolution de ce système prend quelques heures sur une machine Compaq XP/1000 - Alpha EV6. Nous avons bon espoir de réduire ces temps par les méthodes mentionnées ci-dessus.

Exemple Nous obtenons pour $\ell = 11$:

$$\begin{aligned} \Xi_{11}(t_{11}) \quad = \quad & t_{11}^{12} + 550t_{11}^{10} + 27500bt_{11}^9 - 103125t_{11}^8 - 1650000bt_{11}^7 + (-136884000b^2+ \\ & 645700)t_{11}^6 + 20625000bt_{11}^5 + (35793120b^2 - 11407385)t_{11}^4+ \\ & (340419200b^3 - 586141600b)t_{11}^3 + (-1758329760b^2 - 2177802)t_{11}^2+ \\ & (-2350167040b^3 + 1351692b)t_{11} - 1106800640b^4 + 6297984b^2 - 321651. \end{aligned}$$

24.4 Calculs en genre 2

Dans cette section, nous considérons des courbes de genre 2, données par l'équation

$$\mathcal{C} \ : y^2 = x^5 + x^3 + p_3x^2 + p_4x + p_5.$$

Dans ce cas, l'équation modulaire Ξ_ℓ est un polynôme de $\mathbb{Q}(p_3,p_4,p_5)[t_\ell]$, de degré en t_ℓ $\frac{\ell^4-1}{\ell-1} = \ell^3 + \ell^2 + \ell + 1$. La mise en équations n'utilise pas les générateurs des idéaux de torsion donnés par Kampkötter, qui sont très complexes : nous préfèrons revenir aux *formules d'addition de Cantor*. Nous allons nous concentrer sur le cas $\ell = 3$; le calcul dans le cas $\ell = 5$ n'a pas encore été effectué.

24.4.1 Application explicite

Avant de commencer les calculs, donnons-en une application. Dans le cas $\ell = 3$, le polynôme que nous allons calculer commence ainsi :

$$\begin{aligned}
\Xi_3 = {} & t_3^{40} + 312t_3^{38} - 7904p_3t_3^{37} + (-16344 + 183488p_4)t_3^{36} + (337536p_3 - 4179456p_5)t_3^{35} + \\
& (345456 + 464064p_4 - 2381088p_3^2)t_3^{34} + (-6044256p_3 - 196322688p_5 + 33608832p_3p_4)t_3^{33} + \\
& (-3966084 - 58451616p_4 + 52440768p_3^2 + 1324005696p_3p_5 - 364387392p_4^2)t_3^{32} + \\
& (57911232p_3 + 280858752p_5 + 591085824p_3p_4 - 241107200p_3^3 - 1357926912p_4p_5)t_3^{31} + \\
& (26487216 + 388684800p_4 - 378838944p_3^2 + 8135750592p_3p_5 - 8711658240p_4^2 + \\
& 1407612480p_3^2p_4 - 30436734528p_5^2)t_3^{30} + (-308962080p_3 + 15709314816p_5 - 7039348992p_3p_4 + \\
& 1847884416p_3^3 - 21546021888p_4p_5 - 4295003904p_3^2p_5 + 7496418816p_3p_4^2)t_3^{29} + \cdots
\end{aligned}$$

Le théorème suivant, dû à P. Gaudry [73], fournit une utilisation cette équation modulaire :

Théorème 30 *Soit $\mathcal{C}$ une courbe de genre 2 définie sur le corps fini $\mathbb{F}_p$, où p est un nombre premier différent de 3. Alors les motifs de factorisation de $\Xi_3(\mathcal{C})$ impliquent les valeurs de $\#\mathbf{Jac}(\mathcal{C}) \mod 3$ données dans les tableaux :*

$p \equiv 2 \mod 3$	
motif	$\#\mathbf{Jac}(\mathcal{C}) \mod 3$
$(10)^4$	2
$(2)^2(6)^6$	1
$(4)(12)^3$	1
$(2)^{20}$	1
$(1)^2(2)(4)(8)^4$	0
$(1)^2(2)(3)^2(6)^5$	0
$(1)^8(2)^{16}$	0
$(1)^5(2)^4(3)(6)^4$	0
$(4)^{10}$	1,2

$p \equiv 1 \mod 3$	
motif	$\#\mathbf{Jac}(\mathcal{C}) \mod 3$
$(5)^8$	1,2
$(1)(2)^2(3)(4)^2(12)^2$	0,2
$(1)^4(2)^2(4)^8$	0,2
$(1)(3)^4(9)^3$	0,1
$(1)^4(3)^{12}$	0,1
$(1)^{40}$	0,1
$(4)^{10}$	2
$(2)^2(6)^6$	1
$(2)^{20}$	1
$(1)^4(3)^{12}$	0
$(1)^2(2)(3)^2(6)^5$	0
$(1)^5(2)^4(3)(6)^4$	0
$(1)^8(2)^{16}$	0

Ainsi, étant donnée une courbe de genre 2 définie sur le corps fini $\mathbb{F}_p$, il suffit de factoriser un polynôme de degré 40 modulo p pour obtenir des renseignements sur le cardinal de sa Jacobienne modulo 3.

24.4.2 Formules d'addition

Pour décrire les points de ℓ-torsion, il faut expliciter la multiplication $[\ell]$. Cantor [42] propose des formulations de la loi de groupe dans la Jacobienne d'une courbe de genre 2. Notons essentiellement le cas particulier des multiples d'un même diviseur de poids 1. Dans ce cas, ses itérés sont donnés par les *polynômes de division de Cantor* [43], qui généralisent les polynômes de division vus en genre 1. Supposons la courbe donnée par l'équation $y^2 = x^5 + f_2x^3 + f_3x^2 + f_4x + f_4$. Alors nous avons les égalités

$$[\ell]\langle X - x_1, y_1\rangle = \langle X^2 + \delta_1^{(\ell)}(x_1)X + \delta_2^{(\ell)}, y_1\varepsilon_1^{(\ell)}(x_1)X + y_1\varepsilon_2^{(\ell)}(x_1)\rangle,$$

où $(\delta_1^{(\ell)},\delta_2^{(\ell)},\varepsilon_1^{(\ell)},\varepsilon_2^{(\ell)})$ sont des fractions rationnelles en x dont les coefficients dépendent des coefficients (p_3,p_4,p_5).

$$\delta_1^{(\ell)} = \frac{d_1^{(\ell)}}{d_0^{(\ell)}}, \qquad \delta_2^{(\ell)} = \frac{d_2^{(\ell)}}{d_0^{(\ell)}},$$
$$\varepsilon_1^{(\ell)} = \frac{e_1^{(\ell)}}{e_0^{(\ell)}}, \qquad \varepsilon_2^{(\ell)} = \frac{e_2^{(\ell)}}{e_0^{(\ell)}}.$$

Les polynômes $d_0^{(\ell)},d_1^{(\ell)},d_2^{(\ell)},e_0^{(\ell)},e_1^{(\ell)},e_2^{(\ell)}$ sont donnés par des formules de récurrence sur ℓ, que nous n'expliciterons pas. Leurs degrés sont donnés dans le tableau suivant :

Polynôme	$d_0^{(\ell)}$	$d_1^{(\ell)}$	$d_2^{(\ell)}$	$e_0^{(\ell)}$	$e_1^{(\ell)}$	$e_2^{(\ell)}$
Degré	$2\ell^2-1$	$2\ell^2-2$	$2\ell^2-3$	$3\ell^2-2$	$3\ell^2-2$	$3\ell^2-3$

24.4.3 Mise en équation pour $\ell = 3$

En genre 2, tout diviseur réduit peut s'écrire comme somme de deux points

$$D = \langle x^2 + u_1x + u_0, v_1x + v_0\rangle = P_1 + P_2,$$

où les points $(P_i)_{i=1,2}$ s'écrivent :

$$P_i = \langle X - x_i, y_i\rangle,$$

et l'égalité $D = P_1 + P_2$ implique en particulier les égalités

$$u_1 = -(x_1 + x_2), \quad u_0 = x_1x_2.$$

Le diviseur D est un diviseur de 3-torsion si et seulement si

$$\begin{aligned} [3]D = 0 \quad &\Longleftrightarrow \quad [3](P_1 + P_2) = 0 \\ &\Longleftrightarrow \quad [3]P_1 = -([3]P_2). \end{aligned}$$

Cette dernière égalité peut se réécrire en fonction de (x_1,x_2,y_1,y_2) à l'aide des formules de Cantor données ci-dessus. D'après [74], elle est équivalente à

$$\begin{aligned} x_1^5 + x_1^3 + p_3x_1^2 + p_4x_1 + p_5 &= y_1^2, \\ \delta_1^{(3)}(x_1) &= \delta_1^{(3)}(x_2), \\ \delta_2^{(3)}(x_1) &= \delta_2^{(3)}(x_2), \\ \varepsilon_1^{(3)}(x_1)y_1 &= -\varepsilon_1^{(3)}(x_2)y_2, \\ \varepsilon_2^{(3)}(x_1)y_1 &= -\varepsilon_2^{(3)}(x_2)y_2, \\ y_1y_2(x_1 - x_2) &\neq 0. \end{aligned}$$

Nous avons désormais de quoi mettre le problème en équations. On réécrit les formules ci-dessus en fonction des numérateurs et dénominateurs (e,d) des (δ,ε), pour obtenir un système en (x_1,x_2,y_1,y_2), dont il faut exclure les solutions satisfaisant $x_1 = x_2$ et $y_1y_2 = 0$. Suivant la présentation de la première section de ce chapitre, on introduit ensuite l'élément $u_1 = -(x_1 + x_2)$; enfin, dans le cas $\ell = 3$, l'élément t_3 de la définition 25 est défini par

$$t_3 = h_1(u_1),$$

et coïncide donc avec u_1. Le système est finalement :

$$\begin{array}{lrcl}
f_0 : & y_1^2 - (x_1^5 + x_1^3 + p_3x_1^2 + p_4x + 1 + p_5) & = & 0 \\
f_1 : & d_1^{(3)}(x_1)d_2^{(3)}(x_2) - d_1^{(3)}(x_2)d_2^{(3)}(x_1) & = & 0, \\
f_2 : & d_0^{(3)}(x_1)d_2^{(3)}(x_2) - d_0^{(3)}(x_2)d_2^{(3)}(x_1) & = & 0, \\
f_3 : & e_1^{(3)}(x_1)e_2^{(3)}(x_2)y_1 + e_1^{(3)}(x_2)e_2^{(3)}(x_1)y_2 & = & 0, \\
f_4 : & e_0^{(3)}(x_1)e_2^{(3)}(x_2)y_1 + e_0^{(3)}(x_2)e_2^{(3)}(x_1)y_2 & = & 0, \\
f_5 : & u_1 + x_1 + x_2 & = & 0 \\
g : & y_1y_2(x_1 - x_2) & \neq & 0.
\end{array}$$

Les caractéristiques complètes du système sont résumées dans l'annexe E.4. L'équation modulaire est le polynôme minimal de u_1 dans l'extension

$$\mathbb{Q}(p_3,p_4,p_5) \to \mathbb{Q}(p_3,p_4,p_5)[x_1,x_2,y_1,y_2,u_1,z]/(f_0,\ldots,f_5,1-gz).$$

24.4.4 Résolution pour $\ell = 3$

Stratégie employée. L'extension ci-dessus est de degré 160, et le polynôme minimal de u_1 est de degré 40. À chaque valeur de u_1 correspondent deux valeurs du couple (x_1,x_2), et pour chacune de ces valeurs, 2 valeurs de (y_1,y_2). Les solutions se présentent sous la forme d'un ensemble triangulaire :

$$\begin{cases}
T_1(y_1,y_2,x_1,x_2,u_2), \\
T_2(y_2,x_1,x_2,u_2), \\
T_3(x_1,x_2,u_2), \\
T_4(x_2,u_1), \\
\Xi_3 = T_5(u_1),
\end{cases}$$

où les polynômes (T_1,T_2,T_3,T_4,T_5) sont de degrés (1,2,1,2,40) en leur variable principale.

Le sous-système (f_1,f_2,f_5) ne dépend que des variables (x_1,x_2,u). L'une des composantes de ses zéros est la projection de la variété définie par le système pris en entier. En particulier, le polynôme minimal de u_1 dans les zéros simples du système (f_1,f_2,f_5) est un multiple du polynôme Ξ_3. Nous appliquons l'algorithme de la partie VII, en tirant profit de cette remarque. Les phases du calcul sont :

- le calcul des zéros du système (f_1,f_2,f_3,f_4,f_5), en-dehors de l'hypersurface $y_1y_2(x_1 - x_2) = 0$, pour une valeur aléatoire de (p_3,p_4,p_5) ;
- la remontée uniquement des trois derniers polynômes (T_3,T_4,T_5) en utilisant les équations (f_1,f_2,f_5).

Tout comme leurs analogues en genre 1, ces équations s'évaluent bien : elles nécessitent 50 multiplications, pour des degrés de 32. On peut comparer ces chiffres avec la mesure du système du chapitre 23 : nous avions alors 253 multiplications, pour un degré de 10.

Variables	Paramètres	Multiplications	Degrés	Hauteur
3	3	50	(32,32,1)	23

Validité et utilisation du résultat. Sur une machine Compaq XP/1000 - Alpha EV6, le calcul prend 4.25 h. Le calcul montre que ce polynôme de $\mathbb{Q}(p_3,p_4,p_5)[u_1]$ appartient en fait à $\mathbb{Z}[p_3,p_4,p_5][u_1]$, ce qui n'est pas évident a priori.

Le polynôme Ξ_3 a le degré générique en u_1 prévu par la théorie : il ne manque aucun point. Il est possible de lui appliquer un test probabiliste de vérification, dans l'esprit des solutions proposées dans le chapitre 15, cette solution n'a pas encore été mise en œuvre. Les divers calculs qui ont été effectués à partir de ce résultat n'ont pas révélé d'erreur.

Chapitre 25

Sur certaines courbes à Jacobienne décomposable

Résumé

Nous considérons dans ce chapitre des courbes $\mathcal{C}$ de genre 2 dont la Jacobienne est (2,2)-isogène à un produit de courbes elliptiques. Les j-invariants de ces courbes elliptiques sont fonction des modules (j_1,j_2,j_3) de $\mathcal{C}$: ils sont solutions d'un système polynomial, prenant comme paramètres (j_1,j_2,j_3).

Les courbes de ce type sont fréquemment étudiées [108, 135, 129, 146, 105, 130] ; déterminer explicitement les courbes elliptiques sous-jacentes est une question importante. La résolution au cas par cas du système mentionné ci-dessus est relativement aisée, mais il y a un intérêt à calculer une résolution paramétrée : celle-ci peut être intégrée dans des programmes ne permettant pas la résolution de systèmes polynomiaux.

Dans ce chapitre, nous proposons de calculer explicitement le polynôme minimal de ces j-invariants pour une valeur générique de (j_1,j_2,j_3). Les cas dégénérés sont traités in extenso dans l'article [75], écrit avec P. Gaudry, dont est issu ce chapitre. La résolution fait appel à l'algorithme de la partie VII. Nous proposons deux méthodes, l'une directe, l'autre introduisant des invariants qui permettent de simplifier le problème.

Sommaire

Présentation des résultats

Dans la section 22.3.3 du chapitre 22, nous avons défini les courbes de genre 2 dont la Jacobienne est (2,2)-décomposable ; nous allons nous y intéresser de plus près tout au long de ce chapitre. Ces courbes présentent un intérêt certain : les exemples des courbes ayant le plus grand rang ou la plus grande torsion connus [105] sont des courbes de ce type ; c'est également dans ce cadre que la méthode de Dem'janenko-Manin permet de déterminer tous les points rationnels d'une courbe [130].

Ca chapitre reprend l'essentiel de l'article [75], écrit en collaboration avec P. Gaudry du Laboratoire d'Informatique de l'École polytechnique (LIX). Nous y démontrons de manière constructive le théorème suivant.

Théorème 31 *Si $\mathcal{C}$ est une courbe de genre 2 à Jacobienne (2,2)-décomposable, il existe au plus deux courbes elliptiques quotients de degré 2 de* $\mathbf{Jac}(\mathcal{C})$.

Les j-invariants des quotients de degré 2 d'une courbe $\mathcal{C}$ à Jacobienne décomposable sont solutions d'un système d'équations paramétrées par les modules (j_1,j_2,j_3) de $\mathcal{C}$. Pour une courbe $\mathcal{C}$ donnée, à Jacobienne décomposable, ce système est de dimension zéro, et relativement “facile” à résoudre. Notre objectif est ici de fournir des formules de taille raisonnable qui donnent le polynôme minimal de ces j-invariants pour une courbe $\mathcal{C}$ à coefficients génériques. Ces formules peuvent être utilisées dans des environnements ne permettant pas la résolution du système de dimension zéro mentionné ci-dessus.

Les courbes à Jacobienne décomposable forment une sous-variété de dimension 2 de la variété des courbes de genre 2, notée $\mathcal{H}_2$ d'après Lange [135]. Pour déterminer les j-invariants des courbes quotientes, plusieurs cas sont distingués, selon que la courbe $\mathcal{C}$ de $\mathcal{H}_2$ est suffisamment “générique”, ou plus ou moins dégénérée. Dans ce chapitre, nous détaillons l'étude du cas générique, pour lequel les calculs sont les plus ardus, alors que l'article [75] présente l'ensemble des cas possibles. D'un point de vue calculatoire, les cas dégénérés sont plus faciles à traiter : les dégénérescences imposent des conditions sur les modules (j_1,j_2,j_3), et font passer le nombre de degrés de liberté de 2 à 1.

Dans la prochaine section, nous rappelons une classification des courbes à Jacobienne décomposable, amorcée par les travaux de Bolza à la fin du siècle dernier. Nous procédons ensuite à deux mises en équations du problème, que nous résolvons par les méthodes de la partie VII.

- La section 25.2 présente une approche directe du problème ; lors de la mise au point de ce travail, l'algorithme de la partie VII ne permettait pas de résoudre facilement le système qui en découle.
- Un supplément de réflexion a mené à une seconde mise en équations, faisant intervenir des invariants liés aux classes d'isomorphisme des courbes à Jacobienne décomposable. Les calculs sont plus simples, et la sortie de taille moindre. C'est le sujet de la dernière section de ce chapitre.

Les deux approches sont présentées : elles fournissent des systèmes différents, tous deux utiles pour étalonner nos algorithmes. En outre, les techniques de résolution varient d'un cas à l'autre.

Nous considérerons toujours les courbes $\mathcal{C}$ à isomorphisme près : les modules de $\mathcal{C}$ déterminent si celle-ci est décomposable, et si oui, déterminent les courbes elliptiques quotients. Dans toute la suite, les courbes sont définies sur un corps k de caractérstique différente de 2,3 et 5.

25.1 Préliminaires

Le lemme suivant, que l'on trouve en substance chez Igusa [108], caractérise les courbes à Jacobienne décomposable à partir de leurs automorphismes.

Proposition 75 *Soit $\mathcal{C}$ une courbe de genre 2. La Jacobienne de $\mathcal{C}$ est (2,2)-décomposable si et seulement si $\mathcal{C}$ admet une involution autre que l'involution hyperelliptique ι.*

Bolza [30], puis Igusa [108] et Lange [135] ont classifié les courbes admettant des automorphismes, en particulier les involutions. Les modules des courbes de genre 2 admettant une involution non hyperelliptique forment une sous-variété de dimension 2 de la variété des modules ; à la suite de Lange, nous la notons $\mathcal{H}_2$. Une équation de cette hypersurface est donnée par Mestre dans [158]. Avec les coordonnées (j_1, j_2, j_3) que nous avons choisies, elle s'écrit :

$$\begin{aligned}
&8393900389396594682757 12 j_3^2 + 9211413321697223245824 00000 j_3^3 \\
&+ 3298357634722313011 2000 j_1^2 j_3^2 + 182200942574622720 j_3 j_1 j_2^2 \\
&- 37481336758208102 4 j_3 j_1^2 j_2 + 99950231355221606400 00 j_3^2 j_1 j_2 \\
&+ 94143178827 j_2^4 - 562220051373121536 j_3 j_2^2 - 562220051373121536 j_3 j_1^3 \\
&+ 43381176803481600 j_3 j_2^3 - 71964166575759556608000 j_3^2 j_2 \\
&- 38860649950910160568 3200 j_3^2 j_1 - 1156831381426176 j_1^5 j_3 \\
&- 31381059609 j_1^7 + 62762119218 j_1^4 j_2^2 + 13947137604 j_1^3 j_2^3 \\
&- 31381059609 j_1 j_2^4 - 188286357654 j_1^3 j_2^2 - 6973568802 j_1^6 j_2 \\
&+ 192612425007458304 j_1^4 j_3 + 94143178827 j_1^6 - 6973568802 j_2^5 \\
&+ 28920784535654400 j_1^2 j_3 j_2^2 + 164848471853230080 j_1^3 j_3 j_2 = 0.
\end{aligned}$$

Appelons *groupe réduit d'automorphismes de $\mathcal{C}$* le quotient de son groupe d'automorphismes par $\{1, \iota\}$. On peut alors classifier les points de $\mathcal{H}_2$ par leur groupe réduit d'automorphismes G : Igusa [108] montre que l'on est nécessairement dans l'une des situations suivantes.

- Le groupe G est le groupe dihédral D_6, ce qui correspond au point de $\mathcal{H}_2$ associé à la courbe $y^2 = x^6 + 1$.
- Le groupe G est le groupe σ_4, ce qui correspond à la courbe $y^2 = x^5 - x$.
- Le groupe G est le groupe dihédral D_3. Les points correspondants forment une courbe $\mathcal{D}$ de $\mathcal{H}_2$ privée des 2 points précédents.
- Le groupe G est le groupe de Klein V_4. Les points forment une courbe $\mathcal{V}$ de $\mathcal{H}_2$ privée des 2 points précédents ; ces points sont les seules intersections de $\mathcal{D}$ et $\mathcal{V}$.
- Le groupe G est $\mathbb{Z}/2\mathbb{Z}$, ce qui correspond à un ouvert dense de $\mathcal{H}_2$ que nous baptisons Ω, formé de $\mathcal{H}_2$ privé des courbes précédentes. Dans ce cas, il y a exactement deux courbes elliptiques quotients de degré 2. Nous appellerons ce cas le cas *générique.*

Dans le dernier cas, l'énoncé du théorème 31 est immédiat, il ne reste qu'à expliciter les j-invariants des quotients de degré 2 de $\mathcal{C}$. C'est le sujet des prochaines sections de ce chapitre ; les autres cas sont traités dans [75].

25.2 Une première approche

25.2.1 Mise en équations

La proposition suivante est une autre caractérisation des courbes à Jacobienne décomposable ; on la trouve en substance dans [158].

Proposition 76 *Soit $\mathcal{C}$ une courbe de genre 2. La Jacobienne de $\mathcal{C}$ est décomposable si et seulement* *si il existe a et b tels que la courbe $\mathcal{C}$ est isomorphe à la courbe*

$$\mathcal{C}' : y^2 = x^6 + ax^4 + bx^2 + 1.$$

Dans ce cas, les courbes elliptiques quotients de la Jacobienne de $\mathcal{C}$ sont les courbes d'équations

$$\mathcal{E} : y^2 = x^3 + ax^2 + bx + 1.$$

Il est possible de calculer les modules (j_1,j_2,j_3) de la courbe $\mathcal{C}$ en fonction de a et b à partir de l'équation $\mathcal{C}'$, comme des fractions rationnelles $j_i = \mathcal{J}_i(a,b)$, $i = 1,\ldots,3$. Les formules sont données par exemple dans [158]. De même, on peut calculer le j-invariant de $\mathcal{E}$ à partir de a et b par une formule analogue à celle rappelée dans le chapitre 22. Mettant ces formules sous formes de polynômes, nous obtenons le système (f_1,f_2,f_3,f_4) ci-dessous, dont les variables sont (a,b,j) et les paramètres (j_1,j_2,j_3).

$$\begin{aligned}
f_1 \; : \;\; & -144j_1b^4a - 144j_1ba^4 - 8j_1a^5b^2 - 972j_1ba + 68j_1b^3a^3 + 270j_1a^2b^2 + 729j_1 + 16j_1b^6 + \\
& 216j_1a^3 + 216j_1b^3 + j_1a^4b^4 - 8j_1a^2b^5 + 16j_1a^6 - 1555200000 - 518400000ba - \\
& 69120000a^2b^2 - 4608000b^3a^3 - 153600a^4b^4 - 2048a^5b^5, \\
f_2 \; : \;\; & 729j_2 - 512a^5b^5 - 6144a^3b^6 + 41472a^4b^4 + 2350080b^3a^3 + 32486400a^2b^2 + 77760000ba \\
& -20736000b^3 - 20736000a^3 - 699840000 + 16j_2b^6 + 216j_2a^3 + 216j_2b^3 + 16j_2a^6 - 4147200b^4a \\
& -4147200ba^4 - 276480b^5a^2 - 276480b^2a^5 - 6144a^6b^3 - 144j_2ba^4 + j_2a^4b^4 - 8j_2a^5b^2 \\
& -972j_2ba + 68j_2b^3a^3 + 270j_2a^2b^2 - 8j_2a^2b^5 - 144j_2b^4a, \\
f_3 \; : \;\; & 729j_3 - 768a^5b^5 - 5120a^3b^6 - 9472a^4b^4 + 895488b^3a^3 + 19054080a^2b^2 + 33696000ba \\
& -691200b^3 - 691200a^3 - 1244160b^4a - 1244160ba^4 - 156672b^5a^2 - 156672b^2a^5 - 5120a^6b^3 \\
& -863136000 + 216j_3b^3 + 16j_3a^6 + 216j_3a^3 + 16j_3b^6 + j_3a^4b^4 - 8j_3a^2b^5 - 8j_3a^5b^2 + 270j_3a^2b^2 \\
& +68j_3b^3a^3 - 972j_3ba - 144j_3ba^4 - 144j_3b^4a, \\
f_4 \; : \;\; & 4ja^3 - ja^2b^2 + 4jb^3 + 27j - 18jab + 256a^6 - 2304ba^4 + 6912a^2b^2 - 6912b^3.
\end{aligned}$$

FIG. 25.1 – *Le système $j_1j_2j_3$*

Variables	Paramètres	Multiplications	Degrés
4	2	41	(10,10,10,6)

On peut interpréter ce système avec le vocabulaire du chapitre 21 : voyons (j_1,j_2,j_3) comme des paramètres, pour lesquels on souhaite résoudre le système surdéterminé ci-dessus en (a,b,j). Ce système n'est consistant que pour (j_1,j_2,j_3) appartenant à ce que nous avons baptisé une hypersurface de consistance. Dans ce cas précis, il s'agit de la surface $\mathcal{H}_2$ mentionnée dans l'introduction.

25.2.2 Résolution

Pour obtenir la variable j en fonction de (j_1,j_2,j_3), nous procédons comme dans le chapitre 21 : on relâche un paramètre, disons j_1, pour en faire une variable, de sorte à revenir à une situation génériquement intersection complète.

Pour une valeur générique de (j_3,j_2), le système ci-dessus n'admet qu'un nombre fini de valeurs (a,b,j,j_1) solutions de (f_1,f_2,f_3,f_4). En particulier, pour chaque valeur de j_1 solution, le dernier point des remarques préliminaires montre qu'il existe exactement deux valeurs de j solutions. Il en résulte que la solution générique du système (f_1,f_2,f_3,f_4) peut être décrite par un ensemble triangulaire à coefficients dans $\mathbb{Q}(j_3,j_2)$, de la forme

$$\begin{cases} t_1(a,b,j,j_1), \\ t_2(b,j,j_1), \\ t_3(j,j_1), \\ t_4(j_1), \end{cases}$$

où t_4 est l'équation de $\mathcal{H}_2$, et t_3 est le polynôme de degré 2 en j que l'on recherche.

Le calcul qui découle de cette mise en équations sort légèrement du cadre de la partie VII, où nous avons décidé de ne calculer que le polynôme minimal de la dernière variable en fonction des paramètres. Ici, nous allons un cran plus haut, en cherchant le polynôme minimal de j, en fonction des paramètres *et* de la variable j_1. L'ennui théorique est de ne pas connaître un initial par lequel multiplier notre sortie pour lui donner une "taille" raisonnable. Nous utilisons comme dénominateur la dérivée par rapport à j_1 du polynôme t_4, comme préconisé dans la section 18.3 du chapitre 18.

Sur une machine Compaq XP/1000 - Alpha EV6 , le calcul prend 140 s. Nous n'avons pas énoncé la probabilité d'erreur du calcul pour cette situation. Il faut se contenter de savoir que les paramètres qui auraient pu mener à l'échec sont tous enfermés dans des hypersurfaces algébriques.

À l'époque de la conception de l'article [75], l'algorithme de de remontée de Newton pour ensembles triangulaires n'était que partiellement au point, et nous avons dû trouver une approche parallèle, présentée dans la section suivante.

25.3 Les calculs revisités

Le premier pas consiste à partir d'une *forme de Rosenhain* de la courbe $\mathcal{C}$.

25.3.1 Le polynôme minimal à partir d'une forme de Rosenhain

Définition 26 (Forme de Rosenhain) *Soit $\mathcal{C}$ une courbe de genre 2. Il existe une courbe $\mathcal{C}'$ isomorphe à $\mathcal{C}$ donnée par l'équation*

$$\mathcal{C}' : y^2 = x(x-1)(x-\lambda)(x-\mu)(x-\nu),$$

où (λ,μ,ν) sont des nombres algébriques sur k, distincts de 0 et 1, et deux à deux distincts. Nous appellerons $\mathcal{C}'$ une forme de Rosenhain de $\mathcal{C}$

Démonstration. Mettons la courbe sous la forme $y^2 = P(x)$, où P est un polynôme de degré 6. Il suffit d'effectuer une homographie qui envoie trois des racines de P sur $(0,1,\infty)$ pour obtenir une forme de Rosenhain de $\mathcal{C}$. □

Revenons aux courbes à Jacobienne décomposable. Nous prenons comme point de départ [108] où l'on trouve la forme de Rosenhain d'une courbe (2,2)-décomposable.

Théorème 32 *Soit $\mathcal{C}$ une courbe à Jacobienne décomposable. Alors $\mathcal{C}$ est isomorphe à une courbe d'équation*

$$y^2 = x(x-1)(x-\lambda)(x-\mu)(x-\nu), \text{ où } \mu = \nu\frac{1-\lambda}{1-\nu},$$

avec λ, ν, μ deux à deux distincts et différents de 0 et 1. La Jacobienne de $\mathcal{C}$ est alors isogène au produit des courbes elliptiques $y^2 = x(x-1)(x-\Lambda)$, avec Λ donné par l'équation

$$\nu^2\lambda^2\Lambda^2 + 2\nu\mu(-2\nu+\lambda)\Lambda + \mu^2 = 0. \tag{25.1}$$

Démonstration. Supposons que $\mathcal{C}$ est sous forme de Rosenhain. Notons τ une involution non triviale sur $\mathcal{C}$. Celle-ci permute les points $(0,1,\infty,\lambda,\mu,\nu)$, et quitte à échanger les noms de λ, μ, ν, on a $\tau(0) = \lambda$, $\tau(1) = \mu$ et $\tau(\infty) = \nu$. D'autre part, d'après la section 22.2.2, τ est de la forme

$$\tau(x,y) = \left(\frac{ax+b}{cx+d}, \frac{wy}{(cx+d)^3}\right),$$

et, puisque c'est une involution, on a $a = -d$ et $w = \pm(ad-bc)^{3/2}$. L'involution τ peut ainsi être déterminée à l'aide des deux images $\tau(0) = \lambda$ et $\tau(\infty) = \nu$. Nous obtenons ainsi

$$\tau(x,y) = \left(\nu\frac{x-\lambda}{x-\nu}, \frac{u^3y}{(x-\nu)^3}\right), \text{ où } u = \pm\sqrt{\nu(\nu-\lambda)}.$$

La relation $\tau(1) = \mu$ fournit alors la première assertion

$$\mu = \nu\frac{1-\lambda}{1-\nu}.$$

L'étape suivante est de trouver une courbe isomorphe à $\mathcal{C}$ pour laquelle l'involution τ est $(x,y) \mapsto (-x,y)$. Pour cela on cherche de nouveau les coefficients d'une transformation du type $\varphi : x \mapsto (ax+b)/(cx+d)$ telle que $\varphi(0) = -\varphi(\lambda)$, $\varphi(1) = -\varphi(\mu)$, $\varphi(\infty) = -\varphi(\nu)$. On vérifie que la transformation

$$\varphi(x) = \frac{x-\nu-u}{x-\nu+u}, \text{ où } \quad u = \pm\sqrt{\nu(\nu-\lambda)}$$

convient. La courbe $\mathcal{C}$ est donc isomorphe à la courbe $\mathcal{C}''$ d'équation $y^2 = (x^2-x_1^2)(x^2-x_2^2)(x^2-x_3^2)$, où

$$x_1 = \varphi(\infty) = 1, \;\; x_2 = \varphi(0) = \frac{\nu-u}{\nu+u}, \;\; x_3 = \varphi(1) = \frac{1-(\nu-u)}{1-(\nu+u)}.$$

Par la suite, le morphisme $(x,y) \mapsto (x^2,y)$ envoie $\mathcal{C}''$ sur la courbe elliptique $\mathcal{E}$ d'équation

$$y^2 = (x-1)(x-x_2^2)(x-x_3^2),$$

elle-même isomorphe à la courbe d'équation $y^2 = x(x-1)(x-\Lambda)$ avec

$$\Lambda = \frac{x_2^2 - x_3^2}{1-x_3^2} = \frac{\mu}{\left(\nu \pm \sqrt{\nu(\nu-\lambda)}\right)^2}.$$

On obtient ainsi le résultat. Aucun dénominateur ne peut s'annuler, et la courbe $\mathcal{E}$ est non singulière d'après les conditions sur λ, μ, ν. □

Corollaire 12 *Soient $\mathcal{C}$ une courbe dont les modules appartiennent à l'ouvert Ω de $\mathcal{H}_2$, et λ, μ, ν définis comme dans le théorème précédent. Alors les j-invariants des courbes elliptiques quotients de la Jacobienne de $\mathcal{C}$ sont les solutions d'une équation*

$$j^2 + c_1(\lambda,\nu)j + c_0(\lambda,\nu) = 0, \tag{25.2}$$

où c_0 et c_1 sont des fractions rationnelles en deux variables.

Démonstration. Le j-invariant d'une courbe elliptique sous la forme $y^2 = x(x-1)(x-\Lambda)$ est donné par la relation

$$\Lambda^2(\Lambda-1)^2 j - 2^8(\Lambda^2-\Lambda+1)^3 = 0. \tag{25.3}$$

Le théorème précédent nous donne deux courbes elliptiques quotients de la Jacobienne de $\mathcal{C}$, et sur Ω ce sont les seules. Le polynôme attendu s'obtient ainsi par le calcul du résultant entre les équations 25.3 et 25.1 et en tenant compte de la relation $\mu = \nu\frac{1-\lambda}{1-\nu}$. □

Les valeurs de $c_0(\lambda,\nu)$ et $c_1(\lambda,\nu)$ sont données dans l'annexe E.6. De même, il est possible de calculer les modules (j_1,j_2,j_3) en fonction de λ et ν, et les formules sont données dans la même annexe. L'élimination de λ et ν va nous mener au résultat, via une étape intermédiaire.

25.3.2 Invariants des classes d'isomorphismes

Nous introduisons dans cette partie deux invariants caractéristiques des classes d'isomorphismes de courbes (2,2)-décomposables, qui permettent de décomposer le problème en sous-problèmes plus simples.

Théorème 33 *Soit $\mathcal{C}$ une courbe à Jacobienne décomposable. Alors il existe 24 triplets $(\lambda,\mu = \nu\frac{1-\lambda}{1-\nu},\nu)$ pour lesquels la courbe d'équation $y^2 = x(x-1)(x-\lambda)(x-\mu)(x-\nu)$ est isomorphe à $\mathcal{C}$.*

Démonstration. Le théorème 32 fournit un triplet (λ_1,μ_1,ν_1) satisfaisant les conditions requises. On peut donc supposer que $\mathcal{C}$ est la courbe correspondant à ce triplet. Toute courbe isomorphe à $\mathcal{C}$ est obtenue par une transformation birationnelle

$$x \mapsto \frac{ax+b}{cx+d}.$$

On exige de plus que la courbe soit mise sous forme de Rosenhain, donc la transformation doit envoyer 3 des 6 points $(0,1,\infty,\lambda_1,\mu_1,\nu_1)$ sur les points $(0,1,\infty)$. Le test exhaustif pour les 120 possibilités montre que seules 24 formes vérifient la condition supplémentaire liant les nouveaux λ, μ et ν.
Notons $(\lambda_i,\mu_i,\nu_i)_{i=1,\dots,24}$ les triplets solutions. Alors on passe d'un triplet à l'autre par des applications successives des transformations suivantes.

Transformation	$x \mapsto \frac{1}{x}$	$x \mapsto 1-x$	$x \mapsto \frac{x-\lambda_i}{1-\lambda_i}$	$x \mapsto \frac{x}{\mu_i}$
λ_i	$\frac{1}{\nu_i}$	$1-\mu_i$	$\frac{\lambda_i}{\lambda_i-1}$	$\frac{\lambda_i}{\mu_i}$
μ_i	$\frac{1}{\mu_i}$	$1-\lambda_i$	$\frac{\mu_i-\lambda_i}{1-\lambda_i}$	$\frac{1}{\mu_i}$
ν_i	$\frac{1}{\lambda_i}$	$1-\nu_i$	$\frac{\nu_i-\lambda_i}{1-\lambda_i}$	$\frac{\nu_i}{\mu_i}$

□

Les fonctions symétriques en les 24 triplets (λ_i,μ_i,ν_i) sont des invariants de la classe d'isomorphisme de $\mathcal{C}$; il est donc naturel d'y chercher deux invariants caractérisant ces classes.

Définition 27 *Soient $\mathcal{C}$ un courbe à Jacobienne décomposable, et (λ_i,μ_i,ν_i) les triplets définis précédemment. Nous appellerons Ψ et Υ les deux grandeurs :*

$$\begin{array}{rcl} \Psi & = & \sum_{i=1}^{24} \nu_i^2, \\ \Upsilon & = & \sum_{i=1}^{24} \lambda_i \nu_i, \end{array}$$

Ces grandeurs ne dépendent que de la classe d'isomorphisme de $\mathcal{C}$.

Pour l'instant, rien ne prouve que les invariants $\Psi(\lambda,\nu)$ et $\Upsilon(\lambda,\nu)$ sont bien choisis : ils peuvent être liés par une relation algébrique cachée, ou ne pas déterminer de manière univoque la classe d'isomorphisme (j_1,j_2,j_3). Le résultat des calculs ci-dessous va montrer qu'ils sont bien choisis.

25.3.3 Résolution

Résumons la situation : nous disposons des expressions suivantes.

$$\begin{array}{rcrcl} f_1 & : & \Psi & = & \Psi(\lambda,\nu), \\ f_2 & : & \Upsilon & = & \Upsilon(\lambda,\nu), \\ f_3 & : & c_0 & = & c_0(\lambda,\nu), \\ f_4 & : & c_1 & = & c_1(\lambda,\nu), \\ f_{4+i} & : & j_i & = & j_i(\lambda,\nu), \;\; i = 1,2,3; \end{array}$$

ces équations sont données dans l'annexe E.6. Le but est toujours d'obtenir (c_0,c_1) en fonction de (j_1,j_2,j_3). Pour cela nous allons d'abord écrire (c_0,c_1) en fonction de (Ψ,Υ), puis (Ψ,Υ) en fonction de (j_1,j_2,j_3).

Les coefficients c_0 et c_1 en fonction de (Ψ,Υ). La première étape est de déterminer c_0, puis c_1 en fonction de Ψ et Υ. Pour cela, nous résolvons successivement les systèmes $\mathcal{F}_1 = (f_1,f_2,f_3)$ et $\mathcal{F}_2 = (f_1,f_2,f_4)$, en mettant Ψ et Υ comme paramètres, en application de l'algorithme de la partie VII. Si les invariants (Ψ,Υ) sont bien choisis, ils déterminent uniquement la classe d'isomorphisme, donc c_0 et c_1 : le polynôme minimal de c_0 et c_1 doit être de degré 1 en (Ψ,Υ).
Pour détailler les calculs, disons que l'on s'intéresse à c_0. En sortie, nous attendons un ou des systèmes de la forme :

$$\left\{ \begin{array}{rcl} P_\nu(\nu,\lambda,c_0,\Psi,\Upsilon) & = & 0, \\ P_\lambda(\lambda,c_0,\Psi,\Upsilon) & = & 0, \\ P_{c_0}(c_0,\Psi,\Upsilon) & = & 0. \end{array} \right.$$

Nous pouvons résoudre ce système par les méthodes générales de la partie VII : nous avons donné le temps de résolution dans le chapitre 20. La suite présente un moyen d'accélérer ces calculs.

Le seul polynôme qui nous intéresse est le polynôme minimal de c_0, dont on soupçonne qu'il est de degré 1 en c_0. Nous pouvons donc tirer profit du caractère local de notre algorithme de remontée, pour éviter de calculer P_λ et P_ν, d'une manière similaire, en plus simple, aux calculs effectués dans la section 24.3 du chapitre 24.
Choisissons deux valeurs entières arbitraires (λ_0,ν_0) ; nous prenons comme spécialisation (Ψ_0,Υ_0) les valeurs $\Psi(\lambda_0,\nu_0)$ et $\Upsilon(\lambda_0,\nu_0)$. Dans ce cas, la résolution au-dessus de (Ψ_0,Υ_0) se factorise, et q'un des facteurs est de la forme :

$$\left\{ \begin{array}{rcl} \nu - \nu_0 & = & 0, \\ \lambda - \lambda_0 & = & 0, \\ P_{c_0}(c_0,\Psi_0,\Upsilon_0) & = & 0. \end{array} \right.$$

D'après la section 6.7.1 du chapitre 6, il suffit d'effectuer l'algorithme de remontée de la partie VII sur le facteur ci-dessus pour obtenir le polynôme minimal de c_0. Le calcul montre que (λ,ν) prennent 24 valeurs pour une valeur générique de (Ψ,Υ), et cette astuce permet de ne calculer qu'avec une seule d'entre elles.

Prenant en compte c_0 puis c_1, nous obtenons les résultats suivants (qui ont été factorisés après coup).

$$\begin{aligned} c_0 &= -4096\frac{\Upsilon^2(\Psi-32)^3}{\Psi^2(\Psi-8)}, \\ c_1 &= -128\Upsilon\frac{-\Psi^2+4\Psi\Upsilon-56\Psi+512}{\Psi(\Psi-8)}. \end{aligned}$$

Ces formules se vérifient en exprimant (c_0,c_1) ainsi que Ψ et Υ en fonction de λ et ν ; le temps de calcul est de 8 s. sur une machine Compaq XP/1000 - Alpha EV6.

Les j_i en fonction de (Ψ,Υ). Pour obtenir (Ψ,Υ) en fonction de (j_1,j_2,j_3), il est commode de passer par une étape intermédiaire, pour éliminer λ et ν : on détermine j_1, puis j_2 et j_3 à partir de (Ψ,Υ), en résolvant les systèmes $\mathcal{F}_3=(f_1,f_2,f_4)$, $\mathcal{F}_4=(f_1,f_2,f_5)$ puis $\mathcal{F}_5=(f_1,f_2,f_6)$. La stratégie est exactement la même que ci-dessus, on obtient :

$$\begin{aligned} j_1 &= -16\frac{(\Psi-8)^2(3\Psi-2\Upsilon)^5}{\Psi\Upsilon^4} \\ j_2 &= -64\frac{(\Psi-2)(\Psi-8)(3\Psi-2\Upsilon)^3}{\Psi\Upsilon^2} \\ j_3 &= -128\frac{(\Psi-8)(2\Psi^2-\Psi\Upsilon-13\Psi+3\Upsilon)(3\Psi-2\Upsilon)^2}{\Psi\Upsilon^2}. \end{aligned}$$

Le résultat de ce calcul se vérifie à nouveau en exprimant (j_1,j_2,j_3) ainsi que Ψ et Υ en fonction de λ et ν ; le temps de calcul est de 14 s. sur une machine Compaq XP/1000 - Alpha EV6.

(Ψ,Υ) en fonction de (j_1,j_2,j_3). Considérons les formules ci-dessus comme un système paramétré par (j_1,j_2,j_3), avec pour inconnues Ψ et Υ. C'est un système surdéterminé dépendant de paramètres ; pour faire le parallèle avec le chapitre 21, le lieu de consistance de ce système est la variété $\mathcal{H}_2$.

Nous procédons comme dans la section 25.2, en relâchant le paramètre j_1, pour en faire une variable, et en gardant uniquement (j_3,j_2) comme paramètres. Pour une valeur de (j_1,j_2,j_3) appartenant à $\mathcal{H}_2$, Ψ et Υ sont uniquement déterminés ; on peut donc appliquer l'algorithme de la partie VII. La résolution de ce système est moins immédiate que dans les deux cas ci-dessus, elle prend 40 s. sur une machine Compaq XP/1000 - Alpha EV6, et nous n'allons pas imprimer les résultats du calcul (Si ?).

Tout ceci nous permet de réécrire le polynôme minimal 25.2 sous la forme

$$j^2+c_1(j_1,j_2,j_3)j+c_0(j_1,j_2,j_3)=0,$$

où $c_1(j_1,j_2,j_3)$ et $c_0(j_1,j_2,j_3)$ sont des fractions rationnelles à trois indéterminées que l'on se garde bien de développer.

Remarque À l'époque de la conception de l'article [75], ces algorithmes n'étaient qu'à l'état d'ébauche. Nous avons obtenu les formules qui y figurent par un calcul d'interpolation de fractions rationnelles multivariées, qui ne s'est pas fait sans peine.

Exemple Soit $\mathcal{C}$ la courbe définie sur $\mathbb{Q}$ par l'équation

$$y^2 = x^6 - x^5 + x^4 - x^2 - x - 1.$$

Ses invariants d'Igusa sont

$$j_1 = \frac{2^3 \times 3^2 \times 5 \times 13}{37^2},\ j_2 = -\frac{2^3 \times 3^3 \times 11 \times 13}{37^3},\ j_3 = \frac{3^5 \times 53^2}{2^8 \times 37^5}.$$

On vérifie qu'ils annulent l'équation donnant $\mathcal{H}_2$, donc $\mathbf{Jac}(\mathcal{C})$ est isogène au produit de deux courbes elliptiques. Sur cet exemple, trouver ces courbes en passant par une forme de Rosenhain nécessiterait de passer dans une extension de $\mathbb{Q}$ de degré 24. Nos formules permettent de calculer directement

$$c_0 = \frac{2^{14} \times 5^6 \times 37^3}{53^2},\ c_1 = \frac{2^8 \times 3^4 \times 47}{53},$$

et les j-invariants des courbes elliptiques sont définis sur le corps $\mathbb{Q}(i)$ par

$$j = -\frac{2^7 \times 3^4 \times 47}{53} \pm \frac{2^8 \times 7 \times 11 \times 181}{53} i.$$

Dixième partie

Autres applications

Chapitre 26

Points réels sur une hypersurface singulière

Résumé

Ce chapitre présente une application de l'algorithme donné dans la partie VI à une phase de la résolution d'un problème de géométrie réelle : donner un point par composante connexe d'une hypersurface réelle.

L'article de F. Rouillier, M.-F. Roy et M. Safey el Din [187] propose une méthode basée sur un calcul de points critiques pour résoudre ce problème. Si l'hypersurface considérée possède une infinité de singularités, l'algorithme présenté dans [187] procède par *déformation infinitésimale*, et demande en particulier de résoudre un système dépendant du paramètre de déformation. L'algorithme de la partie VI permet de résoudre ce type de systèmes.

En collaboration avec M. Safey el Din, nous avons mis en œuvre ce calcul sur une famille d'exemples issue du *problème d'interpolation de Birkhoff* [88]. Il s'agit d'une des premières réalisations pratiques de ce type de calculs en géométrie réelle, qui a mené à une comparaison avec un algorithme proposé par P. Aubry, F. Rouillier et M. Safey el Din [14]. Cette comparaison est décrite dans [186], écrit avec F. Rouillier et M. Safey el Din.

Sommaire

26.1 Introduction

Ce chapitre se base sur l'article de F. Rouillier, M.F. Roy et M. Safey el Din [187], qui propose un algorithme de détection de toutes les composantes connexes d'une hypersurface réelle, reposant sur des méthodes de points critiques. Si l'hypersurface en question possède une infinité de singularités, cet algorithme demande de résoudre un système dépendant d'un paramètre de déformation ε. En collaboration avec M. Safey el Din de l'équipe Calfor du Laboratoire d'Informatique de Paris VI, nous avons mis en pratique cet algorithme, en le voyant dans le cadre des systèmes paramétrés.

Commençons en rappelant le principe de la méthode des points critiques proposée dans [187]. Soient f un polynôme sans carré de $\mathbb{Q}[x_1, \ldots, x_n]$, et $\mathcal{V}(f)$ l'hypersurface réelle définie par $f = 0$. Soient $\mathbf{A}$ un point de $\mathbb{R}^n$, et $d_{\mathbf{A}}$ la fonction “carré de la distance au point $\mathbf{A}$” :

$$\begin{array}{rcl} d_{\mathbf{A}} : \mathbb{R}^n & \to & \mathbb{R} \\ \mathbf{M} & \mapsto & d_{\mathbf{A}}(\mathbf{M}) = \|\mathbf{AM}\|^2. \end{array}$$

Cette fonction atteint un minimum sur chacune des composantes connexes de $\mathcal{V}(f)$, et pour une valeur générique du point $\mathbf{A}$, ces points critiques sont en nombre fini : leur détermination fournit au moins un point par composante connexe de $\mathcal{V}(f)$.

Ces points critiques sont inclus dans l'ensemble des zéros du système

$$f(\mathbf{M}) = 0, \;\; \mathbf{grad} f(\mathbf{M}) // \mathbf{AM} \tag{26.1}$$

où la dernière condition se traduit par l'annulation des $n - 1$ déterminants 2×2 de la forme $\mathbf{grad} f(\mathbf{M})_i \mathbf{AM}_{i+1} - \mathbf{grad} f(\mathbf{M})_{i+1} \mathbf{AM}_i$. Si l'hypersurface (complexe) $f = 0$ n'a qu'un nombre fini de singularités, ce système est de dimension zéro. Nous considérons que sa résolution et la détection des solutions réelles ne posent pas de difficultés ; voir notamment [188, 184] pour les questions d'isolation des racines réelles. Par contre, en présence d'un nombre infini de points singuliers, le système précedent admet un nombre infini de solutions, parmi lesquelles celles qui nous intéressent.

Un moyen de contourner cette difficulté est de procéder à une *déformation infinitésimale* de $\mathcal{V}(f)$: on l'inscrit dans la famille d'hypersurfaces définies par $f = \varepsilon$, où ε est un nouveau paramètre. Le théorème de Sard montre que ces hypersurfaces sont génériquement lisses. Plus précisément, dans [187], les auteurs montrent que pour une valeur générique de ε et du point $\mathbf{A}$, les solutions $(\lambda, x_1, \ldots, x_n)$ du système

$$f(\mathbf{M}) = \varepsilon, \;\; \lambda \, \mathbf{grad} f(\mathbf{M}) = \mathbf{AM}$$

sont simples, d'où l'on va déduire (proposition 77) que les solutions du système

$$f(\mathbf{M}) = \varepsilon, \;\; \mathbf{grad} f(\mathbf{M}) // \mathbf{AM} \tag{26.2}$$

sont également simples. En outre, l'article [187] décrit comment retrouver un point par composante connexe de $\mathcal{V}(f)$ à partir de la donnée d'une résolution de ce système paramétrée par ε : pour simplifier, il s'agit de déterminer le comportement quand ε tend vers 0 des point décrits par cette résolution, notamment de détecter les solutions bornées.

Dans ce chapitre, nous utilisons l'algorithme de la partie VI pour calculer la résolution en question. Une première comparaison entre cette implantation et celle de l'algorithme dû à P. Aubry, F. Rouillier et M. Safey el Din [14] se trouve dans [186], écrit en collaboration avec F. Rouillier et M. Safey el Din.

L'utilisation des méthodes de points critiques en géométrie réelle remonte au moins aux travaux de Grigoriev et Vorobjov [91] ; elles sont utilisées en particulier dans les travaux de Heintz, Roy et Solernó [101, 102], Renegar [179], Basu, Pollack et Roy [18, 19] et les articles du groupe TERA [16, 17] (voir également la thèse de G. Mbakop [155]). Ces derniers attaquent le problème de détection des composantes connexes d'une hypersurface réelle, puis d'une variété intersection complète et proposent des algorithmes basés sur des techniques d'évaluation. Ces outils sont analogues aux nôtres, mais les questions posées ne sont pas exactement les mêmes. Nous nous limitons ici au cas d'une hypersurface, et travaillons par déformation infinitésimale alors que les articles [16, 17] traitent de variétés algébriques non nécessairement hypersurfaces, et introduisent des *degrés réels* pour mesurer leurs algorithmes ; en contrepartie, ils travaillent sous les hypothèses d'une variété lisse, dont la partie réelle est compacte.

26.2 Mise en pratique

26.2.1 Mise en équations

Si f est un polynôme à coefficients dans $\mathbb{Q}[x_1,\dots,x_n]$ et $\mathbf{A} = (a_1,\dots,a_n)$ un point de $\mathbb{R}^n$, nous notons $\mathcal{F}(f)$ le système d'équations polynomiales en les coordonnées $(x_1,\dots,x_n)$ du point $\mathbf{M}$

$$\mathcal{F} : f(\mathbf{M}) = \varepsilon, \;\; \mathbf{grad} f(\mathbf{M}) // \mathbf{AM},$$

avec les conventions adoptées dans l'introduction :

$$\mathcal{F} = \begin{cases} f(x_1,\dots,x_n) - \varepsilon, \\ \frac{\partial f}{\partial x_1}(x_2 - a_2) - \frac{\partial f}{\partial x_2}(x_1 - a_1), \\ \vdots \\ \frac{\partial f}{\partial x_{n-1}}(x_n - a_n) - \frac{\partial f}{\partial x_n}(x_{n-1} - a_{n-1}). \end{cases}$$

Proposition 77 *Pour presque tout choix du point* $\mathbf{A}$ *et du paramètre de déformation* ε*, les solutions du système* $\mathcal{F}$ *sont simples.*

Démonstration. La première partie de la preuve est empruntée à [187]. Soit φ l'application

$$\begin{array}{ccc} \mathbb{C}^{n+1} & \to & \mathbb{C}^{n+1} \\ (x_1,\dots,x_n,\lambda) & \mapsto & (f(x_1,\dots,x_n), \lambda\frac{\partial f}{\partial x_1} + x_1, \lambda\frac{\partial f}{\partial x_n} + x_n) \end{array}$$

Notons $(\varepsilon,a_1,\dots,a_n)$ les coordonnées de l'espace d'arrivée ; on a $\varphi(x_1,\dots,x_n,\lambda) = (\varepsilon,a_1,\dots,a_n)$ si le gradient au point $(x_1,\dots,x_n)$ de l'hypersurface définie par $f = \varepsilon$ est colinéaire au vecteur $\mathbf{AM}$. En conséquence, d'après le théorème de Sard, les valeurs critiques de φ sont incluses dans une hypersurface de $\mathbb{C}^{n+1}$. Pour toute autre valeur de $(\varepsilon,\mathbf{A})$, les solutions du système $\varphi(x_1,\dots,x_n,\lambda) = (\varepsilon,\mathbf{A})$ sont simples.

Montrons que c'est aussi le cas pour le système $\mathcal{F}$. Tout d'abord, le lieu des zéros dans $\mathbb{C}^{n+1}$ du produit des dérivées partielles $\partial f/\partial x_i$ est de dimension n, donc son image par φ est de dimension au plus n. En conséquence, pour presque toute valeur de $(\varepsilon,\mathbf{A})$, les points de $\varphi^{-1}(\varepsilon,\mathbf{A})$ n'annulent aucune des composantes du gradient de f ni le déterminant jacobien du système φ, d'après le point précédent.

Les équations du système

$$f(\mathbf{M}) = \varepsilon, \;\; \mathbf{grad} f(\mathbf{M}) // \mathbf{AM}$$

s'obtiennent en éliminant λ à partir des équations du système $\varphi(x_1,\ldots,x_n,\lambda) = (\varepsilon,\mathbf{A})$:

$$\mathcal{F}_i = \frac{\partial f}{\partial x_i}(\varphi_{i+1} - a_{i+1}) - \frac{\partial f}{\partial x_{i+1}}(\varphi_i - a_i),$$

pour $i = 2,\ldots,n-1$. Soient $(\varepsilon,\mathbf{A})$ un point satisfaisant les conditions de généricité mentionnées ci-dessus et $(\mathbf{M}_j,\lambda_j)_{j\in\mathcal{J}}$ ses antécédents par φ. Au point $\mathbf{M}_j$, le gradient de l'équation $\mathcal{F}_i$ est égal à

$$\frac{\partial f}{\partial x_i}\mathbf{grad}\varphi_{i+1} - \frac{\partial f}{\partial x_{i+1}}\mathbf{grad}\varphi_i,$$

pour $i = 2,\ldots,n-1$. Les combinaisons linéaires correspondantes mettent la jacobienne de φ sous la forme

$$\begin{bmatrix} \mathbf{Jac}(\mathcal{F}) & \mathbf{0} \\ \cdots & \frac{\partial f}{\partial x_n} \end{bmatrix}$$

et préservent la propriété de non-annulation du déterminant. En conséquence, les racines $\mathbf{M}_j$ du système $\mathcal{F}$ sont simples, ce que l'on voulait prouver. □

En conséquence, pour presque tout choix du point $\mathbf{A}$, les racines du système $\mathcal{F}(f)$ sont génériquement simples, et l'algorithme du chapitre 10 permet d'en calculer une résolution paramétrée par ε.

26.2.2 Exemples

Nous avons mis en œuvre le calcul de la résolution paramétrée sur une séries d'exemples tirés du problème d'*interpolation de Birkhoff* [88]. Le problème de Birkhoff est un problème de décision : on veut savoir si un schéma d'interpolation donné peut toujours être satisfait.

Question : *Soient n et r deux entiers et $\mathcal{I}$ un sous-ensemble de $\{1,\ldots,n\}\times\{1,\ldots,r\}$. Décider si, pour tous réels $(f_{i,j})_{(i,j)\in\mathcal{I}}$ et tous réels distincts $x_1 < \ldots < x_n$, il existe un polynôme univarié P de $\mathbb{R}[x]$ de degré inférieur à r vérifiant*

$$\forall (i,j)\in\mathcal{I},\ P^{(j)}(x_i) = f_{i,j}.$$

Cette question est abordée par L. Gonzalez-Vega dans [88], puis F. Rouillier [184], qui effectuent de nombreuses réductions du problème. En particulier, ils montrent que l'on peut ramener le problème à décider si une hypersurface contient des points réels à coordonnées toutes non nulles.

Avec M. Safey el Din, nous nous sommes intéressés au cas $n = 5$, $r = 4$, pour lequel tous les choix possibles de l'ensemble d'indices $\mathcal{I}$ nécessitent d'étudier 53130 hypersurfaces de $\mathbb{R}^3$; seule une centaine d'entre elles nécessitent l'emploi de techniques de calcul avec infinitésimaux. Nous en avons retenu 10, notées Birkhoff3-1,...,Birkhoff3-10, en des variables notées (t_2,t_3,t_4) :

Birkhoff3-1 : $1 + 2t_2^2 - 2t_3^2 - 2t_4^2 + t_2^4 - 2t_2^2t_3^2 - 2t_2^2t_4^2 + 3t_3^4 + 6t_3^2t_4^2 + 3t_4^4$

Birkhoff3-2 : $t_2^4 + 2t_2^2t_3^2 - t_2^2t_4^2 - t_3^2t_4^2 + t_3^4 + t_4^4$

Birkhoff3-3 : $3t_3^6 + 3t_4^6 + 9t_3^2t_4^4 + 9t_3^4t_4^2 + t_2^2t_4^4 + t_2^2t_3^4 - t_2^4t_4^2 - t_2^4t_3^2 +$
$t_2^6 + 4t_4^4 + 4t_3^4 + 4t_2^4 + 2t_2^2t_3^2t_4^2 + 8t_3^2t_4^2 - 4t_2^2t_4^2 - 4t_2^2t_3^2$

Birkhoff3-4 : $-3t_3^6 - t_4^6 - 3t_3^2t_4^4 + 3t_3^4t_4^2 - 3t_2^2t_4^4 - 9t_2^2t_3^4 + 3t_2^4t_4^2 - 9t_2^4t_3^2 - 3 - 9t_2^2 - 9t_3^2 - 9t_2^4 - 9t_3^4 - 18t_2^2t_3^2 + 6t_2^2t_4^2 + 6t_3^2t_4^2 - 3t_2^6 + 3t_4^2 - 3t_4^4 + 6t_2^2t_3^2t_4^2$

Birkhoff3-5 : $2t_2^2t_3^2t_4^2 - t_2^8 + 2t_2^6 - t_3^4 - 3t_2^4 + t_2^2t_4^4 + 3t_2^4t_4^2 - t_4^4 - 3t_2^2$ $t_4^2 - 2t_3^2t_4^2 - 2t_2^6t_4^2 - 2t_2^6t_3^2 - t_2^4t_4^4 - t_2^4t_3^4 + t_2^2t_3^4 + 3t_2^4t_3^2 - 3t_2^2t_3^2 - 2t_2^4t_3^2t_4^2$

Birkhoff3-6 : $-6t_2^2t_3^2t_4^2 - t_2^8 - 2t_2^6 - t_2^4 + 2t_2^6t_4^2 + 2t_2^6t_3^2 - t_4^4 - t_3^4 - 3t_2^4t_4^4 - 3t_2^4t_3^4 - 3t_2^2t_4^4 - 3t_2^2t_3^4 - 6t_2^4t_3^2t_4^2 + 3t_2^4t_4^2 + 3t_2^4t_3^2 - 2t_3^2t_4^2 + t_2^2t_4^2 + t_2^2t_3^2$

Birkhoff3-7 : $-t_2^4 - t_3^4 - t_4^4 - 3t_2^2t_4^4 - 6t_2^2t_3^4 + 3t_2^4t_4^2 - 6t_2^4t_3^2 - t_2^8 - 2t_3^6 - 2t_2^6 - t_3^8 - 2t_2^2t_3^2 + t_2^2t_4^2 + t_3^2t_4^2 - 3t_3^2t_4^4 + 3t_3^4t_4^2 + 6t_2^2t_3^2t_4^2 + 6t_2^4t_3^2t_4^2 - 6t_2^2t_3^2t_4^4 + 6t_2^2t_3^4t_4^2 + 2t_2^6t_4^2 - 4t_2^6t_3^2 - 3t_2^4t_4^4 - 6t_2^4t_3^4 - 3t_3^4t_4^4 + 2t_3^6t_4^2 - 4t_2^2t_3^6$

Birkhoff3-8 : $t_3^6 + 3t_4^6 + t_3^2t_4^4 - t_3^4t_4^2 + 4t_2^2t_4^4 + 4t_2^2t_3^4 - 4t_2^2t_3^2t_4^2 + 4t_3^4 + 4t_4^4 - 4t_3^2t_4^2$

Birkhoff3-9 : $t_3^6 + t_4^6 + 2t_3^4 + t_3^2t_4^4 - t_3^4t_4^2 + 2t_2^2t_4^4 + 2t_2^2t_3^4 + 2t_4^4 - 2t_3^2t_4^2 - 2t_2^2t_3^2t_4^2$

Birkhoff3-10 : $-t_3^6 - 3 - 4t_3^4t_4^2 + t_2^2t_3^4 - 4t_2^4t_4^2 - t_2^4t_3^2 - 3t_2^6 + t_3^4 - 9t_2^4 - t_3^2 - 9t_2^2 - 4t_4^2 + 4t_2^2t_3^2t_4^2 - 2t_2^2t_3^2 - 8t_2^2t_4^2 + 4t_3^2t_4^2$

Nous avons résolu les systèmes $\mathcal{F}(f)(\mathbf{A},\varepsilon,t_2,t_3,t_4)$, où f décrit l'ensemble des polynômes donnés ci-dessus, et $\mathbf{A}$ est un point à coordonnées entières tiré au hasard — pour l'instant, nous supposons que ce point satisfait les conditions de généricité mentionnées plus haut. Le tableau 26.2.2 donne le temps du calcul de la résolution paramétrée par ε ; l'annexe E.7 donne les calculs d'évaluations codant ces systèmes, pour les hypersurfaces Birkhoff3-3 et Birkhoff3-10.

FIG. 26.1 – *Points réels : problème de Birkhoff3-3*

Variables	Paramètres	Multiplications	Degrés
3	1	90	(6,6,5)

Il convient de voir les résultats du tableau précédent comme un premier élément permettant d'estimer le coût pratique d'une méthode reposant sur une déformation infinitésimale. À ce stade, l'algorithme est toujours probabiliste ; par ailleurs, pour répondre pleinement à la question de Birkhoff, il reste à effectuer la phase de détection de composantes connexes à partir de cette résolution paramétrée puis la discussion sur l'existence de points réels en-dehors des plans de coordonnées — qui se ramène à l'étude d'une nouvelle hypersurface.

FIG. 26.2 – *Calcul d'une résolution dépendant de ε*

Hypersurface	Degré en t	Degré en ε	Hauteur	Remontée
Birk.3-1	16	3	23	3 s.
Birk.3-2	16	3	11	3 s.
Birk.3-3	34	5	56	18 s.
Birk.3-4	36	5	22	18 s.
Birk.3-5	40	7	60	23 s.
Birk.3-6	52	7	75	37 s.
Birk.3-7	52	7	28	42 s.
Birk.3-8	130	19	94	450 s.
Birk.3-9	132	19	71	464 s.
Birk.3-10	136	19	123	935 s.

Au prix de calculs supplémentaires, il est possible de supprimer l'aspect probabiliste de l'ensemble de ces calculs, quitte donc à perdre en partie le contrôle sur la complexité du processus. Voici un bref aperçu des calculs à effectuer.

– *Choix du point* **A**. Le point **A** doit être tel que toutes les racines du système $\mathcal{F}(f)(\mathbf{A},\varepsilon,\mathbf{x})$ sont simples. L'article [187] propose une solution de type "essai et erreur" à cette question, qui repose sur le calcul d'une base Gröbner pour le système $\mathcal{F}(f)(\mathbf{A},\varepsilon,\mathbf{x})$ dans $\mathbb{Q}\langle\varepsilon\rangle[x_1,\ldots,x_n]$, qui est ramené à un calcul dans $\mathbb{Q}[\varepsilon,x_1,\ldots,x_n]$.
– *Choix de la valeur de spécialisation.* L'algorithme de la partie VI fournit une résolution candidat, paramétrée par ε, dont il faut assurer qu'elle a le bon nombre de solutions génériques. Cela se traduit pas une condition sur la valeur de spécialisation du paramètre ε : il faut assurer que les racines du système spécialisé sont en nombre fini, égal au nombre générique de solutions. Ce nombre générique est déterminé à partir de la base de Gröbner calculée au point précédent.
– *Vérification.* Enfin, il faut assurer que la solution proposée est la bonne, c'est-à-dire que la phase de remontée n'a pas été arrêtée trop tot. Il suffit de vérifier que les polynômes formant cette résolution se réduisent à zéro modulo la base de Gröbner précédemment calculée.

  Une seconde approche est de le vérifier en suffisamment de valeur du paramètre ε. La dernière section du chapitre 10 propose un tel test, et indique le nombre de vérifications à effectuer, qui est contrôlé par un nombre de Bézout.

Tous ces points sont détaillés dans l'article en préparation [186]. Les résultats de l'étude du problème de Birkhoff que nous avons entreprise s'y trouvent également.

Chapitre 27

Calculs de relations

Résumé

Ce chapitre décrit une application de l'algorithme de la partie VII au calcul de relations entre invariants associés à des polynômes homogènes sous l'actions du groupe linéaire ; la résolution de ce système a été obtenue en collaboration avec G. Lecerf.

Cette question a été transmise par I. Bershenko, de l'Université du Minnesota : il s'agit d'un exemple de mise en œuvre des méthodes qu'elle décrit dans [25], et dans [26], en collaboration avec P.J. Olver. Ce domaine est certainement porteur d'autres applications de nos méthodes de calcul.

Sommaire

27.1 Introduction

Le cadre de ce bref chapitre est l'étude des symétries et de l'équivalence de polynômes homogènes sous l'action d'un groupe linéaire. Nous reprenons les définitions données dans la thèse d'I. Bershenko [25].

Définition 28 *Soient F et G deux polynômes de $\mathbb{C}[x_1,\ldots,x_n]$. Les polynômes F et G sont dits* équivalents *s'il existe une matrice $\mathbf{A}$ de $\mathrm{GL}(n,\mathbb{C})$ telle que $G(\mathbf{x}) = F(\mathbf{A}\mathbf{x})$.*

Dans [25], la question de la *classification* des polynômes pour la relation d'équivalence ci-dessus est ramenée à la détermination et à la classification d'invariants différentiels définis sur les graphes $y = F(x_1,\ldots,x_n)$ inclus dans $\mathbb{C}^{n+1}$, selon la méthode du repère mobile de Cartan [44]. La classification se ramène *in fine* à déterminer les relations algébriques entre différents invariants, qui sont donnés explicitement comme des fractions rationnelles des variables $(x_1,\ldots,x_n)$.

Nous ne donnons pas plus de détails, le travail théorique est décrit dans [25, 26]. Avec G. Lecerf, en réponse à une question d'I. Bershenko, nous avons achevé l'étude d'un exemple.

27.2 Résolution d'un exemple

La question qui nous a été posée concerne la détermination de la classe d'équivalence du polynôme $P^n + Q^n + R^n$. Après quelques simplifications, la question fut ramenée à déterminer la relation liant les invariants ci-dessous, construits à partir de la partie affine $p^n + q^n + 1$:

$$
\begin{aligned}
x &= \frac{1}{3}\frac{pq^2 + q^2 + q + p^2q - 6pq + p + p^2}{pq},\\
y &= -\frac{1}{9}\frac{(q-1+p)(-q-1+p)(-q+1+p)}{pq},\\
z &= \frac{1}{36}\frac{(p+1)^2(q+1)^2(q+2q^2+q^3+p+pq^3+2p^2-p^2q^2+p^3+qp^3)^2}{p^2q^2(p+q+1)^4}
\end{aligned}
$$

Autrement dit, nous cherchons une équation de l'image de l'application $\varphi : (p,q) \mapsto (x,y,z)(p,q)$. Cette image est irréductible ; en conséquence, si nous obtenons une équation irréductible candidat, il suffit de vérifier que les paramétrisations donnant (x,y,z) s'y annulent pour savoir que c'est la bonne.

FIG. 27.1 – *Système bershenko*

Variables	Paramètres	Multiplications	Degrés
3	2	32	(3,4,12)

Nous avons déja rencontré cette situation dans le chapitre 25, et nous allons suivre une démarche de résolution analogue. Nous mettons y et z dans le corps des paramètres ; la relation que l'on recherche est le numérateur du polynôme minimal de x, qui a ses coefficients dans $\mathbb{Q}(y,z)$. En appliquant l'algorithme de la partie VII, nous obtenons la relation :

$$\begin{aligned}
&x^{12}-6x^{11}y+9x^{10}y^2+76/3x^{11}-144x^{10}y+204x^9y^2-8x^{10}z+48x^9yz-\\
&72x^8y^2z+2642/9x^{10}-4696/3x^9y+2070x^8y^2-176x^9z+984x^8yz-1344x^7y^2z-\\
&72x^6y^3z+16x^8z^2-96x^7yz^2+144x^6y^2z^2+18520/9x^9-91556/9x^8y+37132/3x^7y^2-\\
&16744/9x^8z+27976/3x^7yz-11380x^6y^2z-912x^5y^3z+896/3x^7z^2-1632x^6yz^2+\\
&2112x^5y^2z^2+288x^4y^3z^2+787267/81x^8-395216/9x^7y+434465/9x^6y^2-109136/9x^7z+\\
&475840/9x^6yz-169552/3x^5y^2z-4744x^4y^3z+30400/9x^6z^2-44320/3x^5yz^2+16048x^4y^2z^2+\\
&2112x^3y^3z^2+144x^2y^4z^2+7918184/243x^7-10706368/81x^6y+1154296/9x^5y^2-\\
&4257280/81x^6z+5256904/27x^5yz-1601548/9x^4y^2z-38848/3x^3y^3z+630400/27x^5z^2-\\
&713824/9x^4yz^2+70016x^3y^2z^2+6912x^2y^3z^2+384xy^4z^2-16384/9x^4z^3+16384/3x^3yz^3-\\
&6144x^2y^2z^3+3072xy^3z^3-576y^4z^3+57952474/729x^6-68766490/243x^5y+19030160/81x^4y^2-\\
&37717136/243x^5z+38633968/81x^4yz-9714928/27x^3y^2z-175744/9x^2y^3z+\\
&2586400/27x^4z^2-6764896/27x^3yz^2+515344/3x^2y^2z^2+35168/3xy^3z^2+256y^4z^2-\\
&458752/27x^3z^3+114688/3x^2yz^3-28672xy^2z^3+7168y^3z^3+310857668/2187x^5-\\
&104667448/243x^4y+71153152/243x^3y^2-8438072/27x^4z+62399648/81x^3yz-\\
&36645440/81x^2y^2z-415744/27xy^3z+56743040/243x^3z^2-36715552/81x^2yz^2+\\
&220608xy^2z^2+71552/9y^3z^2-1605632/27x^2z^3+802816/9xyz^3-100352/3y^2z^3+\\
&1212061129/6561x^4-998159360/2187x^3y+57702400/243x^2y^2-920822576/2187x^3z+\\
&573501304/729x^2yz-26046464/81xy^2z-401408/81y^3z+80515136/243x^2z^2-\\
&105097568/243xyz^2+3135536/27y^2z^2-22478848/243xz^3+5619712/81yz^3+\\
&371879456/2187x^3-2101439488/6561x^2y+247267328/2187xy^2-2378255080/6561x^2z+\\
&335141504/729xyz-23965696/243y^2z+551342464/2187xz^2-122325280/729yz^2-\\
&39337984/729z^3+6204370304/59049x^2-877764608/6561xy+157351936/6561y^2-\\
&3547630208/19683xz+256231936/2187yz+520993552/6561z^2+6920243200/177147x-\\
&1485209600/59049y-2328984064/59049z+3504640000/531441=0
\end{aligned}$$

Cet exemple est d'une taille modérée, et rentre bien dans notre cadre : on ne recherche qu'une seule relation, que l'on considère comme un polynôme minimal. Davantage de relations s'interprêteraient comme autant de polynômes d'un ensemble triangulaire. Sur une machine Compaq XP/1000 - Alpha EV6, le calcul ne demande qu'un vingtaine de secondes ; dans le chapitre 20, nous donnons le temps pris par un calcul de base de Gröbner, pour un ordre délimination.

Annexe A

Index

Représentation des zéros, chapitre 1	
$\mathcal{R}(t) = (t,q(t),\mathbf{v}(t))$	représentation de Kronecker d'élément primitif t de polynôme minimal $q(t)$ et de paramétrisations $q'(t)x_i = v_i(t)$
$\mathcal{R}(t) = (t,q(t),\mathbf{w}(t))$	représentation "Shape Lemma" d'élément primitif t de polynôme minimal $q(t)$ et de paramétrisations $x_i = w_i(t)$
$\mathbf{t} = (t_1,\ldots,t_n)$	ensemble triangulaire

Notations de complexité, chapitre 5	
$O_{\log}(X)$	$O(X)$ à des facteurs logarithmiques près
$\mathcal{T}(d,m) = \binom{d+m}{m}$	Nombre de monômes de degré inférieur à d en m variables
$\mathcal{M}_{\mathbb{N}}(d)$	$(+,\times)$ sur les entiers à d chiffres
$\mathcal{M}(d)$	$(+,\times)$ sur les polynômes univariés de degré d
$\mathcal{M}_m(m,d)$	$(+,\times)$ sur les polynômes en m variables de degré d
$\mathcal{M}_s(m,d)$	$(+,\times)$ sur les séries en m variables à précision d
$\mathcal{M}(d_1,\ldots,d_n)$	$(+,\times)$ modulo un ensemble triangulaire de degrés $(d_1,\ldots,d_n)$
n^{Ω}	$(+,\times,\div)$ sur les matrices $n \times n$

Géométrie	
$\overline{k}$	une clôture algébrique de k
$\mathbb{A}^n(k)$	espace affine à n dimensions sur k
$\mathcal{V}(f_1,\ldots,f_n)$	lieu des zéros de $(f_1,\ldots,f_n)$

Notations de degré, chapitres 7 et 8	
$\mathbf{x} = (x_1,\ldots,x_n)$	inconnues
$\mathbf{p} = (p_1,\ldots,p_m)$	paramètres
$\mathbf{Jac}(\mathbf{f})$	matrice jacobienne de $\mathbf{f}$ par rapport aux variables $\mathbf{x}$
$\mathbf{jac}(\mathbf{f})$	déterminant jacobien de $\mathbf{f}$ par rapport aux variables $\mathbf{x}$

$\mathcal{U}$	ensemble des composantes simples de $\mathcal{V}(f_1,\ldots,f_n)$
$d_{\mathbf{x}}$	maximum des degrés partiels de $(f_1,\ldots,f_n)$ en $\mathbf{x}$
$d_{\mathbf{p},\mathbf{x}}$	maximum des degrés de $(f_1,\ldots,f_n)$
$\mathcal{D}_{\mathbf{p},\mathbf{x}}$	degré de $\mathcal{U}$
$\mathcal{D}_{\mathbf{x}}$	nombre générique de solutions simples de $\mathbf{f}$
$\mathcal{R}(t)$	résolution générique pour l'élément séparant t
$\mathcal{D}(t)$	maximum des degrés en les paramètres des coefficients de $\mathcal{R}(t)$ (page 75 et suivantes)
m_g	polynôme minimal de g dans $\mathcal{U}$, à coefficients dans $k(p_1,\ldots,p_m)$ (page 82)

Notations de hauteur, chapitres 12 et 14

h	maximum des hauteurs des polynômes $(f_1,\ldots,f_n)$
$\mathcal{H}(t)$	hauteur de la résolution $\mathcal{R}(t)$
$\mathcal{H}(h,h_t,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}})$	majorant de $\mathcal{H}(t)$
$e_{\mathrm{Red}}(\mathbf{p},t)$	entier discriminant
$\mathcal{H}_{\mathrm{Red}}(h,h_t,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}},M)$	borne sur la hauteur du précédent (page 122)
$e_{\mathrm{Padé}}(\mathbf{p},t,\gamma)$	entier contrôlant l'exécution modulaire de la reconstruction d'une fraction rationnelle
$\mathcal{H}_{\mathrm{Padé}}(h,h_t,n,m,d_{\mathbf{x}},d_{\mathbf{p},\mathbf{x}},M)$	borne sur la hauteur du précédent (page 126)

Systèmes surdéterminés, chapitre 21

Δ	hypersurface de consistance (page 184)

Calculs dans les courbes, chapitres 23, 25 et 24

$\mathcal{E}$	courbe elliptique
j	j-invariant de $\mathcal{E}$ (page 199)
$\mathcal{C}$	courbe hyperelliptique
$\mathbf{Jac}(\mathcal{C})$	Jacobienne de $\mathcal{C}$
(j_1,j_2,j_3)	modules de $\mathcal{C}$ (page 199)

Annexe B

Calculs d'évaluation

Les algorithmes des parties centrales de ce texte reposent sur la donnée d'une entrée codée par des *calculs d'évaluation*, ou "*Straight-Line Programs*".

Définition 29 (Calcul d'évaluation) *Un calcul d'évaluation, dépendant d'un certain nombre de variables d'entrée, est une suite d'instructions de la forme*

$$a_i \leftarrow b_i \circ c_i$$

où
- $\circ$ *est l'une des opérations* $\{+, -, \times\}$,
- b_i *et* c_i *sont des constantes, des variables d'entrée ou des variables définies lors d'une étape précédente.*

La taille d'un calcul d'évaluation est le nombre d'étapes de calcul ; elle est notée L *dans ce mémoire.*

Des définitions plus précises se trouvent dans [113, 99, 37, 81].

Exemple Le calcul suivant prend comme variable d'entrée x et évalue le polynôme $(x+1)^{32}$

```
t0:=x+1;
t1:=t0*t0;
t2:=t1*t1;
t3:=t2*t2;
t4:=t3*t3;
t5:=t4*t4;
```

De nombreuses références abordent l'algorithmique des calculs d'évaluations, le problème du test à zéro et de la dérivation [103, 20, 107, 113, 99, 123, 94]. Les travaux du groupe TERA [82, 123, 81, 83, 100] menant à la conception de l'algorithme de résolution gémétrique montrent que l'utilisation de cette structure de donnée dans les algorithmes d'élimination permet d'éviter une complexité d'ordre $d^{O(n^2)}$ due à la représentation des objets éliminants sur la base monomiale. Dans ce mémoire, nous utilisons cette structure de données en conjonction avec l'opérateur de Newton formel du chapitre 6. Les sorties de nos algorithmes sont développées sur la base monomiale, refletant l'implantation qui en est faite. Pour cette représentation, nos résultats sont proches de l'optimal.

Pour conclure, notons que les systèmes donnés dans les annexes suivantes témoignent souvent d'une "bonne" complexité d'évaluation, par exemple par rapport à leur nombre de monômes une fois développés.

Annexe C

Retour sur quelques points d'algorithmique

C.1 Multiplication rapide en caractéristique zéro

Cette section présente deux résultats de multiplication rapide de polynômes et de séries multivariés en caractéristique zéro, basés sur un procédé d'évaluation et d'interpolation multivariée dû à Canny, Kaltofen et Lakshman [41]. Ben Or et Tiwari [23], Kaltofen et Lakshman [114] et Zippel [212] abordent également le sujet de l'interpolation multivariée, dans le contexte des polynômes creux.

Proposition 78 [41, Théorème 1] *Soient k un corps de caractéristique zéro et f et g deux polynômes de $k[p_1,\ldots,p_m]$ de degrés bornés par d. On peut calculer le produit fg en*

$$O\big(\mathcal{M}(\mathcal{T}(2d,m))\log\mathcal{T}(2d,m)\big)$$

opérations de k.

Le résultat analogue pour les séries multivariées est dû à G. Lecerf [141].

Proposition 79 *Soient k un corps de caractéristique zéro et f et g deux séries de $k[[p_1,\ldots,p_m]]$ de précision d. On peut calculer le produit fg à précision d en*

$$O\big(\mathcal{M}(d)\mathcal{M}(\mathcal{T}(d,m-1))\log\mathcal{T}(d,m-1)\big)$$

opérations de k.

Voici l'idée de la preuve : le procédé d'évaluation rapide mentionné ci-dessus permet de se ramener à effectuer des produits de séries univariées, dont on interpole le résultat, de manière rapide.

C.2 Algorithme d'Euclide étendu unitaire et sous-résultants

Soient k un corps et f et g deux polynômes de $k[t]$. L'algorithme d'Euclide étendu unitaire produit les suites de polynômes (r_i,s_i,t_i) où r_i est le $i$ème reste normalisé produit par l'algorithme d'Euclide appliqué à f et g, et s_i et t_i vérifient $s_if+t_ig=r_i$. En particulier la dernière ligne (r_l,s_l,t_l) vérifie $s_lf+t_lg=\mathrm{pgcd}(f,g)$, avec la convention usuelle de normalisation du pgcd. On trouvera la description dans [204, Algorithm 3.6].

Définition 30 (Sous-résultants) *Soient f et g deux polynômes de $k[t]$ de degrés respectifs $n \geq m$. Pour tout entier d, notons P_d l'espace vectoriel des polynômes à coefficients dans k et de degré strictement inférieur à d.*
Soit d un entier et φ_d l'application linéaire :

$$\begin{array}{rrcl} \varphi_d : & P_{m-d} \times P_{n-d} & \to & P_{n+m-2d} \\ & (s,t) & \mapsto & sf + tg \mod t^d. \end{array}$$

Nous appellerons dème sous-résultant *et noterons $S_d(f,g)$ le déterminant de S_d.*

Remarque Il existe une définition, donnée par exemple dans [36] du *d*ème sous-résultant comme un polynôme de $k[t]$ dont les coefficients sont des déterminants de matrices égales à la matrice de l'application φ_d, à la modification d'une colonne près. Ce sont les coefficients dominants de ces polynômes que l'on appelle ici sous-résultants.

Lemme 18 *Soient f et g deux polynômes de $k[t]$. Dans l'exécution de l'algorithme d'Euclide étendu unitaire, les termes de tête des restes successifs r_i et tous les dénominateurs apparaissant sont des produits des sous-résultants non nuls associés à f et g et des termes de tête de f et g.*

Démonstration. C'est une conséquence du théorème "fundamental theorem of pseudo-remainder sequences" donné par Loos dans [36]. □

À partir de polynômes f et g à coefficients dans un anneau de polynômes $k[p_1,\ldots,p_m]$, le lemme ci-dessus permet de contrôler le risque qu'une valeur $\mathbf{p}$ de k^m annule un dénominateur apparaissant au cours du calcul.

Corollaire 13 *Soient g un polynôme de $k[t]$ de degré d, dont les coefficients ont un degré au plus $\mathcal{D}$, et f le polynôme t^{d+1}.*
Il existe un polynôme P de $\mathbb{Z}[p_1,\ldots,p_m]$ tel que si $\mathbf{p}$ n'annule pas P alors $\mathbf{p}$ n'annule aucun des termes de tête ni des dénominateurs apparaissant au cours de l'algorithme d'Euclide étendu appliqué à f et g dans $k(p_1,\ldots,p_m)[t]$. On peut prendre P de degré au plus

$$d(d+1)\mathcal{D}.$$

Si les polynômes f et g sont à coefficients dans $\mathbb{Z}[p_1,\ldots,p_m]$ et ont une hauteur au plus h, alors on peut prendre P de hauteur au plus

$$d(d+1)\big(h + \log(2d) + \mathcal{D}\log(m+1)\big).$$

Démonstration. D'après le lemme précédent, il suffit de prendre pour P le produit des sous-résultants non nuls associés à f et g ; le polynôme f est unitaire, et le dernier sous-résultant S_d est égal au coefficient de tête de g.
D'après la proposition 9 du chapitre 3, le *d*ème sous-résultant, s'il est non nul, a un degré inférieur à $2(d-k)\mathcal{D}$, et si l'on travaille sur $\mathbb{Z}$, une hauteur inférieure à

$$2(d-k)\big(h + \log 2(d-k) + \mathcal{D}\log(m+1)\big),$$

ce qui conduit au résultat □

C.3 Translation multivariée

Dans les parties V, VI et VII, l'utilisation de l'opérateur de Newton formel dans un anneau de séries formelles est accompagnée d'un déplacement de l'origine des coordonnées ; il faut mesurer le coût de retour dans les coordonnées d'origine. Nous répondons à la question par un procédé d'évaluations et d'interpolations, en bougeant une variable après l'autre.

Lemme 19 (Évaluation et interpolation rapides) *Soient d un entier et $\mathcal{A}$ un anneau unitaire, et supposons qu'il existe $(d+1)$ points $(u_0,\ldots,u_d)$ dans $\mathcal{A}$ tels que pour tout couple (i,j) avec $i \neq j$, $u_i - u_j$ est inversible dans $\mathcal{A}$. Soient f un polynôme de $\mathcal{A}[t]$ de degré $d+1$ et $(u_0,\ldots,u_d)$ des points de $\mathcal{A}$. Alors :*

- *on peut évaluer f sur tous les points en $O(\mathcal{M}(d)\log d)$ opérations de $\mathcal{A}$;*
- *réciproquement, on peut déterminer les coefficients de f à partir des valeurs $(f(u_0),\ldots,f(u_d))$ en $O(\mathcal{M}(d)\log d)$ opérations de $\mathcal{A}$.*

Ces résultats apparaissent initialement dans [149] et se basent sur un algorithme de type "diviser pour régner". Ils sont présentés en détail dans [28, 204].

Corollaire 14 (Translation univariée) *Soient d un entier et $\mathcal{A}$ un anneau unitaire, et supposons qu'il existe $(d+1)$ points $(u_0,\ldots,u_d)$ dans $\mathcal{A}$ tels que pour tout couple (i,j) avec $i \neq j$, $u_i - u_j$ est inversible dans $\mathcal{A}$. Soient f un polynôme de $\mathcal{A}[t]$ de degré d, et t_1 un point de $\mathcal{A}$. On peut calculer $f(t+t_1)$ en $O(\mathcal{M}(d)\log d)$ opérations de $\mathcal{A}$.*

Démonstration. On évalue le polynôme f aux points u_i, puis on interpole ces valeurs aux points $u_i + t_1$, le tout avec la complexité annoncée. □

Corollaire 15 (Translation multivariée) *Soient d un entier et $\mathcal{A}$ un anneau unitaire, et supposons qu'il existe $(d+1)$ points $(u_0,\ldots,u_d)$ dans $\mathcal{A}$ tels que pour tout couple (i,j) avec $i \neq j$, $u_i - u_j$ est inversible dans $\mathcal{A}$. Soient f un polynôme de $\mathcal{A}[p_1,\ldots,p_m]$ de degré d et $\mathbf{p}_1 = (p_{1,1},\ldots,p_{1,m})$. On peut calculer $f(p_1+p_{1,1},\ldots,p_m+p_{1,m})$ en*

$$O\big(m\mathcal{T}(d,m-1)\mathcal{M}(d)\log d\big)$$

opérations de $\mathcal{A}$.

Démonstration. On bouge une variable après l'autre, ce qui ramène la question à effectuer m translations de polynômes univariés dans les anneaux $\mathcal{A}[p_1,\ldots,p_{i-1},p_{i+1},\ldots,p_m][p_i]$. Choisissons comme valeurs de spécialisation les points $(u_0,\ldots,u_d)$. Toutes les opérations faisant intervenir des polynômes de $\mathcal{A}[p_1,\ldots,p_{i-1},p_{i+1},\ldots,p_m]$ sont des additions et des multiplications par des constantes, on en déduit le résultat demandé. □

La preuve de la proposition suivante est immédiate.

Proposition 80 *Dans les deux modèles de multiplication des séries vus jusqu'ici, le coût de la reconstruction rationnelle est*

$$O_{log}(m\mathcal{M}_s(d',m))$$

opérations de k.

C.4 Reconstruction rationnelle multivariée

Cette section présente un algorithme de reconstruction rationnelle d'une fraction rationnelle multivariée à partir de son développement de Taylor à l'origine. Dans les parties V, VI et VII, cette fonction intervient à la sortie d'un processus de remontée de Newton, qui produit des développements en séries, à partir desquels on reconstruit des fractions rationnelles.

Dans toute cette section, on travaille sur un corps k. Pour un point $\gamma = (\gamma_2, \ldots, \gamma_m)$ de k^{m-1}, nous noterons $\mathbf{A}_\gamma$ la matrice du changement de variables

$$\begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_m \end{bmatrix} \leftarrow \begin{bmatrix} p'_1 \\ p'_2 + \gamma_2 p'_1 \\ \vdots \\ p'_m + \gamma_m p'_1 \end{bmatrix}.$$

Si f est un polynôme de $k[p_1, \ldots, p_m]$, le terme constant de $f \circ \mathbf{A}_\gamma$ est $P(1, \gamma_2, \ldots, \gamma_m)$.

Lemme 20 (Complexité du changement de variables) *Soient f un polynôme de $k[p_1, \ldots, p_m]$ de degre d inférieur au cardinal de k et $\gamma = (\gamma_2, \ldots, \gamma_m)$ un point de k^{m-1}. Il est possible de calculer la composée $f \circ \mathbf{A}_\gamma$ en*

$$O\Big(d\mathcal{M}(d) \log d \binom{d+m-1}{m-2}\Big)$$

opérations de k.

Démonstration. Le changement de variables considéré ne mélange pas les composantes homogènes de f, on s'intéresse donc à chacune d'entre elles séparément. Sur chacune de ces composantes, on peut faire $p_1 = 1$, la question se ramène alors à faire une translation de vecteur γ sur un polynôme de $m - 1$ variables. On applique les résultats de la section précédente, la complexité totale du processus est inférieure à

$$\begin{aligned} \sum_{i=1}^{d} m\mathcal{T}(i, m-2)\mathcal{M}(i) \log i &\leq m\mathcal{M}(d) \log d \sum_{i=1}^{d} \mathcal{T}(i, m-2) \\ &\leq 2\mathcal{M}(d) d \log d \binom{d+m-1}{m-2}, \end{aligned}$$

le passage de l'avant-dernière ligne à la dernière repose sur le calcul explicite de la somme

$$\sum_{i=1}^{d} \mathcal{T}(i, m-2),$$

qui se vérifie formellement, en utilisant par exemple le logiciel Maple. □

Nous proposons un algorithme probabiliste qui se ramène à une situation euclidienne. Pour cela, nous utilisons l'application

$$\begin{array}{rccl} \tilde{}: & k[p_1, \ldots, p_m] & \to & k(p_1, \ldots, p_m)[t] \\ & f(p_1, \ldots, p_m) & \mapsto & \tilde{f} = f(p_1 t, p_2 t, \ldots, p_m t). \end{array}$$

On s'intéresse spécialement aux séries r qui sont les développements de Taylor de fractions p/q :

Lemme 21 *Soient p et q deux polynômes de $k[p_1, \ldots, p_m]$ de degrés au plus d, avec $q(0) \neq 0$, et r leur développement de Taylor à l'origine à précision $d' \geq 2d$. Soient P et Q les polynômes de $k(p_1, \ldots, p_m)[t]$ produits par le calcul de reconstruction rationnelle de la section 5.2 du chapitre 5 appliqué à $\tilde{r}$ dans $k(p_1, \ldots, p_m)[t]$. Alors $P/Q|_{t=1} = p/q$.*

Démonstration. Les images $\tilde{p}$ et $\tilde{q}$ sont des approximants de $\tilde{r}$. L'unicité des approximants implique l'égalité $P/Q = \tilde{p}/\tilde{q}$, et la spécialisation en $t = 1$ donne le résultat. □

Sous les hypothèses du lemme ci-dessus, effectuons le calcul de reconstruction dans $k(p_1, \ldots ,p_m)[t]$ par l'application de l'algorithme d'Euclide étendu appliqué à $\tilde{r}$ et $t^{d'+1}$ (voir la section 5.2 du chaptre 5). Tous les dénominateurs et numérateurs des coefficients sont homogènes en les variables $\mathbf{p}$, on peut donc spécialiser la variable p_1 à 1 dans ce calcul. Pour retrouver p/q à partir de P/Q où p_1 est spécialisé à 1, il faut homogénéiser tous les coefficients en p_1 puis évaluer en $t = 1$. Pour écrire l'algorithme de reconstruction rationnelle, nous noterons `ReconstructionFraction`(P/Q) une telle procédure.

Effectuons les calculs à précision bornée. Si la variable p_1 ne divise aucun terme de tête des restes successifs, on peut tronquer tous les calculs à précision $O((p_2, \ldots ,p_m)^{d+1})$ pour retrouver P/Q. Pour se ramener sous de telles hypothèses, nous effectuons un changement de variables linéaire du type $\mathbf{A}_\gamma$ introduit au début de cette section. Dans ce cas, p'_1 ne divise aucun des termes de tête après changement de variables si et seulement si $(1,\gamma_2, \ldots ,\gamma_m)$ n'annule aucun des coefficients dominants des restes successifs.

FIG. C.1 – *Reconstruction rationnelle multivariée*

procedure `ReconstructionRationnelle`(s,γ)

Entree
\# s élément de $k[[p_1, \ldots ,p_m]]$ à précision d
\# γ élément de k^{m-1}
Sortie
\# **true**,(p,q) s'il existe p,q de $k[x_1, \ldots ,x_n]$ de degrés inférieurs à $d/2$
\# tels que $p = qs$ à précision d et $q(0) \neq 0$.
\# **false** sinon

$\mathbf{p} \leftarrow \mathbf{A}_\gamma \mathbf{p}$;
$s \leftarrow s(t,p_2t, \ldots ,p_mt)$;
\# On travaille à précision $d/2$,
\# l'algorithme suivant peut lever une erreur de division par zéro
$b,(P,Q) \leftarrow$ `ReconstructionRationnelleUnivariee`(s);
$\mathbf{p} \leftarrow \mathbf{A}_{-\gamma} \mathbf{p}$;
if not b **then return false fi**;
$p/q \leftarrow$ `ReconstructionFraction`(P/Q);
return true,p/q
end;

Résumons le comportement de l'algorithme dans la proposition suivante :

Proposition 81 *Soient p et q deux polynômes de $k[p_1, \ldots ,p_m]$ de degrés au plus d, avec $q(0) \neq 0$, et r le développement de Taylor de p/q à l'origine à précision $d' \geq 2d$. Il existe un polynôme P de $k[\gamma_2, \ldots ,\gamma_m]$ de degré au plus*

$$(d' + 1)d'^2$$

tel que pour tout γ n'annulant pas P, l'algorithme précédent appliqué à (r,γ) calcule p/q en

$$O\Big(\mathcal{M}(d')\log d'(d'\binom{d'+m-1}{m-2} + \mathcal{M}_s(d'/2,m-1))\Big)$$

opérations de k.

Démonstration. On prend pour P le produit des sous résultants non nuls de $\tilde{r}$ et $t^{d'+1}$. D'après la proposition 18, si γ n'annule pas P alors il n'annule aucun des termes apparaissant au cours du calcul. D'après le lemme 20 énoncé en début de section, le changement de variables et le changement de variables inverses se font en

$$O\Big(d'\mathcal{M}(d')\log d'\binom{d'+m-1}{m-2}\Big)$$

opérations de k. En cas de succès, le coût du calcul d'approximants est

$$O\big(\mathcal{M}(d')\log d'\mathcal{M}_s(d'/2,m-1)\big)$$

opérations de k. □

Proposition 82 *Soient p et q deux polynômes de $\mathbb{Z}[p_1,\ldots,p_m]$ de degrés d, de hauteur h, avec $q(0)\neq 0$. Soit r le développement de Taylor de p/q à l'origine à précision $d'\geq 2d$.*
Il existe un polynôme P de $\mathbb{Q}[\gamma_2,\ldots,\gamma_m]$ de degré au plus

$$(d'+1)d'^2$$

tel que pour tout γ n'annulant pas P, l'algorithme précédent appliqué à (r,γ) calcule p/q.
Soient M un entier et Γ l'ensemble $\{1,\ldots,M\}$. Soit γ un point chanceux de Γ^{m-1} pour l'algorithme de reconstruction rationnelle de r. Alors il existe un entier non nul e tel que pour tout premier p_1 ne divisant pas $q(0)e$, l'algorithme précédent effectué modulo p_1 est la trace du calcul sur les rationnels. On peut prendre e de hauteur inférieure à

$$d'(d'+1)^2(h+\log M+(d+2)\log(m+1)+2).$$

Démonstration. La première assertion est une redite de la proposition précédente ; on se concentre sur l'estimation de la hauteur de P. Écrivons le développement r sous la forme

$$\frac{s}{g(0)^{d'+1}}.$$

D'après la proposition 7 du chapitre 3, les hauteurs des coefficients de s et de $\tilde{s}$ sont bornées par $h'=(d'+1)(h+d\log(m+1)+1)$. Prenons pour polynôme P le produit des sous-résultants non nuls associés à $\tilde{r}$ et $t^{d'+1}$. D'après la proposition 13, les coefficients de P sont de hauteur au plus $d'(d'+1)(h'+\log(2d')+d'\log(m+1))$, que l'on peut majorer par

$$d'(d'+1)^2(h+(d+1)\log(m+1)+2).$$

Soit γ un point chanceux de Γ^m. Alors l'évaluation de P en γ est un entier non nul de hauteur au plus

$$d'(d'+1)^2(h+\log M+(d+2)\log(m+1)+2).$$

qui satisfait le critère demandé. □

Cette dernière proposition établit le résultat de complexité donné dans le chapitre 5.

Proposition 83 *Dans les deux modèles de multiplication des séries vus jusqu'ici, le coût de la reconstruction rationnelle est*

$$O_{log}(m\mathcal{M}_s(d',m))$$

opérations de k.

Démonstration. Aux facteurs logarithmiques près, la complexité de la reconstruction rationnelle est la somme de $C_1 = \mathcal{M}(d')d'\binom{d'+m-1}{m-2}$ et $C_2 = \mathcal{M}(d')\mathcal{M}_s(d'/2,m-1)$.

- Considérons pour commencer le modèle donné dans le chapitre 5 :

$$\mathcal{M}_s(d',m) = \mathcal{M}((2d'+1)^m).$$

  Le terme C_1 vérifie les inégalités

$$\begin{aligned} C_1 = \mathcal{M}(d')d'\frac{(d'+m-1)!}{(m-2)!(d'+1)!} &= O(\mathcal{M}(d')(d'+\frac{1}{m-2})\dots(d'+\frac{m-1}{m-2})) \\ &= O(\mathcal{M}((2d'+1)^m)). \end{aligned}$$

  La majoration pour le terme C_2 est immédiate.

- Mettons-nous dans le modèle donné dans la section C.1 :

$$\mathcal{M}_s(d',m) = O_{\log}(\mathcal{M}(d')\mathcal{M}(\mathcal{T}(d',m-1))).$$

  Le terme C_1 vérifie les inégalités

$$\begin{aligned} C_1 = \mathcal{M}(d')d'\frac{(d'+m-1)!}{(m-2)!(d'+1)!} &= O\big(\mathcal{M}(d')\frac{(d'+m-1)!}{(m-2)!d'!}\big) \\ &= O\big(m\mathcal{M}(d')\frac{(d'+m-1)!}{(m-1)!d'!}\big) \\ &= O\big(m\mathcal{M}(d')\mathcal{M}(\mathcal{T}(d',m-1))\big). \end{aligned}$$

  L'inégalité

$$d'\frac{(d'+m-1)!}{(m-1)!d'!} \leq m\frac{(d'+m)!}{m!d'!}$$

  implique

$$\mathcal{M}(d')\mathcal{M}(\mathcal{T}(d',m-1)) = O\big(m\mathcal{M}(\mathcal{T}(d',m))\big)$$

  donc

$$\mathcal{M}(d')\mathcal{M}(d')\mathcal{M}(\mathcal{T}(d',m-1)) = O\big(m\mathcal{M}(d')\mathcal{M}(\mathcal{T}(d',m))\big)$$

  et en particulier

$$\mathcal{M}(d')\mathcal{M}_s(d'/2,m-1) = O\big(m\mathcal{M}_s(d',m)\big),$$

  qui implique l'inégalité recherchée pour le terme C_2.

□

Annexe D

Arithmétique de Magma

Ce chapitre présente une brève évaluation des performances arithmétiques du logiciel Magma dans sa version 2.6, qui a été retenu pour l'implantation des algorithmes présentés dans ce mémoire. Tous les calculs présentés ci-dessous sont faits sur une machine Compaq XP/1000 - Alpha EV6 de l'UMS MEDICIS.

Les calculs présentés dans les parties V, VI et VII sont effectués dans des (empilements de) *quotients univariés sur des anneaux valués*, tels les séries formelles ou les nombres p-adiques : c'est le cadre de fonctionnement de l'opérateur de Newton formel de la partie 6. Voici donc une estimation du coût pratique de la multiplication dans un quotient de ce type.

Conclusion Retenons que le domaine de séries univariées est moins performant que le domaine des entiers modulo un nombre p^k, pour des éléments de taille analogues. Par ailleurs, nous devrons remplacer en pratique les calculs sur des séries multivariées par une suite de calculs sur des séries univariées, même si cette solution est moins bonne en théorie.

D.1 Calculs dans le quotient $\mathbb{Z}/p^k\mathbb{Z}[t]/q(t)$

Le tableau suivant donne le temps en secondes d'un produit de polynômes dans un quotient univarié de la forme $(\mathbb{Z}/p^k\mathbb{Z})[t]/q(t)$, pour $p = 9001$. Nous faisons varier à la fois le degré n de q (colonnes) et la précision k des coefficients (lignes).
D'après les résultats rappelés dans le chapitre 5, le coût d'un tel calcul est, à des facteurs logarithmiques près, linéaire en n et k. Pour des données de taille importante (la partie inférieure droite du tableau), le comportement pratique est proche d'être linéaire en le degré n du quotient, mais pas dans la précision k des coefficients. Dans ces plages d'utilisation, la multiplication des polynômes se fait déjà par transformée de Fourier, ce qui explique que le temps de calcul est approximativement doublé au passage d'une ligne à la suivante. Par contre, la multiplication des entiers suit toujours l'algorithme de Karatsuba : le temps est triplé au passage d'une colonne à l'autre.

D.2 Calculs dans le quotient $k[[s]][t]/q(t)$

Nous travaillons toujours dans un quotient univarié ; cette fois l'anneau de base n'est plus un anneau d'entiers modulo 9001^k, mais de séries formelles de précision k sur le corps $\mathbb{F}_{9001}$. Le tableau ci-dessous présente les mêmes expériences, pour des valeurs du degré n de l'extension et de la précision k du même ordre que dans le paragraphe précédent.

FIG. D.1 – *Multiplication dans* $\mathbb{Z}/p^k\mathbb{Z}[t]/q(t)$

n	k						
	16	32	64	128	256	512	1024
16	0.017	0.017	0.017	0.017	0.017	0.084	0.2
32	0.017	0.017	0.017	0.033	0.067	0.183	0.566
64	0.016	0.017	0.017	0.066	0.183	0.55	1.617
128	0.017	0.017	0.05	0.167	0.467	1.383	4.1
256	0.034	0.05	0.116	0.25	0.65	1.834	5.3
512	0.066	0.1	0.234	0.517	1.3	3.784	10.883
1024	0.217	0.25	0.483	1.083	2.733	7.9	23.084

FIG. D.2 – *Multiplication dans* $k[[s]][t]/q(t)$

n	k					
	8	16	32	64	128	256
8	0.017	0.017	0.017	0.05	0.133	0.45
16	0.017	0.017	0.05	0.15	0.45	1.401
32	0.017	0.05	0.15	0.466	1.467	4.501
64	0.066	0.166	0.467	1.45	4.484	13.983
128	0.217	0.517	1.466	4.5	13.85	43
256	0.683	1.567	4.434	13.649	42.233	131.651

La complexité théorique est à nouveau quasiment linéaire en n et k. En pratique, ce n'est pas le cas : le temps triple au passage d'une ligne ou d'une colonne à la suivante, signe de l'utilisation de l'algorithme de Karatsuba. Le calcul dans un anneau de séries formelles à précision k sur $\mathbb{F}_{9001}$ devrait être au moins aussi rapide que dans le quotient $\mathbb{Z}/9001^k\mathbb{Z}$, puisque les éléments de ces deux structures ont des représentations analogues. Ce n'est pas le cas.

D.3 Multiplications de séries multivariées

La suite naturelle des considérations précédentes est le calcul dans une extension d'un anneau de *séries multivariées*. Le tableau suivant donne le coût en secondes d'une multiplication de séries multivariées. Ces temps sont tels que nous utiliserons très peu ce domaine.

FIG. D.3 – *Multiplication dans* $k[[p_1,\ldots,p_m]]$

Variables	Précision						
	10	15	20	25	30	35	40
2	0.05	0.28	0.63	1.5	3.3	7.3	11
3	0.8	3.3	12.3	44.8	141	479	965
4	1.83	32	198	1157			

En pratique, cette constatation a des répercussions importantes. Nous avons remplacé une suite d'additions et de multiplications dans l'anneau $k[[p_1,\ldots,p_m]]$ par des calculs à coefficients dans un anneau de séries univariées, de la manière suivante.
À une série s appartenant à $k[[p_1,\ldots,p_m]]$, nous associons la série $s(t,\alpha_2 t,\ldots,\alpha_m t)$ de $k[[t]]$, pour un jeu de valeurs $(\alpha_2,\ldots,\alpha_m)$ de k^{m-1}. Nous effectuons la suite de calculs pour suffisamment de valeurs des α, et retrouvons le résultat dans $k[[p_1,\ldots,p_m]]$ en interpolant ses coefficients. L'interpolation polynomiale multivariée est un problème délicat en théorie ; nous nous sommes contentés d'un processus d'interpolation univariée récursif, qui est la méthode la plus simple décrite par Zippel dans [213]. Si on dispose d'un processus d'évaluation et d'interpolation rapide, on aboutit à l'algorithme de produit rapide présenté dans l'annexe C.

Le tableau ci-dessous décrit les temps d'une multiplication effectuée par cette méthode, pour les mêmes ordres de grandeur que précédemment.

FIG. D.4 – *Multiplication dans* $k[[p_1,\ldots,p_m]]$*, 2*^{e} *forme*

Variables	Précision						
	10	15	20	25	30	35	40
2	0.016	0.03	0.06	0.13	0.2	0.33	0.41
3	0.13	0.51	1.4	3.2	6.2	11	17
4	1.3	7.5	28	79	185	385	680

Il apparaît que pour des plus grandes valeurs des arguments, qui ne sont pas rencontrées dans ce mémoire, la première méthode redevient meilleure.

Conclusion Les implantations utilisent cet artifice, même si sa complexité est théoriquement moins bonne.

D.4 Opérations modulo un ensemble triangulaire

Nous abordons maintenant le calcul modulo un ensemble triangulaire. L'anneau de base est un anneau de séries formelles univariées : nous fixons un entier d, et travaillons dans un quotient de degré d de $\mathbb{F}_{9001}[[s]][x_1,\dots,x_n]$, en comparant deux cas de figure :

- Dans le premier cas, la variable x_n est séparante, son polynôme minimal est de degré d, et les variables $(x_1,\dots,x_{n-1})$ sont données par un polynôme en x_n de degré au plus $d-1$. Il suffit de travailler dans une extension univariée, le domaine de calcul est $\mathbb{F}_{9001}[[s]][x_n]/q(x_n)$.
- Nous considérons ensuite un ensemble triangulaire $\mathbf{t} = (t_1,\dots,t_n)$ de la forme

$$\begin{cases} t_1(x_1,x_2,\dots,x_n), \\ t_2(x_2,\dots,x_n), \\ \vdots \\ t_n(x_n), \end{cases}$$

  où pour tout k de $\{1,\dots,n\}$, le polynôme t_k est unitaire de degré d_k en x_k. Le produit des degrés $d_1 \dots d_n$ est pris égal à d, et le domaine de calculs est un empilement de n extensions univariées de $\mathbb{F}_{9001}[[s]]$.

Nous allons prendre $n = 6$, $d = 256$, et étudier pour commencer le cas où x_n est une variable séparante, puis le cas où $(d_1,\dots,d_n)$ est égal à (1,1,1,2,2,64), puis à (256,1,1,1,1,1) et enfin à (2,4,2,2,4,2). Les coefficients sont des séries formelles univariées dont la précision varie de 0 (constantes) à 50.

FIG. D.5 – *Multiplication dans un quotient triangulaire*

Suite des degrés	Univarié	(1,1,1,2,2,64)	(256,1,1,1,1,1)	(2,4,2,2,4,2)
Précision=0	0.35	0.8	2	2
Précision=10	1	2.25	3	7
Précision=50	10	20	20	75

Conclusion Le calcul modulo un ensemble triangulaire entraîne un surcoût, dépendant de la suite des degrés et difficile à estimer a priori.

Annexe E

Index des systèmes rencontrés

Cette annexe regroupe l'ensemble des systèmes utilisés dans ce texte. Nous les donnons sous forme de calculs d'évaluation, en reprenant directement le code Magma utilisé dans les implantations.

E.1 La plateforme de Stewart

Ce système est utilisé dans le chapitre 21, où l'on met en œuvre un procédé de résolution de systèmes surdéterminés à paramètres. Nous n'avons pas cherché à le résoudre formellement.

Fig. E.1 – *Plateforme de Stewart surdéterminée*

Variables	Paramètres	Équations
7	7	8

```
Syst:=function(Para,Var)

// 6+1 mesures
  d1:=Para[1];   d2:=Para[2];   d3:=Para[3];
  d4:=Para[4];   d5:=Para[5];   d6:=Para[6];
  d7:=Para[7];

// 7 inconnues (un quaternion et une translation)
  e0:=Var[1];     e1:=Var[2];     e2:=Var[3];
  e3:=Var[4];     Tx:=Var[5];     Ty:=Var[6];
  Tz:=Var[7];

// matrice de rotation
  c0:=e0^2;       c1:=e1^2;       c2:=e2^2;
  c3:=e3^2;       r12:=e1*e2;     r13:=e1*e3;
  r03:=e0*e3;     r01:=e0*e1;     r02:=e0*e2;
  r23:=e2*e3;
```

```
  R11:=c0+c1-c2-c3;        R12:=2*(r12+r03);
  R13:=2*(r13-r02);        R21:=2*(r12-r03);
  R22:=c0-c1+c2-c3;        R23:=2*(r23+r01);
  R31:=2*(r13+r02);        R32:=2*(r23-r01);
  R33:=c0-c1-c2+c3;

// Coordonnees des points fixes
  x1:=3*2;    x2:=4*2;     x3:=0;
  x4:=-1*2;   x5:=-5*2;    x6:=-5*2;
  y1:=2*2;    y2:=-2*2;    y3:=4*2;
  y4:=-4*2;   y5:=2*2;     y6:=-2*2;
  z1:=0;      z2:=0;       z3:=0;
  z4:=0;      z5:=0;       z6:=0;

  X1:=R11*x1+R12*y1+R13*z1+Tx;
  Y1:=R21*x1+R22*y1+R23*z1+Ty;
  Z1:=R31*x1+R32*y1+R33*z1+Tz;
  X2:=R11*x2+R12*y2+R13*z2+Tx;
  Y2:=R21*x2+R22*y2+R23*z2+Ty;
  Z2:=R31*x2+R32*y2+R33*z2+Tz;
  X3:=R11*x3+R12*y3+R13*z3+Tx;
  Y3:=R21*x3+R22*y3+R23*z3+Ty;
  Z3:=R31*x3+R32*y3+R33*z3+Tz;
  X4:=R11*x4+R12*y4+R13*z4+Tx;
  Y4:=R21*x4+R22*y4+R23*z4+Ty;
  Z4:=R31*x4+R32*y4+R33*z4+Tz;
  X5:=R11*x5+R12*y5+R13*z5+Tx;
  Y5:=R21*x5+R22*y5+R23*z5+Ty;
  Z5:=R31*x5+R32*y5+R33*z5+Tz;
  X6:=R11*x6+R12*y6+R13*z6+Tx;
  Y6:=R21*x6+R22*y6+R23*z6+Ty;
  Z6:=R31*x6+R32*y6+R33*z6+Tz;

  f0:=c0+c1+c2+c3-1;
  f1:=(x1-X1)^2+(y1-Y1)^2+(z1-Z1)^2-d1^2;
  f2:=(x2-X2)^2+(y2-Y2)^2+(z2-Z2)^2-d2^2;
  f3:=(x3-X3)^2+(y3-Y3)^2+(z3-Z3)^2-d3^2;
  f4:=(x4-X4)^2+(y4-Y4)^2+(z4-Z4)^2-d4^2;
  f5:=(x5-X5)^2+(y5-Y5)^2+(z5-Z5)^2-d5^2;
  f6:=(x6-X6)^2+(y6-Y6)^2+(z6-Z6)^2-d6^2;
// 8 equations
  return [f0,f1,f2,f3,f4,f5,f6,X1-d7];
end function;
```

E.2 Le système P19

Ce système a été proposé par P. Gaudry et R. Harley, lors de la mise au point de leur article [74]. Sa résolution permet la division par deux dans la Jacobienne d'une courbe hyperelliptique (voir le chapitre 23). Le corps de base est $\mathbb{F}_p$, où p est le plus petit nombre premier supérieur à 10^{19}, et utilise les constantes

$$\begin{aligned} f_1 &= 3141592653589793238, \\ f_2 &= 4626433832795028841, \\ f_3 &= 9716939937510582097, \\ f_4 &= 4944592307816406286, \\ f_5 &= 2089986280348253421. \end{aligned}$$

Le système est formé de 4 équations dépendant de 4 inconnues (V_0,V_1,U_1,U_0) et de deux paramètres (C_1,C_2). Ses caractéristiques sont données dans le tableau ci-dessous, les calculs sont effectués dans l'ouvert défini par Z3$\neq$ 0, cette grandeur est définie dans le programme qui suit.

FIG. E.2 – *Division par deux dans une Jacobienne*

Variables	Paramètres	Multiplications	Degrés
4	2	253	(4,4,9,10)

```
Syst:=function(Para,Var)

  C1:=Para[1];
  C2:=Para[2];
  V0:=Var[1];
  V1:=Var[2];
  U1:=Var[3];
  U0:=Var[4];

  t1 := U1^2;     t2 := t1^2;      t3 := t1*U1;       t4 := f1*t3;
  t5 := 3*U0;     t8 := f1*U0;     t10 := V0^2;       t13 := U0^2;
  t14 := f2*U0;   t20 := 2*t13;    t23 := f1*t13;     t26 := V1^2;
  t28 := t10^2;   t29 := t28*t10;  t33 := 8*U0;       t34 := 2*f2;
  t37 := 4*t8;    t38 := 2*f3;     t43 := f1*V1;      t46 := 4*U0;
  t50 := t10*V0;  t52 := t2*t1;    t53 := t2*U1;      t56 := f1^2;
  t57 := -t33+t56+t34;             t59 := 12*t8;      t60 := f2*f1;
  t61 := 2*t60;   t62 := t59-t61-t38;                 t64 := 16*t13;
  t65 := 4*t56;   t66 := 8*f2;     t67 := -t65-t66;   t69 := f3*f1;
  t70 := 2*t69;   t71 := f2^2;     t72 := t64+t67*U0+t70+t71;
  t75 := 4*t60;   t76 := 8*f3;     t79 := f3*f2;      t80 := 2*t79;
  t81 := -16*t23+(t75+t76)*U0-t80; t84 := 4*t56*t13;  t86 := 4*t69*U0;
  t87 := f3^2;    t100 := 6*f3;    t108 := 4*t69;     t109 := 2*t71;
```

```
  t114 := t60+f3; t122 := f1*t26;  t126 := 6*f2;      t130 := 8*t8;
  t134 := 4*t13;  t141 := t28*V0;  t146 := 5*U0;      t155 := 10*U0;
  t164 := t2*t3;  t172 := 2*t56;   t173 := 4*f2;      t178 := 5*t26;
  t184 := 2*t122; t185 := t13*U0;  t187 := f2*t13;    t188 := 6*t187;
  t193 := 2*U0;   t196 := f1*t185; t199 := t79*U0;
  t200 := t164-2*f1*t52+(-t146+t56+t34)*t53+(7*t8-t61-t38)*t2+
         (t20+(-t172-t173)*U0+t70+t71)*t3+(-t178+(t60+5*f3)*U0-t80)*t1+
         (t184+8*t185-t188+(-t70+t71)*U0+t87)*U1+
         (-t193-f2)*t26-4*t196+2*t114*t13-t199;
  t214:=-t67*U0; t218 := t26*V1;  t219 := 4*t218;    t226 := 4*t185;
  t259:= t26^2;  t261 := 4*t23;
  t266:= 2*U1*t29-2*V1*t141+2*(2*t2-2*t4+(-t146+t34)*t1+2*(t8-f3)*U1-t134+t14)
         *t28+2*(V1*t3+(t155-f2)*V1*U1+(-t37+t38)*V1)*t50+2*t200*t10+2*
         (2*V1*t52-3*t43*t53+(t5+t56+t34)*V1*t2+(-t130-t60-f3)*V1*t3+
         (-20*t13+t214)*V1*t1+(t219+(20*t23-6*t114*U0+t79)*V1)*U1+
         (t226+4*(-t56-f2)*t13+(t108+t71)*U0-t87)*V1)*V0-6*t26*t53+10*t122*t2+
         2*(14*U0-t172-t173)*t26*t3+2*(-16*t8+3*t60+3*f3)*t26*t1+2*
         (-8*t13+(t65+t126)*U0-t70-t71)*t26*U1-4*t259+2*
         (t261-2*t114*U0+t79)*t26;
  t303 :=t2^2;   t326 := 2*t199;  t331 := t13^2;
  t336 := t303-2*f1*t164+(-t193+t56+t34)*t52+2*(t8-t60-f3)*t53+
         (-3*t13+t70+t71)*t2+(2*t26+t261+2*(-t60+f3)*U0-t80)*t3+
         (-t184+t226-t188+2*t71*U0+t87)*t1+(2*f2*t26+4*f3*t13-t326)*U1+
         (t37-t100)*t26+4*t331-4*f2*t185+t71*t13;
  t394 :=(t1+t46)*t29-6*V1*U1*t141+(2*t53-2*f1*t2+2*(U0+f2)*t3+(-t37-t38)*t1
         +(-12*t13+6*t14)*U1+t178+t261-4*f3*U0)*t28+(-8*V1*t2+8*t43*t3+
         (22*U0-t66)*V1*t1+(-t59+t76)*V1*U1+(-t134+2*t14)*V1)*t50+t336*t10+
         (-2*V1*t164+4*t43*t52+(t155-t172-t173)*V1*t53+(-14*t8+t75+4*f3)*
         V1*t2+(-t134+t214-t108-t109)*V1*t3+(-t219+((-t61-10*f3)*U0+4*t79)
         *V1)*t1+(-16*t185+12*t187+(t108-t109)*U0-2*t87)*V1*U1+(-t33+t173)*
         t218+(8*t196-4*t114*t13+t326)*V1)*V0+t26*t52-2*t122*t53+t57*t26*t2+
         t62*t26*t3+t72*t26*t1+(8*t259+t81*t26)*U1-4*f1*t259+(t84-t86+t87)*t26;
  Z1 :=  t2-t4+(-t5+f2)*t1+(2*t8+t10-f3)*U1+t13-t14-2*V1*V0+f4;
  Z2 :=  U0*t3-t8*t1+(-t20+t14)*U1+t23+(t10-f3)*U0-t26+f5;
  Z3 :=  t29+(2*t3-2*f1*t1+(-t33+t34)*U1+t37-t38)*t28+(6*V1*t1-4*t43*U1+
         (-t46+t34)*V1)*t50+(t52-2*f1*t53+t57*t2+t62*t3+t72*t1+t81*U1+t84-
         t86+t87)*t10+(6*V1*t53-10*t43*t2+(-28*U0+t65+t66)*V1*t3+
         (32*t8-6*t60-t100)*V1*t1+(t64+(-8*t56-12*f2)*U0+t108+t109)*V1*U1
         +(-8*t23+4*t114*U0-t80)*V1)*V0+9*t26*t2-12*t122*t3+(-12*U0+t65+t126)
         *t26*t1+(t130-t75)*t26*U1+(t134-4*t14+t71)*t26;
  Z4 := t266;
  Z5 := t394;
  U:=Z4-C1*Z3;
  V:=Z5-C2*Z3;
  return([Z1,Z2,U,V]);
end function;
```

E.3 Les équations modulaires en genre 1

Voici le code Magma qui permet d'évaluer la famille de systèmes de la section 24.3 du chapitre 24. Nous engendrons des tables contenant des termes carrés ψ_k^2 et rectangles $\psi_k \psi_{k+2}$, qui permettent de partager ces expressions au mieux. La sortie du calcul est l'ensemble des trois numérateurs du système.

Pour un nombre premier ℓ fixé, la question est de déterminer le polynôme minimal de la dernière variable, en fonction du paramètre b, en-dehors de l'hypersurface définie par $y = 0$. Ce polynôme est de degré $\ell + 1$ en t.

FIG. E.3 – *Équations modulaires en genre 1,* $\ell = 13$

Variables	Paramètres	Multiplications	Degrés
3	1	app. 130	(3,89,87)

```
forward Do_c,Do_R,Do_Psi;

Do_C:=procedure(c,b,x,y,~TablePsi,~TableR,~TableC)
  if TablePsi[c][1] eq false then
    Do_Psi(c,b,x,y,~TablePsi,~TableR,~TableC);
  end if;
  TableC[c]:=[*true,[TablePsi[c][2][1]^2,TablePsi[c][2][2]^2*];
end procedure;

Do_R:=procedure(c,b,x,y,~TablePsi,~TableR,~TableC)
  if TablePsi[c][1] eq false then
    Do_Psi(c,b,x,y,~TablePsi,~TableR,~TableC);
  end if;
  if TablePsi[c+2][1] eq false then
    Do_Psi(c+2,b,x,y,~TablePsi,~TableR,~TableC);
  end if;
  TableR[c]:=[*true,[TablePsi[c][2][1]*TablePsi[c+2][2][1],
                     TablePsi[c][2][2]*TablePsi[c+2][2][2]]*];
end procedure;

Do_Psi:=procedure(c,b,x,y,~TablePsi,~TableR,~TableC)
  if c eq 1 then
    TablePsi[c]:= [*true,[Parent(y)!1,1]*];
  elif  c eq 2 then
    TablePsi[c]:= [*true,[2*y,1]*];
  elif c eq 3 then
    TablePsi[c]:= [*true,[3*x^4+6*x^2+12*b*x-1,1]*];
  elif c eq 4 then
```

```
    TablePsi[c]:= [*true,[2*y*(2*x^6+10*x^4+40*b*x^3-10*x^2-8*b*x-16*b^2-2),1]*];
  elif IsEven(c) then
    n:=c div 2;
    if TableR[n][1] eq false then
      Do_R(n,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    if TableR[n-2][1] eq false then
      Do_R(n-2,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    if TableC[n-1][1] eq false then
      Do_C(n-1,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    if TableC[n+1][1] eq false then
      Do_C(n+1,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    TablePsi[c]:=[*true,
      [TableR[n][2][1]*TableC[n-1][2][1]*TableR[n-2][2][2]*TableC[n+1][2][2] -
       TableR[n][2][2]*TableC[n-1][2][2]*TableR[n-2][2][1]*TableC[n+1][2][1],
    TableR[n][2][2]*TableC[n-1][2][2]*TableR[n-2][2][2]*TableC[n+1][2][2]*2*y]*];
  else
    n:=(c-1) div 2;
    if TableR[n][1] eq false then
      Do_R(n,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    if TableR[n-1][1] eq false then
      Do_R(n-1,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    if TableC[n][1] eq false then
      Do_C(n,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    if TableC[n+1][1] eq false then
      Do_C(n+1,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    TablePsi[c]:=[*true,
      [TableR[n][2][1]*TableC[n][2][1]*TableR[n-1][2][2]*TableC[n+1][2][2] -
       TableR[n][2][2]*TableC[n][2][2]*TableR[n-1][2][1]*TableC[n+1][2][1],
     TableR[n][2][2]*TableC[n][2][2]*TableR[n-1][2][2]*TableC[n+1][2][2]]*];
  end if;
end procedure;


Syst:=function(Para,Var,l)
  b:=Para[1];
  x:=Var[1];
  y:=Var[2];
  z:=Var[3];
```

```
  TablePsi:=[ [*false,[1,0]*] : i in [1..l+2]];
  TableR:=[ [*false,[1,0]*] : i in [1..l+2]];
  TableC:=[ [*false,[1,0]*] : i in [1..l+2]];
  Do_Psi(l,b,x,y,~TablePsi,~TableR,~TableC);
  G:=[0,1];

  for i:=2 to (l-1) div 2 do
    if TableR[i-1][1] eq false then
      Do_R(i-1,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    if TableC[i][1] eq false then
      Do_C(i,b,x,y,~TablePsi,~TableR,~TableC);
    end if;
    G:=[TableR[i-1][2][1]*TableC[i][2][2]*G[2]+TableR[i-1][2][2]*
        TableC[i][2][1]*G[1],TableR[i-1][2][2]*TableC[i][2][1]*G[2]];
  end for;

  G:=[G[2]*(l-1)*x-2*G[1],2*G[2]];
  Tmp:=TablePsi[l][2][1];
  return [y^2-(x^3+x+b),Tmp,G[2]*z-G[1]];
end function;

K<b,x,y,z>:=PolynomialRing(Rationals(),4);
Syst([b],[x,y,z],13);
```

E.4 L'équation modulaire en genre 2 pour $\ell = 3$

Le système suivant est utilisé pour calculer le polynôme Ξ_3 pour le genre 2 ; la construction est décrite dans le chapitre 24. Avec les notations de la section 24.4 du chapitre 24, le calcul d'évaluation ci-dessous évalue les polynômes (f_1,f_2,f_5), sur lequel nous effectuons la remontée de Newton.

Variables	Paramètres	Multiplications	Degrés	Hauteur
3	3	50	(32,32,1)	23

```
Syst:=function(Para,Var)
  p3:=Para[1];
  p4:=Para[2];
  p5:=Para[3];

  x:=Var[2];
  x_2:=x^2;         x_3:=x*x_2;      x_4:=x_2*x_2;      x_5:=x_4*x;
  f:= x_5+x_3+p3*x_2+p4*x+p5;      fp:=-3*x_2-5*x_4-2*p3*x-p4;
  fp2:=3*x+p3+10*x_3;              fp3:=-1-10*x_2;  fp4:=5*x;
```

```
  t1:=fp^2;         t10:=-2*t1*fp+8*f*(fp*fp2-2*fp3*f);
  t11:=t1^2;        t13:=fp2*f;     t16:=f^2;        t17:=fp2^2;
  t20:=t16*fp;
  t26:=-5*t11+24*t13*t1-16*t16*t17-32*t20*fp3+64*fp4*t16*f;
  Psi4:=t10;        Psi3:=4*f;      Tmp2:=t26;      Gamma3:= -Tmp2;
  Alpha3:= -2*fp;
  d0_1:=-Psi4;      d1_1:=-Gamma3+Psi4*Alpha3;       d2_1:=-16*f*Psi3^2;
  D1_1:=2*d1_1+x*d2_1;
  D0_1:=x_2*d2_1+4*x*d1_1+16*d0_1*f;

  x:=Var[1];
  x_2:=x^2;          x_3:=x*x_2;    x_4:=x_2*x_2;    x_5:=x_4*x;
  f:= x_5+x_3+p3*x_2+p4*x+p5;      fp:=-3*x_2-5*x_4-2*p3*x-p4;
  fp2:=3*x+p3+10*x_3;              fp3:=-1-10*x_2;  fp4:=5*x;
  t1:=fp^2;          t10:=-2*t1*fp+8*f*(fp*fp2-2*fp3*f);
  t11:=t1^2;         t13:=fp2*f;    t16:=f^2;        t17:=fp2^2;
  t20:=t16*fp;
  t26:=-5*t11+24*t13*t1-16*t16*t17-32*t20*fp3+64*fp4*t16*f;
  Psi4:=t10;         Psi3:=4*f;     Tmp2:=t26;      Gamma3:= -Tmp2;
  Alpha3:= -2*fp;
  d0_2:=-Psi4;       d1_2:=-Gamma3+Psi4*Alpha3;      d2_2:=-16*f*Psi3^2;
  D1_2:=2*d1_2+x*d2_2;
  D0_2:=x_2*d2_2+4*x*d1_2+16*d0_2*f;

  return [D1_1*d2_2-D1_2*d2_1, D0_1*d2_2-D0_2*d2_1,Var[1]+Var[2]+Var[3]];
end function;
```

La solution initiale est calculée à l'aide du même système, auquel nous rajoutons les équations ci-dessous, pour sélectionner la composante qui nous intéresse. Toujours avec les notations de la section 24.4 du chapitre 24, voici les polynômes (f_0,f_3,f_4), cette fois sous forme de code Maple. Nous effectuons tous les calculs dans l'ouvert défini par l'équation

$$(x_1 - x_2)y_1y_2 \neq 0.$$

```
Eps0:=
4669440*x^25 + 18186240*x^23 + 3932160*p3*x^22 + (-21790720*p4 +
    32358400)*x^21 + (-2949120*p3 + 19660800*p5)*x^20 + (-82575360*p3^2
    - 133201920*p4 + 46202880)*x^19 + (-464650240*p3*p4 + 90439680*p3 +
    25231360*p5)*x^18 + (-47185920*p3^2 - 539688960*p3*p5 -
    531578880*p4^2 - 38338560*p4 + 28753920)*x^17 + (-78643200*p3^3 -
    743178240*p3*p4 + 147456000*p3 - 1535508480*p4*p5 +
    349962240*p5)*x^16 + (-723517440*p3^2*p4 + 231997440*p3^2 -
    549191680*p3*p5 - 1097400320*p4^2 + 177930240*p4 - 1258291200*p5^2
    - 4096000)*x^15 + (173015040*p3^3 - 896532480*p3^2*p5 -
    1604321280*p3*p4^2 + 306708480*p3*p4 - 7864320*p3 -
    3110338560*p4*p5 + 678297600*p5)*x^14 + (62914560*p3^4 +
    235929600*p3^2*p4 - 7864320*p3^2 - 4706795520*p3*p4*p5 +
    1386086400*p3*p5 - 976158720*p4^3 - 254115840*p4^2 + 33914880*p4 -
```

```
2736783360*p5^2 - 245760)*x^13 + (178257920*p3^3*p4 - 5242880*p3^3
+ 1014497280*p3^2*p5 - 607518720*p3*p4^2 + 77332480*p3*p4 -
3895459840*p3*p5^2 - 327680*p3 - 4367319040*p4^2*p5 -
369623040*p4*p5 + 267386880*p5)*x^12 + (408944640*p3^3*p5 -
23592960*p3^2*p4^2 + 70778880*p3^2*p4 - 1427374080*p3*p4*p5 +
876871680*p3*p5 - 675348480*p4^3 - 18677760*p4^2 -
6810501120*p4*p5^2 + 2457600*p4 - 719585280*p5^2)*x^11 +
(62914560*p3^3*p4 - 47185920*p3^2*p4*p5 + 1132462080*p3^2*p5 -
287047680*p3*p4^3 - 98304000*p3*p4^2 + 5898240*p3*p4 -
2052587520*p3*p5^2 - 3159490560*p4^2*p5 + 483655680*p4*p5 -
3538944000*p5^3 + 22609920*p5)*x^10 + (20971520*p3^4*p4 +
734003200*p3^3*p5 + 15728640*p3^2*p4^2 + 2621440*p3^2*p4 -
562298880*p3^2*p5^2 - 1507983360*p3*p4^2*p5 + 1167851520*p3*p4*p5 +
84213760*p3*p5 - 131809280*p4^4 - 179896320*p4^3 - 1392640*p4^2 -
5882511360*p4*p5^2 + 364380160*p5^2)*x^9 + (188743680*p3^4*p5 +
47185920*p3^3*p4^2 + 1462763520*p3^2*p4*p5 + 117964800*p3^2*p5 -
129761280*p3*p4^3 - 17694720*p3*p4^2 - 3350200320*p3*p4*p5^2 +
849346560*p3*p5^2 - 861143040*p4^3*p5 - 342097920*p4^2*p5 +
64880640*p4*p5 - 3633315840*p5^3)*x^8 + (629145600*p3^3*p4*p5 +
78643200*p3^3*p5 + 31457280*p3^2*p4^3 - 19660800*p3^2*p4^2 +
1108869120*p3^2*p5^2 + 428605440*p3*p4^2*p5 + 153354240*p3*p4*p5 -
2264924160*p3*p5^3 - 66355200*p4^4 - 24576000*p4^3 -
2296381440*p4^2*p5^2 - 825753600*p4*p5^2 + 82575360*p5^2)*x^7 +
(20971520*p3^4*p5 - 5242880*p3^3*p4^2 + 503316480*p3^3*p5^2 +
707788800*p3^2*p4^2*p5 + 157286400*p3^2*p4*p5 + 11796480*p3*p4^4 -
41943040*p3*p4^3 + 298844160*p3*p4*p5^2 + 228065280*p3*p5^2 -
116654080*p4^3*p5 - 37355520*p4^2*p5 - 2548039680*p4*p5^3 -
844103680*p5^3)*x^6 + (62914560*p3^3*p4*p5 - 15728640*p3^2*p4^3 +
1014497280*p3^2*p4*p5^2 + 271319040*p3^2*p5^2 +
389283840*p3*p4^3*p5 - 29491200*p3*p4^2*p5 - 503316480*p3*p5^3 +
4423680*p4^5 - 17448960*p4^4 - 377487360*p4^2*p5^2 -
23592960*p4*p5^2 - 943718400*p5^4)*x^5 + (141557760*p3^3*p5^2 +
7864320*p3^2*p4^2*p5 + 251658240*p3^2*p5^3 - 10813440*p3*p4^4 +
833617920*p3*p4^2*p5^2 + 94371840*p3*p4*p5^2 + 106168320*p4^4*p5 -
62914560*p4^3*p5 - 754974720*p4*p5^3 - 62914560*p5^3)*x^4 +
(20971520*p3^4*p5^2 - 10485760*p3^3*p4^2*p5 + 1310720*p3^2*p4^4 +
165150720*p3^2*p4*p5^2 - 35389440*p3*p4^3*p5 + 482344960*p3*p4*p5^3
- 36700160*p3*p5^3 - 1474560*p4^5 + 319815680*p4^3*p5^2 -
90439680*p4^2*p5^2 - 377487360*p5^4)*x^3 + (31457280*p3^3*p4*p5^2 -
15728640*p3^2*p4^3*p5 + 47185920*p3^2*p5^3 + 1966080*p3*p4^5 +
23592960*p3*p4^2*p5^2 + 62914560*p3*p5^4 - 8847360*p4^4*p5 +
361758720*p4^2*p5^3 - 125829120*p4*p5^3)*x^2 + (15728640*p3^3*p5^3
+ 3932160*p3^2*p4^2*p5^2 - 4915200*p3*p4^4*p5 + 737280*p4^6 +
188743680*p4*p5^4 - 62914560*p5^4)*x + 10485760*p3^2*p4*p5^3 -
5242880*p3*p4^3*p5^2 - 20971520*p3*p5^4 + 655360*p4^5*p5 +
5242880*p4^2*p5^3 + 41943040*p5^5;
```

```
Eps1:=
-11878400*x^24 - 57507840*x^22 - 20316160*p3*x^21 + (133201920*p4 -
    157532160)*x^20 + (-289996800*p3 + 403046400*p5)*x^19 +
    (-146800640*p3^2 + 185630720*p4 - 203489280)*x^18 +
    (212336640*p3*p4 - 778567680*p3 + 1111818240*p5)*x^17 +
    (-1038090240*p3^2 + 1227816960*p3*p5 + 462766080*p4^2 -
    362741760*p4 - 66109440)*x^16 + (-477102080*p3^3 - 1340866560*p3*p4
    - 460718080*p3 + 2325217280*p4*p5 + 637009920*p5)*x^15 +
    (-1179648000*p3^2*p4 - 1073479680*p3^2 + 648806400*p3*p5 -
    103219200*p4^2 - 382402560*p4 + 2202009600*p5^2 + 32931840)*x^14 +
    (-1038090240*p3^3 - 283115520*p3^2*p5 - 813957120*p3*p4^2 -
    2135162880*p3*p4 + 135659520*p3 + 2677800960*p4*p5 -
    29491200*p5)*x^13 + (-356515840*p3^4 - 3428843520*p3^2*p4 +
    222822400*p3^2 + 853278720*p3*p4*p5 - 1346764800*p3*p5 -
    137953280*p4^3 - 953057280*p4^2 + 140410880*p4 + 3680501760*p5^2 +
    2211840)*x^12 + (-1667235840*p3^3*p4 + 204472320*p3^3 -
    3090677760*p3^2*p5 - 3698196480*p3*p4^2 + 397148160*p3*p4 +
    2453667840*p3*p5^2 + 8847360*p3 + 1097072640*p4^2*p5 -
    460062720*p4*p5 + 178913280*p5)*x^11 + (125829120*p3^4 -
    1761607680*p3^3*p5 - 2925527040*p3^2*p4^2 + 448266240*p3^2*p4 +
    11796480*p3^2 - 5886443520*p3*p4*p5 + 397148160*p3*p5 -
    1369374720*p4^3 + 207421440*p4^2 + 3350200320*p4*p5^2 + 13271040*p4
    + 1285816320*p5^2)*x^10 + (41943040*p3^5 + 367001600*p3^3*p4 +
    5242880*p3^3 - 5976883200*p3^2*p4*p5 + 173015040*p3^2*p5 -
    2346188800*p3*p4^3 + 330301440*p3*p4^2 + 35389440*p3*p4 -
    904396800*p3*p5^2 - 3135897600*p4^2*p5 + 614727680*p4*p5 +
    2123366400*p5^3 + 26542080*p5)*x^9 + (188743680*p3^4*p4 +
    377487360*p3^2*p4^2 + 23592960*p3^2*p4 - 2583429120*p3^2*p5^2 -
    7260733440*p3*p4^2*p5 + 483655680*p3*p4*p5 + 79626240*p3*p5 -
    736542720*p4^4 + 61931520*p4^3 + 24330240*p4^2 - 1038090240*p4*p5^2
    + 684195840*p5^2)*x^8 + (62914560*p3^4*p5 + 361758720*p3^3*p4^2 -
    141557760*p3^2*p4*p5 + 70778880*p3^2*p5 + 121896960*p3*p4^3 +
    29491200*p3*p4^2 - 6794772480*p3*p4*p5^2 + 802160640*p3*p5^2 -
    3169320960*p4^3*p5 + 188743680*p4^2*p5 + 106168320*p4*p5 +
    723517440*p5^3)*x^7 + (272629760*p3^3*p4*p5 + 15728640*p3^3*p5 +
    372244480*p3^2*p4^3 - 3932160*p3^2*p4^2 - 39321600*p3^2*p5^2 -
    507248640*p3*p4^2*p5 + 176947200*p3*p4*p5 - 1929379840*p3*p5^3 -
    14909440*p4^4 + 8847360*p4^3 - 4938792960*p4^2*p5^2 +
    762839040*p4*p5^2 + 106168320*p5^2)*x^6 + (31457280*p3^3*p5^2 +
    471859200*p3^2*p4^2*p5 + 47185920*p3^2*p4*p5 + 210370560*p3*p4^4 -
    11796480*p3*p4^3 - 613416960*p3*p4*p5^2 + 212336640*p3*p5^2 -
    420741120*p4^3*p5 + 88473600*p4^2*p5 - 3428843520*p4*p5^3 +
    644874240*p5^3)*x^5 + (117964800*p3^2*p4*p5^2 + 106168320*p3^2*p5^2
    + 412876800*p3*p4^3*p5 + 5898240*p3*p4^2*p5 + 54558720*p4^5 -
    8110080*p4^4 - 943718400*p4^2*p5^2 + 212336640*p4*p5^2 -
    943718400*p5^4)*x^4 + (15728640*p3^3*p5^2 - 7864320*p3^2*p4^2*p5 +
    10485760*p3^2*p5^3 + 983040*p3*p4^4 + 188743680*p3*p4^2*p5^2 +
```

```
    141557760*p3*p4*p5^2 + 172359680*p4^4*p5 - 35389440*p4^3*p5 -
    765460480*p4*p5^3 + 141557760*p5^3)*x^3 + (23592960*p3^2*p4*p5^2 -
    11796480*p3*p4^3*p5 - 62914560*p3*p4*p5^3 + 141557760*p3*p5^3 +
    1474560*p4^5 + 204472320*p4^3*p5^2 - 35389440*p4^2*p5^2 -
    251658240*p5^4)*x^2 + (47185920*p3^2*p5^3 - 23592960*p3*p4^2*p5^2 -
    62914560*p3*p5^4 + 2949120*p4^4*p5 + 110100480*p4^2*p5^3)*x +
    5242880*p3^3*p5^3 - 3932160*p3^2*p4^2*p5^2 + 983040*p3*p4^4*p5 -
    81920*p4^6 + 20971520*p4*p5^4;

Denom:=
-41943040*x^25 - 209715200*x^23 - 209715200*p3*x^22 + (-209715200*p4 -
    419430400)*x^21 + (-838860800*p3 - 209715200*p5)*x^20 +
    (-419430400*p3^2 - 838860800*p4 - 419430400)*x^19 +
    (-838860800*p3*p4 - 1258291200*p3 - 838860800*p5)*x^18 +
    (-1258291200*p3^2 - 838860800*p3*p5 - 419430400*p4^2 -
    1258291200*p4 - 209715200)*x^17 + (-419430400*p3^3 -
    2516582400*p3*p4 - 838860800*p3 - 838860800*p4*p5 -
    1258291200*p5)*x^16 + (-1258291200*p3^2*p4 - 1258291200*p3^2 -
    2516582400*p3*p5 - 1258291200*p4^2 - 838860800*p4 - 419430400*p5^2
    - 41943040)*x^15 + (-838860800*p3^3 - 1258291200*p3^2*p5 -
    1258291200*p3*p4^2 - 2516582400*p3*p4 - 209715200*p3 -
    2516582400*p4*p5 - 838860800*p5)*x^14 + (-209715200*p3^4 -
    2516582400*p3^2*p4 - 419430400*p3^2 - 2516582400*p3*p4*p5 -
    2516582400*p3*p5 - 419430400*p4^3 - 1258291200*p4^2 - 209715200*p4
    - 1258291200*p5^2)*x^13 + (-838860800*p3^3*p4 - 419430400*p3^3 -
    2516582400*p3^2*p5 - 2516582400*p3*p4^2 - 838860800*p3*p4 -
    1258291200*p3*p5^2 - 1258291200*p4^2*p5 - 2516582400*p4*p5 -
    209715200*p5)*x^12 + (-209715200*p3^4 - 838860800*p3^3*p5 -
    1258291200*p3^2*p4^2 - 1258291200*p3^2*p4 - 5033164800*p3*p4*p5 -
    838860800*p3*p5 - 838860800*p4^3 - 419430400*p4^2 -
    1258291200*p4*p5^2 - 1258291200*p5^2)*x^11 + (-41943040*p3^5 -
    838860800*p3^3*p4 - 2516582400*p3^2*p4*p5 - 1258291200*p3^2*p5 -
    838860800*p3*p4^3 - 1258291200*p3*p4^2 - 2516582400*p3*p5^2 -
    2516582400*p4^2*p5 - 838860800*p4*p5 - 419430400*p5^3)*x^10 +
    (-209715200*p3^4*p4 - 838860800*p3^3*p5 - 1258291200*p3^2*p4^2 -
    1258291200*p3^2*p5^2 - 2516582400*p3*p4^2*p5 - 2516582400*p3*p4*p5
    - 209715200*p4^4 - 419430400*p4^3 - 2516582400*p4*p5^2 -
    419430400*p5^2)*x^9 + (-209715200*p3^4*p5 - 419430400*p3^3*p4^2 -
    2516582400*p3^2*p4*p5 - 838860800*p3*p4^3 - 2516582400*p3*p4*p5^2 -
    1258291200*p3*p5^2 - 838860800*p4^3*p5 - 1258291200*p4^2*p5 -
    838860800*p5^3)*x^8 + (-838860800*p3^3*p4*p5 - 419430400*p3^2*p4^3
    - 1258291200*p3^2*p5^2 - 2516582400*p3*p4^2*p5 - 838860800*p3*p5^3
    - 209715200*p4^4 - 1258291200*p4^2*p5^2 - 1258291200*p4*p5^2)*x^7 +
    (-419430400*p3^3*p5^2 - 1258291200*p3^2*p4^2*p5 - 209715200*p3*p4^4
    - 2516582400*p3*p4*p5^2 - 838860800*p4^3*p5 - 838860800*p4*p5^3 -
    419430400*p5^3)*x^6 + (-1258291200*p3^2*p4*p5^2 -
    838860800*p3*p4^3*p5 - 838860800*p3*p5^3 - 41943040*p4^5 -
```

```
    1258291200*p4^2*p5^2 - 209715200*p5^4)*x^5 + (-419430400*p3^2*p5^3
    - 1258291200*p3*p4^2*p5^2 - 209715200*p4^4*p5 -
    838860800*p4*p5^3)*x^4 + (-838860800*p3*p4*p5^3 -
    419430400*p4^3*p5^2 - 209715200*p5^4)*x^3 + (-209715200*p3*p5^4 -
    419430400*p4^2*p5^3)*x^2 - 209715200*p4*p5^4*x - 41943040*p5^5;

f0:=y1^2-(x1^5+x1^3+x1^2+2*x1+3);
f3:=subs(x=x2,Denom)*subs(x=x1,Eps1)*y1+subs(x=x1,Denom)*subs(x=x2,Eps1)*y2;
f4:=subs(x=x2,Denom)*subs(x=x1,Eps0)*y1+subs(x=x1,Denom)*subs(x=x2,Eps0)*y2;
```

E.5 Le système $j_1j_2j_3$

Le système ci-dessous est issu d'une question de classification des courbes à Jacobienne (2,2)-décomposable. La terminologie est définie dans le chapitre 22 et la motivation du calcul est donnée dans le chapitre 25.

FIG. E.4 – *Le système $j_1j_2j_3$*

Variables	Paramètres	Multiplications	Degrés
4	2	41	(10,10,10,6)

Ce système dépend de deux paramètres (j_3,j_2) et de 4 inconnues (a,b,j,j_1). On cherche le polynôme minimal de j en fonction de (j_1,j_2,j_3), en sachant à l'avance qu'il est de degré 2.

```
Syst:=function(Para,Var)

  j1:=Para[1];
  j2:=Para[2];
  a:=Var[1];
  b:=Var[2];
  j:=Var[3];
  j3:=Var[4];

  t1:= a*b;
  t3:= a^2;
  t4:= b^2;
  t5:= t3*t4;
  t6:= t3*a;
  t8:= t4*b;
  t10:= 27-18*t1-t5+4*t6+4*t8;
  t11:= t10^2;
  t13:= 15+t1;
  t14:= t13^2;
  t15:= t14^2;
```

```
  t21:= 12*t8;
  t22:= 12*t6;
  t33:= t3^2;
  t34:= t33*b;
  t36:= t4^2;
  S1:= j1*t11-2048*t15*t13;
  S2:= j2*t11-512*t14*t13*(t21+t22+t5+405-126*t1);
  S3:= j3*t11-256*t14*(-2583*t1+t21+t22-53*t5+3*t6*t8+14985+20*t34+20*t36*a);
  S4:= t10*j-6912*t8+256*t33*t3-2304*t34+6912*t5;
  return [S1,S2,S3,S4];
end function;
```

E.6 Les j-invariants en fonction d'une forme de Rosenhain

Les équations présentées ici sont utilisées dans la section 25.3 du chapitre 25. Nous résolvons en particulier les systèmes $\mathcal{F}_1$, $\mathcal{F}_2$, $\mathcal{F}_3$, $\mathcal{F}_4$ et $\mathcal{F}_5$ définis par les fonction Magma ci-dessous.
Les variables sont respectivement (c_0,l,n),(c_1,l,n),(j_1,l,n),(j_2,l,n) et (j_3,l,n), les paramètres sont Ψ et Υ, et nous demandons le polynôme minimal de respectivement c_0,c_1,j_1,j_2 et j_3 en fonction des paramètres. Tous les calculs sont faits dans l'ouvert défini par

$$ln(n-l)(l-1)(n-1)(1+n(l-2)) \neq 0.$$

.

FIG. E.5 – *Caractéristiques du système $\mathcal{F}_1$.*

Variables	Paramètres	Multiplications	Degrés
3	2	79	(24,12,8)

```
C0:=function(l,n)
    t553 := l^2;     t557 := t553^2;    t570 := -2*t557;    t569 := -1 + l;
    t568 := -1 + n;  t567 := 1 + (l - 2)*n;                 t550 := t553*l;
    t549 := n^2;
    t566 := 1 + t557 - 2*l - 2*t550 + 3*t553
         + (t570 + 16*t553 - 4 - 8*t550 - 6*l)*n + (20 + 16*t550 - 12*l
         - 21*t553 + 3*t557 + (t570 - 32 - 12*t553 + 48*l - 6*t550)*n
         + (t557 + 16 - 32*l - 4*t550 + 20*t553)*t549)*t549;
    t564 := t567^2;  t562 := t566^2;    t560 := t568^2;    t558 := t569^2;
    nt1:=65536*t562*t566;
    dt1:=(t557*t564^2*t558^2*t560^2);
    return [nt1,dt1];
end function;

C1:=function(l,n)
```

```
    t588 := l^2;    t600 := t588^2 ;   t601 := t600*l;  t574 := t601^2;
    t611 := 6*t574; t610 := -3*t574;  t609 := 1 + (l - 2)*n;
    t607 := t609^2; t599 := t588*l ;   t581 := n^2;     t591 := t581*n;
    t594 := t591^2; t593 := t591*t581; t592 := t581^2;  t585 := t599^2;
    t584 := t600*t599;                 t578 := t600^2;  t576 := t600*t601;
    t572 := -1 + l; t571 := -1 + n;
    nt1 := -512*(t578 + 6*t600 - 3*t584 - 3*t599 + t588 - 7*t601 + 6*t585
         + (-39*t585 + 19*t578 - 11*t584 - 15*t599 - 8*l + 22*t588
         - 37*t600 + 59*t601 - 6*t576)*n + (215*t585 - 54*t578 + 32*l
         + t574 - 126*t601 + 330*t599 - 93*t600 - 34*t584 + 8 - 198*t588
         + 19*t576)*t581 + (-93*t578 - 37*t576 - 3040*t599 - 2992*t601
         + 924*t584 - 768*l + 5856*t600 - 964*t585 + 128 + 2016*t588
         + t611)*t594 + (215*t578 + 688*t588 - 704*l + 192 - 726*t585
         + t611 + 624*t599 + 1748*t601 - 298*t584 - 964*t600 - 39*t576 + (
        688*t585 + 352*t584 + 22*t576 - 198*t578 + t574 - 2592*t601
         + 2016*t600)*t592)*t592 + (-126*t578 + 1748*t585 + 1280*l
         + 2576*t599 - 288*t601 - 7*t574 - 466*t584 - 2992*t600 - 256
         + 59*t576 - 2592*t588
         + (192*t585 - 256*t601 + 8*t578 + 128*t600 - 64*t584)*t593)*t593
         + ((-1152*t584 + 330*t578 + 2576*t601 + t610 + 624*t585
         - 3040*t600 - 15*t576)*t592 + 96*l + 352*t588 - 298*t585
         + 924*t600 + t610 - 34*t578 + 312*t584 - 11*t576 - 466*t601 - 64
         - 1152*t599 +
        (-768*t600 + 96*t584 - 8*t576 - 704*t585 + 32*t578 + 1280*t601)*
        t594)*t591);
    dt1:=(t600*t607^2*t572^2*t571^2);
    return [nt1,dt1];
end function;

J1:=function(l,n)
    t6 := n^2;    t7 := n*t6;     t26 := -4*t7;   t2 := t6^2;
    t4 := l^2;
    t25 := -8*t7 + 8*t2 + (3*t4^2 + 4)*t6
         + (4*t2 + 3 + 4*t7 + 12*t6 - 8*n)*t4
         + ((2*n + t26 - 8*t6)*t4 - 4*n - 8*t2 + t26 + 4*t6)*l;
    t22 := t25^2; t21 := n - 1;   t20 := l - n;   t19 := l - 1;
    t17 := t19^2; t14 := t20^2;   t15 := t20*t14; t12 := t21^2;
    t1 := 1 + (l - 2)*n;
    nt3 := 4*t25*t22^2;
    dt3:=(t7^2*l^2*t12^2*t17^2*t1^2*t15^2);
    return [nt3,dt3];
end function;

J2:=function(l,n)
    t33 := l^2;    t30 := t33^2;   t55 := -2*t30;     t35 := n^2;
    t54 := 3*t35;  t53 := n - 1;   t52 := l - n;      t51 := l - 1;
    t50 := t30*t54 + 3*t33;        t49 := 1 - 2*l + t30;
```

```
    t32 := n*t35;  t34 := l*t33;   t37 := t35^2;
    t48 := 8*t37 - 8*t32 + 4*t35 + (2*n - 4*t32 - 8*t35)*t34 + t50
         + 4*(t37 + t32 + t54 - 2*n)*t33 + 4*(-n - 2*t37 - t32 + t35)*l;
    t46 := t48^2;  t44 := t51^2;   t42 := t52^2;      t40 := t53^2;
    t27 := 1 + (l - 2)*n;
    nt1 := 64*(-2*t34 + (5 - 12*l + 24*t33 - 14*t34)*t35
         + (5*t33 - 4*t34 + t49)*t37
         + (3*l + 9*t34 - 12*t33 + t55 - 2)*t32
         + (9*l - 14*t33 - 4 + 7*t34 + t55)*n + t49 + t50)*t48*t46;
    dt1:=(l^2* t37*t40^2*t44^2*t27^2*t42^2);
    return [nt1,dt1];
end function;

J3:=function(l,n)
    t114 := l^2;      t123 := t114*l;  t111 := t123^2;  t141 := 4*t111;
    t124 := t114^2;   t103 := t124^2;  t140 := 4*t103;  t139 := -8*t103;
    t138 := n - 1;    t137 := l - n;   t136 := l - 1;   t116 := n^2;
    t135 := -4*n*l + 4*t116;           t133 := t136^2;  t131 := t137^2;
    t129 := t138^2;   t118 := t116^2;  t117 := t116*n;  t110 := t123*t114;
    t104 := t123*t124;                 t100 := 1 + (l - 2)*n;
    t99 := 3*t116*t124 - 8*t117 + 8*t118 + (2*n - 4*t117 - 8*t116)*t123
         + 4*(-2*t118 - t117 + t116)*l
         + (4*t118 + 3 + 4*t117 + 12*t116 - 8*n)*t114 + t135;
    nt1 := 64*(12*t124 + 4*t114 + t141 - 8*t123 - 8*t110
         + (-74*t124 - 28*t111 + 4*t104 - 20*t114 + 51*t110 + 43*t123)*n
         + (t140 - 28*t104 - 176*t110 + 63*t114 + 12*l + 100*t111
         + 235*t124 - 126*t123)*t116 + t135 + (-434*t110 - 90*l + 62
         + 235*t111 + 548*t124 + 12*t103 - 402*t123 + 343*t114 - 74*t104
         + (62*t124 - 88*t123 + t141 - 36*l + 74*t114 + 12 - 24*t110)*t118
        )*t118 + ((-4*t104 - 90*t124 - 36 + 12*t111 + 8*t110 + 78*l
         - 170*t114 + 174*t123)*t118 + 334*t110 + 8*l - 176*t111
         + 251*t123 - 166*t114 - 24 + 51*t104 + t139 - 434*t124 + (
        -402*t124 + 251*t110 + 43*t104 + 174*l - 466*t114 + 458*t123
         - 126*t111 + t139 - 88)*t116 + (-170*l + t140 + 74 + 343*t124
         - 466*t123 - 166*t110 + 63*t111 + 426*t114 - 20*t104)*t117)*t117)
        *t99^2;
    dt1:=(l^2*t118*t129^2*t133^2*t100^2*t131^2);
    return [nt1,dt1];
end function;

XXYY:=function(l,n)
    t524 := -2*l;    t517 := l^2;    t516 := t517^2;    t523 := -2*t516;
    t508 := n*l;     t505 := t508 - 2*n + 1;            t513 := n^2;
    t514 := t517*l;
    nt522 := (t516 - 2*t514 + t524 + 1 + 3*t517 + 10*t508
         + (t523 - 16*t517 - 4 + 8*t514)*n + ((10*t514 - 12*t517 + t523)*n
         + 27*t517 - 12*l + 4 + 3*t516 - 16*t514
```

```
        + (t516 - 4*t514 + 4*t517)*t513)*t513);
    dt522:=((-1 + l)^2*(-1 + n)^2);
    nxx := -2*nt522;
    dxx:=dt522*((-l + n)*n);
    nyy := 4*(-l - t517*n + t517 + (t524 + 2 + t517)*t513 + t505)*nt522;
    dyy:=dt522*l^2*t505^2;
    return [nxx,dxx],[nyy,dyy];
end function;

F1:=function(Para,Var)
  XX:=Para[1];
  YY:=Para[2];
  l:=Var[1];
  n:=Var[2];
  c0:=Var[3];
  XX:=Parent(l)!XX;
  YY:=Parent(l)!YY;

  _c0:=C0(l,n);
  _XX,_YY:=XXYY(l,n);
  return [ c0*_c0[2]-_c0[1], XX*_XX[2]-_XX[1],YY*_YY[2]-_YY[1]];
end function;

F2:=function(Para,Var)
  XX:=Para[1];
  YY:=Para[2];
  l:=Var[1];
  n:=Var[2];
  c1:=Var[3];
  XX:=Parent(l)!XX;
  YY:=Parent(l)!YY;

  _c1:=C1(l,n);
  _XX,_YY:=XXYY(l,n);
  return [ c1*_c1[2]-_c1[1], XX*_XX[2]-_XX[1],YY*_YY[2]-_YY[1]];
end function;

F3:=function(Para,Var)
  XX:=Para[1];
  YY:=Para[2];
  l:=Var[1];
  n:=Var[2];
  j1:=Var[3];
  XX:=Parent(l)!XX;
  YY:=Parent(l)!YY;

  _j1:=J1(l,n);
```

```
   _XX,_YY:=XXYY(l,n);
  return [ j1*_j1[2]-_j1[1], XX*_XX[2]-_XX[1],YY*_YY[2]-_YY[1]];
end function;

F4:=function(Para,Var)
  XX:=Para[1];
  YY:=Para[2];
  l:=Var[1];
  n:=Var[2];
  j2:=Var[3];
  XX:=Parent(l)!XX;
  YY:=Parent(l)!YY;

  _j2:=J2(l,n);
  _XX,_YY:=XXYY(l,n);
  return [ j2*_j2[2]-_j2[1], XX*_XX[2]-_XX[1],YY*_YY[2]-_YY[1]];
end function;

F5:=function(Para,Var)
  XX:=Para[1];
  YY:=Para[2];
  l:=Var[1];
  n:=Var[2];
  j3:=Var[3];
  XX:=Parent(l)!XX;
  YY:=Parent(l)!YY;

  _j3:=J3(l,n);
  _XX,_YY:=XXYY(l,n);
  return [ j3*_j3[2]-_j3[1], XX*_XX[2]-_XX[1],YY*_YY[2]-_YY[1]];
end function;
```

E.7 Les systèmes Birkhoff

Les systèmes Birkhoff sont présentés dans le chapitre 26 : leur résolution permet de déterminer les composantes connexes réelles d'une hypersurface algébrique singulière.

FIG. E.6 – *Points réels : problème de Birkhoff-10*

Variables	Paramètres	Multiplications	Degrés
3	1	90	(6,6,5)

```
Birk3_3:=function(Para,Var)
```

```
  e:=Para[1];
  T2:=Var[1];
  T3:=Var[2];
  T4:=Var[3];

  t15 := T3;
  t3 := t15^2;
  t33 := 8*t3;
  t32 := -16+t33;
  t13 := T2;
  t1 := t13^2;
  t31 := 16+4*t1;
  t14 := T4;
  t2 := t14^2;
  t30 := 24-12*t2;
  t16 := t13*t1;
  t29 := -8*t16+t31;
  t17 := t1^2;
  t5 := t13*t17;
  t28 := 8*t5-8*t1+24*t16-2*t17;
  t25 := t3^2;
  t24 := t15*t3;
  t21 := t2^2;
  t20 := t14*t2;
  t11 := t15*t25;
  t8 := t14*t21;
  return [-e+t16^2+3*t24^2+4*t21+4*t17+4*t25+3*t20^2+(9*t21-t17)*t3+
          (-t17+t33+9*t25)*t2+(t25-4*t3+t21+(-4+2*t3)*t2)*t1,
          18*t11-12*t5+(36*t2+t31)*t24+8*(-4+t2+t3-t24)*t16+(18*t21+t29*t2+t28)
          *t15+(-24*t24-16*t11+16*t3-4*t25+(-16*t15-4)*t21+(-32*t24-24*t15
          -t32)*t2)*t13,
          36*t8-54*t11+(-108*t2-48)*t24+(32+72*t3)*t20+(-48*t2+6*t17-54*t21)*
          t15+(-4*t17+36*t25+32*t3)*t14+(8*t20-12*t24+t30*t15+t32*t14)*t1,
          18*t8-18*t5+(36*t3+t31)*t20+(-48+12*t2+12*t3-8*t20)*t16+(18*t25+
          t29*t3+t28)*t14+(24*t2-6*t21-24*t20-16*t8+(-16*t14-6)*t25+
          (-32*t20-24*t14+t30)*t3)*t13];
end function;

Birk3_10:=function(Para,Var)

  e:=Para[1];
  T2:=Var[1];
  T3:=Var[2];
  T4:=Var[3];

  t13 := T3;
  t9 := t13^2;
```

```
  t31 := 4*t9;
  t19 := t13*t9;
  t30 := -4*t19;
  t11 := T2;
  t4 := t11^2;
  t14 := t11*t4;
  t29 := -12*t14;
  t28 := -9-t9;
  t27 := 6*t13-16;
  t12 := T4;
  t6 := t12^2;
  t1 := t12*t6;
  t26 := 8*t1-24*t6;
  t25 := 48*t6-16*t1;
  t24 := 16*t9+t30;
  t23 := 12+t26;
  t20 := t9^2;
  t15 := t4^2;
  t7 := t13*t20;
  t2 := t11*t15;
  return [-e-3*t14^2+t20-3-t19^2-t9+t28*t15+(t20-2*t9-9)*t4+
          (-4*t20-4*t15+t31-4+(t31-8)*t4)*t6,
          36*t2-6*t7+2*(2*t4-8*t6+2)*t19+8*(-t19+4*t6-t28)*t14+2*
          (-t15-8*t2-16*t14-2*t4-1+(t29+4+4*t4)*t6)*t13+2*(t30+4*t7-8*t13
          +t31-2*t20+18+(-8*t9+12*t19+16-12*t13)*t6)*t11,
          18*t7+(-12*t4-12+t25)*t19+(6*t15+6+t23*t4+t26)*t13+(2*t7-16*t20
          +t27*t15+(12*t13-32+t24)*t4+t24+t27)*t12,
          54*t2+(12*t9+108+t25)*t14+(54-6*t20+t23*t9+t25)*t11+
          (-20*t14-10*t2-8*t15-10*t11-16*t4-8+(10*t11-8)*t20+(t29+8*t4+8-
          12*t11)*t9)*t12];
end function;
```

E.8 Le système Bershenko

Le système suivant a été fourni par I. Bershenko [25, 26]. Les variables sont (p,q,x), les paramètres sont (y,z), il faut déterminer le polynôme minimal de x en fonction de (y,z), dans l'ouvert défini par $pq(p+q+1) \neq 0$.

FIG. E.7 – *Le système bershenko .*

Variables	Paramètres	Multiplications	Degrés
3	2	32	(3,4,12)

```
Syst:=function(Para,Var)
```

```
  y:=Para[1];
  z:=Para[2];
  p:=Var[1];
  q:=Var[2];
  x:=Var[3];


  t1:=p*q;
  t2:=p^2;
  t3:=q^2;

  t4:=(t3-2*q+1);
  t5:=(t2-2*p+1);
  t6:=(t3+2*q+1);
  t7:=(t2+2*p+1);
  t8:=p*t2;
  t9:=q*t3;
  t10:=t6+t7-1+2*t1;
  P2:= p*t4+q*t5-2*t1+t3+t2-3*x*p*q;
  P3 := -(t4-t2)*(q+1-p)-9*y*t1;
  P4 := t6*t7*(q+p+2*t2+2*t3+(1+q)*t8+(1+p)*t9-t1^2)^2-36*z*t2*t3*t10^2;
  return [P2,P3,P4];
end function;
```

E.9 Le système Hawes4

Cette annexe décrit les équations du système Hawes4 4, obtenu sur le serveur PolyData [15]. Les paramètres sont (a,b), les variables sont (c,x,y_1,z_1,y_2,z_2), il faut déterminer le polynôme minimal de c en fonction de a et b, en-dehors de l'hypersurface définie par $(y_1 - y_2)(z_1 - z_2)$.

Fig. E.8 – *Le système Hawes4*

Variables	Paramètres	Multiplications	Degrés
6	2	55	(2,2,4,4,5,9)

Le système était donné sous forme développée, avec une complexité d'évaluation double de celle ci-dessus. Après un peu de travail, nous avons reconstitué la forme suivante. Il fait peu de doutes que la forme initiale du système en était proche.

```
Syst:=function(Para,Var)

  a:=Para[1];
  b:=Para[2];
```

```
  c:=Var[1];
  x:=Var[2];
  z1:=Var[3];
  y2:=Var[4];
  z2:=Var[5];
  y1:=Var[6];

 return [3*z1^2+y1^2+b,
         y2^2+3*z2^2+b,
         2*c*y1+x+2*z1*y1+5*y1^4+3*a*y1^2,
         2*c*y2+x+5*y2^4+3*a*y2^2+2*y2*z2,
         -c*y2^2+c*y1^2-x*y2+x*y1+z1^3+z1*y1^2+b*z1-y2^5
         -a*y2^3-y2^2*z2-z2^3-b*z2+y1^5+a*y1^3,
         -6*c*y1^2*z1*(3*x*z2-45*y2^4*z2+y2^3-3*y2*z2^2-b*y2)+
         6*c*y2^2*z2*(3*x*z1-3*z1^2*y1-45*z1*y1^4+y1^3-b*y1)-
         2*(3*z1^2+30*z1*y1^3+9*a*z1*y1-y1^2)*
         (-9*a*y2^2*z2+3*x*z2-45*y2^4*z2+y2^3-3*y2*z2^2-b*y2)
         +2*(30*y2^3*z2-y2^2+9*a*y2*z2+3*z2^2)*
         (-9*a*z1*y1^2+3*x*z1-3*z1^2*y1-45*z1*y1^4+y1^3-b*y1)];
end function;
```

Table des matières

Bibliographie

[1] Fgb. http://posso.lip6.fr/ jcf/.

[2] Magma. http://www.maths.usyd.edu.au:8000/u/magma/.

[3] Medicis. http://www.medicis.polytechnique.fr/.

[4] Page d'accueil de tera. http://tera.medicis.polytechnique.fr/.

[5] J. Abdeljaoued. *Algorithmes rapides pour le Calcul du Polynôme Caractéristique.* PhD thesis, Université de Franche Compté, Besançon, France, 1997.

[6] S.A. Abramov and K.Y. Kvashenko. On the greatest common divisor of polynomials which depend on a parameter. In *Proceedings ISSAC'93.* ACM Press, 1993.

[7] A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The design and analysis of computer algorithms.* Addison-Wesley, 1974.

[8] E. Allgower and K. Georg. Continuation and path following. *Acta Numerica*, pages 1–64, 1993.

[9] M. E. Alonso, E. Becker, M.-F. Roy, and T. Wörmann. Zeroes, multiplicities and idempotents for zerodimensional systems. In *Algorithms in Algebraic Geometry and Applications. Proceedings MEGA'94*, volume 142 of *Progress in Mathematics*, pages 1–15. Birkhäuser, 1996.

[10] A. Assi. On flatness of generic projections. *Journal of Symbolic Computation*, 18(5):447–462, 1994.

[11] P. Aubry. *Ensembles triangulaires de polynômes et résolution de systèmes algébriques. Implantation en Axiom.* PhD thesis, Université Paris VI, 1999.

[12] P. Aubry, D. Lazard, and M. Moreno Maza. On the theories of triangular sets. *Journal of Symbolic. Computation*, 28(1,2):45–124, 1999.

[13] P. Aubry and M. Moreno Maza. Triangular sets for solving polynomial systems : a comparative implementation of four methods. *Journal of Symbolic Computation*, 1999.

[14] P. Aubry, F. Rouillier, and M. Safey. Real solving positive for positive dimensional systems. Technical report, LIP6, Université Paris VI, 2000.

[15] O. Bachmann, M. Dengel, J.-C. Faugère, H.-G. Graebe, and H. Schoenemann. The polydata project. http://polydata.medicis.polytechnique.fr/.

[16] B. Bank, M. Giusti, J. Heintz, and G. Mbakop. Polar varieties and efficient real equation solving: The hypersurface case. *Journal of Complexity*, 13(1):5–27, 1997.

[17] B. Bank, M. Giusti, J. Heintz, and M. Mbakop. Polar varieties and efficient real elimination. *Mathematische Zeitschrift*, 2000 (to appear).

[18] S. Basu, R. Pollack, and M.-F. Roy. On the combinatorial and algebraic complexity of quantifier elimination. *J. Assoc. Comput. Machin.*, 43:1002–1045, 1996.

[19] S. Basu, R. Pollack, and M.-F. Roy. *Quantifier elimination and Cylindrical algebraic decomposition*, chapter A new algorithm to find a point in every cell defined by a family of polynomials, pages 341–349. Texts and Monographs in Symbolic Computation. Springer-Verlag, 1998.

[20] W. Baur and V. Strassen. The complexity of partial derivatives. *Theoret. Comp. Sci.*, 22:317–330, 1983.

[21] B. Beckermann and G. Labahn. A uniform approach for Hermite Padé and simultaneous Padé approximants and their matrix-type generalizations. *Numerical Algorithms*, 3:45–54, 1992.

[22] B. Beckermann and G. Labahn. A uniform approach for the fast computation of matrix-type Padé approximants. *SIAM Journal on Matrix Analysis and Applications*, 15:804–823, 1994.

[23] M. Ben-Or and P. Tiwari. A deterministic algorithm for sparse multivariate polynomial interpolation. In *20th Annual ACM Symp. Theory Comp.*, 1988.

[24] S. J. Berkowitz. On computing the determinant in small parallel time using a small number of processors. *Information Processing Letters*, 18:147–150, 1984.

[25] I. Bershenko-Kogan. *Inductive approache to Cartan's Moving Frame Method with applications to classical invariant theory.* PhD thesis, University of Minnesota, 2000.

[26] I. Bershenko-Kogan and P.J. Olver. Symmetries of polynomials. *Journal of Symbolic Computation*, 29:485–515, 2000.

[27] W.-J. Beyn. On smoothness and invariance properties of the Gauss-Newton method. *Num. Funct. Anal. and Opt.*, 14:243–252, 1993.

[28] D. Bini and V.Y. Pan. *Polynomial and matrix computations.* Progress in Theoretical Computer Science. Birkhäuser, 1994.

[29] L. Blum, P. Cucker, M. Shub, and S. Smale. *Complexity and real computation.* Springer-Verlag, 1998.

[30] O. Bolza. On binary sextics with linear transformations onto themselves. *Amer. J. Math.*, 10:47–70, 1888.

[31] D. Bondyfalat, B. Mourrain, and V.Y. Pan. Controlled iterative methods for solving polynomial systems. In *Proceedings ISSAC'98*, pages 252–259. ACM Press., 1998.

[32] W. Bosma, J. Cannon, and C. Playoust. The Magma algebra system I: The user language. *Journal of Symbolic Computation*, 24, 1997.

[33] J.-B. Bost, H. Gillet, and C. Soulé. Un analogue arithmétique du théorème de Bézout. *C. R. Acad. Sci. Paris*, 312:845–848, 1991.

[34] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot. Representation for the radical of a finitely generated differential ideal. In *Proceedings ISSAC '95*, pages 158–166. ACM, ACM Press, 1995.

[35] B. Buchberger. Gröbner bases: An algoritmic method in polynomial ideal theory. In *Multidimensional System Theory*, pages 374–383. Reidel, Dordrecht, 1985.

[36] B. Buchberger, G.E. Collins, and R. Loos, editors. *Computer Algebra.* Springer Verlag, 1982.

[37] P. Bürgisser, M. Clausen, and M. A. Shokrollahi. *Algebraic Complexity Theory.* Springer, 1997.

[38] J. Cannon and C. Playoust. Magma: A new computer algebra system. *Euromath Bulletin*, 2(1):113–144, 1996.

[39] J. Canny. Some algebraic and geometric problems in PSPACE. In *Proceedings 20 ACM STOC*, pages 460–467, 1988.

[40] J. Canny and I. Emiris. An efficient algorithm for the sparse mixed resultant. In *Applied Algebra, Algebraic Algorithms and Error Correcting Codes. Proce AAECC-10*, volume 673 of *Lecture Notes in Computer Science*, pages 89–104. Springer-Verlag, 1993.

[41] J. Canny, E. Kaltofen, and Y. Lakshman. Solving systems of non-linear polynomial equations faster. In *Proceedings ISSAC '89*. ACM Press, 1989.

[42] D. G. Cantor. Computing in the Jacobian of an hyperelliptic curve. *Math. Comp.*, 48(177):95–101, 1987.

[43] D. G. Cantor. On the analogue of the division polynomials for hyperelliptic curves. *J. Reine Angew. Math.*, 447:91–145, 1994.

[44] É. Cartan. *La méthode du repère mobile, la théorie des groupes continus et les espaces généralisés.* exposés de Géométriie No. 5. Hermann, Paris, 1935.

[45] D. Castro, K. Hägele, J.E. Morais, and L.M. Pardo. Kronecker and Newton approaches to solving: a first comparison. *To appear in J. of Complexity*, 2000.

[46] L.S. Charlap, R. Coley, and D. Robbins. Enumeration of rational points on elliptic curves over finite fields. Prépublication, 1991.

[47] J.S.R. Chisholm73. Rational approximants defined from doule power series. *Math. of Comp.*, 27:841–848, 1973.

[48] A. L. Chistov and D. Y. Grigoriev. Polynomial-time factoring of multivariable polynomials over a global field. LOMI preprint E-5-82, Steklov Institute, Leningrad, 1982.

[49] H. Cohen. *Advanced Topics in computational number theory.* Number 193 in Graduate Texts in Mathematics. Springer-Verlag, 2000.

[50] S. A. Cook. *On the minimum computation times of functions.* PhD thesis, Harvard University, 1966.

[51] J. Cooley and J. Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation*, 19, 1965.

[52] D. Coppersmith and S. Winograd. Matrix multiplications via arithmetic progression. In *19th ACM STOC*, pages 1–6, 1987.

[53] R. Corless, P. Gianni, B. Trager, and S. Watt. The singular value decomposition for polynomial systems. In *Proceedings ISSAC '95*, pages 195–207, 1995.

[54] A. Cuyt. Multivariate Padé approximants. *J. Math. Anal. Appl.*, 96:283–293, 1983.

[55] J.-P. Dedieu and M.-H. Kim. Newton's method for analytic systems of equations with constant rank derivatives. Technical report, Laboratoire MIP, Toulouse, 1999.

[56] J.-P. Dedieu and M. Shub. Newton's method for overdetermined systems of equations. *Mathematics of Computation*, 1998.

[57] S. Dellière. *Triangularisation de systèmes constructibles — Application à l'évaluation dynamique.* PhD thesis, Université de Limoges, 1999.

[58] S. Dellière. Pgcd de deux polynômes à paramètres : approche par la clôture constructible dynamique et généralisation de la méthode de S.A. Abramov, K.Yu. Kvashenko. Technical report, INRIA RR-3882, 2000.

[59] H. Derksen. An algorithm to compute generalized Padé-Hermite forms. Disponible par ftp à daisy.math.unibas.ch dans /pub/hderksen/pade.dvi, 1994.

[60] J. Dixon. Exact solution of linear equations using p-adic expansions. *Numer. Math.*, 40:137–141, 1982.

[61] D. Duval. *Diverses questions relatives au calcul formel avec des nombres algébriques.* PhD thesis, Université scientifique, technologique et médicale de Grenoble, 1987.

[62] D. Duval. Algebraic numbers : an example of dynamic evaluation. *J. Symb. Comp.*, 18:429–447, 1994.

[63] G. L. Ebert. Some comments on the modular approach to Gröbner bases. *ACM SIGSAM Bull.*, 17(2):28–32, 1983.

[64] S. Egner. Semi-numerical solution to 6/6-Stewart platform kinematics based on symmetry. *Applicable Algebra in Engineering, Communication and Computing*, 7:449–468, 1996.

[65] D. Eisenbud. *Commutative Algebra with a view toward Algebraic Geometry.* Number 150 in Graduate Texts in Mathematics. Springer, 1996.

[66] I. Z. Emiris and B. Mourrain. Matrices in elimination theory. *Journal of Symbolic Computation*, 28(1-2), 1999.

[67] I.Z. Emiris, A. Galligo, and H. Lombardi. Numerical univariate polynomial gcd. In *Proc. AMS-SIAM Summer seminar on Math. of Numerical analysis*, Lectures in Applied Mathematics, 1995.

[68] I.Z. Emiris, A. Galligo, and H. Lombardi. Certified approximate univariate gcds. In *Proc. 4th Effective Methods in Algebraic Geometry (MEGA)*, 1997.

[69] J.C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *Journal of Symbolic Computation*, 16(4):329–344, 1993.

[70] Jean-Charles Faugre. A new efficient algorithm for computing gröbner bases (F4). In *Proceedings of MEGA'98*, 1998.

[71] P. Fitzpatrick and J. Flynn. A Gröbner basis technique for Padé approximation. *Journal of Symbolic Computation*, 13:133–138, 1992.

[72] G. Gallo and B. Mishra. Efficient algorithms and bounds for Wu-Ritt characteristic sets. In C. Traverso and T. Mora, editors, *Proceedings Mega '90*, volume 94 of *Progr. Math.* Birkhäuser, 1990.

[73] P. Gaudry. *Courbes hyperelliptiques et cryptographie.* PhD thesis, École polytechnique, 2000.

[74] P. Gaudry and R. Harley. Counting points on hyperelliptic curves over finite fields. In *Proceedings ANTS IV*, number 1838 in Lecture Notes in Computer Science, 2000.

[75] P. Gaudry and É. Schost. Invariants des quotients de la Jacobienne d'une courbe de genre 2. Technical report, http://www.medicis.polytechnique.fr/gage/notes/2000.html, 2000.

[76] P. Gianni. Properties of Gröbner bases under specializations. In *European Conference on Computer Algebra*, number 378 in Lecture Notes in Computer Science, pages 293–297, 1987.

[77] P. Gianni, E. Fortuna, and B. Trager. Degree reduction under specialization. Exposé à MEGA 2000, 2000.

[78] P. Gianni and T. Mora. Algebraic solution of systems of polynomial equations using Gröbner bases. In *Applied Algebra, Algebraic Algorithms and Error Correcting Codes, Proceedings AAECC-5*, volume 356 of *Lecture Notes in Computer Science*, pages 247–257. Springer, 1989.

[79] M. Giusti, K. Hägele, J. Heintz, J. E. Morais, J. L. Montaña, and L. M. Pardo. Lower bounds for Diophantine approximation. In *Proceedings MEGA'96*, number 117,118 in Journal of Pure and Applied Algebra, pages 277–317, 1997.

[80] M. Giusti and J. Heintz. La détermination des points isolés et de la dimension d'une variété algébrique peut se faire en temps polynomial. In D. Eisenbud and L. Robbiano, editors, *Computational Algebraic Geometry and Commutative Algebra*, volume XXXIV of *Symposia Matematica*, pages 216–256. Cambridge University Press, 1993.

[81] M. Giusti, J. Heintz, J. E. Morais, J. Morgenstern, and L. M. Pardo. Straight-line programs in geometric elimination theory. *J. of Pure and App. Algebra*, 124:101–146, 1998.

[82] M. Giusti, J. Heintz, J. E. Morais, and L. M. Pardo. When polynomial equation systems can be solved fast? In G. Cohen, M. Giusti, and T. Mora, editors, *Applied Algebra, Algebraic*

Algorithms and Error Correcting Codes, Proceedings AAECC-11, volume 948 of *Lecture Notes in Computer Science*, pages 205–231. Springer, 1995.

[83] M. Giusti, J. Heintz, J. E. Morais, and L. M. Pardo. Le rôle des structures de données dans les problèmes d'élimination. *C. R. Acad. Sci. Paris*, 325:1223–1228, 1997.

[84] M. Giusti, G. Lecerf, and B. Salvy. A Gröbner free alternative for polynomial system solving. *À paraître dans les actes de la conférence FOCM '99*, 1999.

[85] M. Giusti and É. Schost. Solving some overdetermined polynomial systems. In *Proceedings ISSAC'99*, pages 1–8. ACM Press., 1999.

[86] T. Gomez-Diaz. *Quelques applications de l'évaluation dynamique.* PhD thesis, Université de Limoges, 1994.

[87] L. Gonzalez-Vega. *Computer Algebra in Science and Engineering*, chapter Some examples of problem solving by using the symbolic viewpoint when dealing with polynomial systems of equations, pages 102–116. World Scientific Publishing, 1995.

[88] L. Gonzalez-Vega. Applying quantifier elimination to the Birkhoff interpolation problem. *Journal of Symbolic Computation*, 1996.

[89] H.-G. Gräbe. On lucky primes. *Journal of Symbolic Computation*, 15:199–209, 1993.

[90] D. Grigoriev and N. Vorobjov. Bounds on numbers of vectors of multiplicities for polynomials which are easy to compute. In C. Traverso, editor, *Proceedings ISSAC 2000*, pages 137–145. ACM Press, 2000.

[91] D.Y. Grigoriev and N. Vorobjov. Solving systems of polynomial inequalities in subexponential time. *J. Symbolic Comput.*, 5:37–64, 1988.

[92] P. Guillaume and A. Huart. Multivariate Padé approximation. Technical report, Laboratoire MIP, Toulouse, 1999.

[93] K. Hägele. *Intrinsic height estimates for the Nullstellensatz.* PhD thesis, Universidad de Cantabria, Santander, 1998.

[94] K. Hägele and J.L. Montaña. Polynomial random test for the equivalence problem of integers given by arithmetic circuits. 1997.

[95] K. Hägele, J.E. Morais, L.M. Pardo, and M. Sombra. On the intrinsic complexity of the arithmetic Nullstellensatz. *J. Pure Appl. Algebra*, 146:103–183, 2000.

[96] K. Han, W. Chung, and Y. Youm. New resolution scheme of the forward kinematics of parallel manipulators using extra sensor data. *J. of Mechanical Design*, 118(2):214–219, 1996.

[97] R. Hartshorne. *Algebraic Geometry.* Number 52 in Graduate Texts in Mathematics. Springer-Verlag, 1977.

[98] J. Heintz. Definability and fast quantifier elimination in algebraically closed fields. *Theor. Comput. Sci.*, 24(3):239–277, 1983.

[99] J. Heintz. On the computational complexity of polynomials and bilinear mappings. A survey. In *Applied Algebra, Algebraic Algorithms and Error Correcting Codes, Proceedings of AAECC-5*, volume 356 of *LNCS*, pages 269–300. Springer, 1989.

[100] J. Heintz, T. Krick, S. Puddu, J. Sabia, and A. Waissbein. Deformation techniques for efficient polynomial equation solving. *Journal of Complexity*, 16(1), 2000.

[101] J. Heintz, M.-F Roy, and P. Solernó. On the complexity of semi-algebraic sets. In *Proc. IFIP 89, San Francisco*, pages 293–298. North-Holland, 1989.

[102] J. Heintz, M.-F Roy, and P. Solernó. On the theoretical and practical complexity of the existential theory of the reals. *Comput. J.*, 36(5):427–431, 1993.

[103] J. Heintz and C. P. Schnorr. Testing polynomials which are easy to compute. In *Proccedings of ACM 12th Symposium on Theory of Computing*, pages 262–272, 1980.

[104] H.Kobayashi, S. Moritsugu, and R. W. Hogan. On solving systems of algebraic equations. In P. Gianni, editor, *Proceedings ISSAC '88*, number 358 in Lecture Notes in Computer Science, pages 139–149, 1988.

[105] E. Howe, F. Leprévost, and B. Poonen. Large torsion subgroups of split Jacobians of curves of genus two or three. *Forum Math.*, 12:315–364, 2000.

[106] Y. Huang, H. Stetter, W. Wu, and L. Zhi. Pseudofactors of multivariate polynomials. In *Proceedings ISSAC 2000*, pages 161–168. ACM Press, 2000.

[107] O. H. Ibarra and S. Moran. Equivalence of straight-line programs. *Journal of the ACM*, 30:217–228, 1983.

[108] J. Igusa. Arithmetic variety of moduli for genus two. *Ann. of Math.*, 72:612–649, 1960.

[109] C. Dicrescenzo J. Della Dora and D. Duval. About a new method for computing in algebraic number fields. In *Proceedings Eurocal '85 Vol. 2*, volume 204 of *Lecture Notes in Computer Science*, pages 289–290, 1985.

[110] M. Kalkbrener. Solving systems of algebraic equations by using Gröbner bases. In *European Conference on Computer Algebra*, number 378 in Lecture Notes in Computer Science, pages 282–292, 1987.

[111] M. Kalkbrener. *Three contributions to elimination theory.* PhD thesis, Kepler University, Linz, 1991.

[112] M. Kalkbrener. On stability of Gröbner bases under specializations. *Journal of Symbolic Computation*, 24(1):51–58, 1997.

[113] E. Kaltofen. Greatest common divisors of polynomials given by straight-line programs. *Journal of the ACM*, 35(1):231–264, January 1988.

[114] E. Kaltofen and Y. Lakshman. Improved sparse multivariate polynomial interpolation algorithms. In *Proceedings ISSAC 1988.* ACM Press, 1988.

[115] E. Kaltofen and H. Rolletschek. Computing greatest common divisors and factorizations in quadratic number fields. *Mathematics of Computation*, 53:697–720, 1989.

[116] W. Kampkötter. *Explizite Gleichungen für Jacobische Varietäten hyperelliptischer Kurven.* PhD thesis, Universität Gesamthochschule Essen, August 1991.

[117] A. Karatsuba and Yu. Ofman. Multiplication of multidigit numbers on automata. *Soviet Physics-Doklady*, 7:595–596, 1963.

[118] M-H. Kim. *Computational Complexity of the Euler type Algorithm for the Roots of Polynomials.* PhD thesis, CUNY, 1986.

[119] D. E. Knuth. *The Art of Computer Programming II: Semi-numerical Algorithms.* Addison-Wesley, 1998.

[120] H. Kobayashi, T. Fujise, and A. Furukawa. On solving systems of algebraic equations by general elimination method. *Journal of symbolic computation*, 5:303–320, 1988.

[121] N. Koblitz. *Algebraic Aspects of Cryptography.* Springer-Verlag, 1998.

[122] J. Kollár. Sharp effective Nullstellensatz. *J. Amer. Math. Soc*, 1:963–975, 1988.

[123] T. Krick and L. M. Pardo. A computational method for Diophantine approximation. In L. González-Vega and T. Recio, editors, *Algorithms in Algebraic Geometry and Applications. Proceedings of MEGA'94*, volume 143 of *Progress in Mathematics*, pages 193–254. Birkhäuser Verlag, 1996.

[124] T. Krick, L. M. Pardo, and M. Sombra. Sharp estimates for the arithmetic Nullstellensatz. In *Proceeings EACA '99*, 1999.

[125] T. Krick, J. Sabia, and P. Solernó. On intrinsic bounds in the Nullstellensatz. *AAECC Journal*, 8:125–134, 1197.

[126] L. Kronecker. Zur Theorie der Elimination einer Variabeln aus zwei algebraishen Gleichungen. *Monatsberichte der Königlich Preussichen Akademie der Wissenschaften*, pages 535–600, 1881.

[127] L. Kronecker. Grundzüge einer arithmetischen Theorie der algebraischen Grössen. *J. reine angew. Math.*, 92:1–122, 1882.

[128] E. Kruppa. Zur Ermittlung eines Objektes aus zwei Perpesktiven mitinnerer Orientierung. *Sitz.-Ber. Akad. Wiss., Wien. Math. Naturw. Kl.*, 1913.

[129] L. Kulesz. Courbes algébriques de genre 2 possédant de nombreux points rationnels. *C. R. Acad. Sci. Paris*, 321:91–94, 1995.

[130] L. Kulesz. Application de la méthode de Dem'janenko-Manin à certaines familles de courbes de genre 2 et 3. *J. Number Theory*, 76:130–146, 1999.

[131] H. T. Kung. On computing the reciprocals of power series. *Numer. Math.*, 22:341–348, 1974.

[132] Y. N. Lakshman and D. Lazard. On the complexity of zero-dimensional algebraic systems. In *Effective methods in algebraic geometry*, volume 94 of *Progress in Mathematics*, pages 217–225. Birkhäuser, 1991.

[133] C. Lanczos. Solutions of systems of linear equations by minimized iterations. *Journal of Research of the National Bureau of Standards*, 49:33–53, 1952.

[134] S. Lang. *Algebra, 3rd edition.* Addison Wesley, 1993.

[135] H. Lange. Über die Modulvarietät der Kurven vom Geschlecht 2. *J. Reine Angew. Math*, 281:80–96, 1976.

[136] L. Langemyr. Algorithms for a multiple algebraic extension. In T. Mora and C. Traverso, editors, *Effective Methods in Algebraic Geometry, Proceedings MEGA 90.* Birkhäuser, 1990.

[137] D. Lazard. A new method for solvong algebraic systems of positive dimension. *Disc. Appl. Math.*, 33:147–160, 1991.

[138] D. Lazard. Solving zero-dimensional algebraic systems. *Journal of symbolic computation*, 13:117–133, 1992.

[139] D. Lazard. Stewart platform and Gröbner bases. In Parenti-Castelli V. and Lenaric J., editors, *Proceedings of the third international workshop on advances of robot kinematics*, pages 136–142. Ferrare, 1992.

[140] G. Lecerf. *Kronecker, a package for Magma for polynomial system solving.* UMS MEDICIS, Laboratoire GAGE, http://kronecker.medicis.polytechnique.fr/, 1999.

[141] G. Lecerf. Communication privée, 2000.

[142] D. H. Lehmer. Factorization of certain cyclotomic functions. *Ann. of Math.*, 34:461–479, 1933.

[143] D. H. Lehmer. Euclid's algorithm for large integers. *The American Mathematical Monthly*, 45:227–233, 1938.

[144] P. Lelong. Mesure de Mahler et calcul des constantes universelles pour les polynômes de n variables. *Math. Ann.*, 299:673–695, 1994.

[145] A.K. Lenstra, H.W. Lenstra Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261:515–534, 1982.

[146] F. Leprévost and F. Morain. Revêtements de courbes elliptiques multiplication complexe par des courbes hyperelliptiques et sommes de caractères. *J. Number Theory*, 64:165–182, 1997.

[147] U. J. J. Leverrier. Sur les variations séculaires des éléments elliptiques des sept planètes principales: Mercure, Venus, la terre, Mars, Jupiter, Saturne et Uranus. *J. Math. Pures Appli.*, 4:220–254, 1840.

[148] D. Levin. Padé-type and Padé approximants defined from double power series. *J. Inst. Math. Appl.*, 18:395–407, 1976.

[149] J. D. Lipson. Chinese remainder algorithm and interpolation algorithms. In S. R. Petrick, editor, *Proceedings 2nd ACM Symposium of Symbolic and Algebraic Manipulation.* ACM Press, 1971.

[150] F. S. Macaulay. *The Algebraic Theory of Modular Systems.* Cambridge University Press, 1916.

[151] http://www.maplesoft.com/.

[152] S. Maybank and O. Faugeras. A theory of self-calibration of a moving camera. *International Journal of Computer Vision*, 8(2):123–151, 1992.

[153] M. Moreno Maza. *Calculs de pgcd au-dessus des tours d'extensions simples et résolution des systèmes d'équations algébriques.* PhD thesis, Université Paris VI, 1997.

[154] M. Moreno Maza and R. Rioboo. Polynomial gcd computations over tower of algebraic extensions. In G. Cohen, M. Giusti, and T. Mora, editors, *Proceedings AAECC-11*, number 948 in Lecture Notes in Computer Science, pages 365–382. Springer, 1995.

[155] G. M. Mbakop. *Effiziente Lözung reeller Polynomialer Gleichungsysteme.* PhD thesis, Humbold-Universität, Berlin, 1999.

[156] P.J. McCarthy. *Algebraic Extensions of fields.* Dover Publications, 1966.

[157] J.P. Merlet. Parallel manipulators - 3rd part. Technical report, Inria 1003, 1988.

[158] J.-F. Mestre. Construction de courbes de genre 2 à partir de leurs modules. In C. Traverso and T. Mora, editors, *Proceedings Mega 90*, volume 94 of *Progr. Math.* Birkhäuser, 1990.

[159] M. Mignotte. *Mathématiques pour le calcul formel.* Presses Universitaires de France, 1989.

[160] J. E. Morais. *Resolución eficaz de sistemas de ecuaciones polinomiales.* PhD thesis, Universidad de Cantabria, Santander, Spain, 1997.

[161] S. Morrison. The differential ideal $[P] : M^{\infty}$. *J. Symb Comp.*, 28:631–656, 1999.

[162] B. Mourrain. Enumeration problems in Geometry, Robotics and Vision. In L. González-Vega and T. Recio, editors, *Algorithms in Algebraic Geometry and Applications*, volume 143 of *Prog. in Math.*, pages 285–306. Birkhäuser, Basel, 1996.

[163] B. Mourrain and V. Pan. Multivariate polynomials, duality and strucured matrices. *J. of Complexity*, 16(1), 2000.

[164] D. Mumford. *Algebraic Geometry 1: complex algebraic varieties.* Classics in Mathematics. Springer, 1976.

[165] D. Mumford. *Tata Lectures on Theta II.* Birkhäuser, 1984.

[166] R. Narasimham. *Introduction to the theory of analytic spaces.* Number 25 in Lecture notes in Mathematics. Springer-verlag, 1966.

[167] R. Nauheim. Systems of algebraic equations with bad reduction. *Journal of Symbolic Computation*, 25:619–641, 19998.

[168] H. Padé. Sur la représentation approchée d'une fonction par des fractions rationnelles. *Annales scientifiques de l'École Normale Supérieure*, 9, 1892.

[169] V.Y. Pan. Parametrization of Newton's iteration for computations with structured matrices and applications. *Computers and Mathematics (with Applications)*, 24(3), 1992.

[170] V.Y. Pan. Approximate polynomials gcds, Padé approximation, polynomial zeros and bipartite graph. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[171] L. M. Pardo. How lower and upper complexity bounds meet in elimination theory. In G. Cohen, M. Giusti, and T. Mora, editors, *Applied algebra, algebraic algorithms and error-correcting codes (Paris, 1995)*, volume 948 of *Lecture Notes in Computer Science*, pages 33–69. Springer, Berlin, 1995.

[172] F. Pauer. On lucky ideals for Gröbner basis computation. *Journal of Symbolic Computation*, 14(5):471–482, 1992.

[173] P. Philippon. Critères pour l'indépendance algébrique. *Publ. Math. IHES*, 64:5–52, 1986.

[174] P. Philippon. Sur des hauteurs alternatives 1. *Math. Ann.*, 289:255–283, 1991.

[175] P. Philippon. Sur des hauteurs alternatives 2. *Ann. Inst. Fourier*, 44:1043–1065, 1994.

[176] P. Philippon. Sur des hauteurs alternatives 3. *J. Math. Pures Appl.*, 74:345–365, 1995.

[177] J. Pila. Frobenius maps of abelian varieties and finding roots of unity in finite fields. *Math. Comp.*, 55(192):745–763, October 1990.

[178] F. Gustavson R. Brendt and D. Yun. Fast solution of Toeplitz systems of equations and computation of Padé approximants. *Journal of Algorithms*, 1:259–295, 1980.

[179] J. Renegar. On the computational complexity and geometry of the first-order theory of the reals. Part I. *Journal of Symbolic Computation*, 13(3):255–299, March 1992.

[180] J.F. Ritt. Differential equations from an algebraic standpoint. *American Mathematical Society Colloquium Publications*, 14, 1932.

[181] J.F. Ritt. *Differential Algebra.* Dover Publications, 1966.

[182] J.M. Rojas. Computing complex dimension faster and deterministically. Technical report, Department of Mathematics, City University of Hong Kong, 2000.

[183] F. Ronga and T. Vust. Stewart platforms without computer? In F. Broglia, M. Galbiati, and A. Tognoli, editors, *International Conference on Real Analytic and Algebraic Geometry.* W. de Gruyter, 1995.

[184] F. Rouillier. *Algorithmes efficaces pour l'étude des zéros réels des systèmes polynomiaux.* PhD thesis, Université de Rennes I, may 1996.

[185] F. Rouillier. Solving zero-dimensional systems through the Rational Univariate Representation. *Applicable Algebra in Engineering, Communication and Computing*, 9(5):433–461, 1999.

[186] F. Rouillier, M. Safey el Din, and É. Schost. Solving the Birkhoff interpolation problem via the critical point method : an experimental study. Technical report, 2000.

[187] F. Rouillier, M.F. Roy, and M. Safey el Din. Finding at least one point in each connected component of a real algebraic set defined by a single equation. *Journal of Complexity*, 1999.

[188] M. F. Roy. Basic algorithms in real algebraic geometry : from sturm theorem to the existential theory of reals. In *Lectures of real geometry in memoriam of Mario Raimondo*, volume 23 of *Expositions in Mathematics*, 1996.

[189] D. Rupprecht. An algorithm for computing certified approximate GCD of n univariate polynomials. In *Effective Methods in algebraic geometry*, volume 139 of *Journal of Pure and Applied Algebra*, pages 255–284, 1999.

[190] J. Sabia and P. Solernó. Bounds for traces in complete intersections and degrees in the Nullstellensatz. *Applicable Algebra in Engineering Communications and Computing (AAECC Journal)*, 6:353–376, 1996.

[191] A. Schönhage. Quasi-gcd computations. *Journal of Complexity*, 1:118–137, 1985.

[192] A. Schönhage and V. Strassen. Schnelle Multiplikation großer Zahlen. *Computing*, 7:281–292, 1971.

[193] R. Schoof. Elliptic curves over finite fields and the computation of square roots mod p. *Math. Comp.*, 44:483–494, 1985.

[194] J. T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27(4):701–717, October 1980.

[195] M. Sieveking. An algorithm for division of power series. *Computing*, 10:153–156, 1972.

[196] S. Smale. Algorithms for Solving Equations. In *Proceedings of the International Congress of Mathematicians*, pages 172–195, Berkeley, California, USA, 1986.

[197] D. Stewart. A platform with 6 degrees of freedom. In *institution of mechanical engineers*, volume 180, pages 371–386, 1965-66.

[198] V. Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:354–356, 1969.

[199] L. Tancredi and M. Teillaud. Application de la géométrie synthétique au problème de modélisation géométrique directe des robots parallèles. *Mechanism and Machine Theory*, 34(2):255–269, 1999.

[200] A. Thue. Et par andtydninger til en talteoretisk methode. *Christiana*, 7, 1902.

[201] Q.N. Tran. A symbolic-numerical method for finding a real solution of an arbitrary system of nonlinear algebraic. *Journal of Symbolic Computation*, 26:739–761, 1998.

[202] W. Trinks. On improving approximate results of Buchberger's algorithm by Newton's method. In B. Caviness, editor, *Proceedings EUROCAL '85*, number 204 in Lecture Notes in Computer Science, pages 608–611. Springer-Verlag, 1985.

[203] B. L. Van der Waerden. *Moderne Algebra I.* Berlin, 1930.

[204] J. von zur Gathen and J. Gerhard. *Modern computer algebra.* Cambridge University Press, 1999.

[205] D. Wang. An elimination method for polynomial systems. *J. Symb. Comp.*, 16:83–114, 1993.

[206] V. Weispfenning. Comprehensive Gröbner bases. *Journal of Symbolic Computation*, 14(1):1–30, 1992.

[207] D. Wiedemann. Solving sparse linear equations over finite fields. *IEEE Transactions on informations theory*, IT-32:54–62, 1986.

[208] F. Winkler. A p-adic approach to the computation of Gröbner bases. *Journal of Symbolic Computation*, 6:287–304, 1988.

[209] W.T. Wu. On zeros of algebraic equations — an application of Ritt principle. *Kexue Tongbao*, 31:1–5, 1986.

[210] H. Zassenhaus. On Hensel factorization. *I. J. Number Theory*, pages 291–311, 1969.

[211] R. Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings EUROSAM' 79*, number 72 in Lecture Notes in Computer Science. Springer, 1979.

[212] R. Zippel. Inteprolating polynomials from their values. *Journal of Symbolic Computation*, 9, 1990.

[213] R. Zippel. *Effective Polynomial Computation.* ECS 241. Kluwer Academic, 1993.