Calcul algébrique efficace de résolvantes relatives

Philippe Aubry, Annick Valibouze

CALCUL ALGÉBRIQUE EFFICACE DE RÉSOLVANTES RELATIVES

PHILIPPE AUBRY ET ANNICK VALIBOUZE

1. Introduction

La résolvante est un élément central du calcul algébrique ; en particulier pour déterminer le groupe de Galois d'un polynôme univarié, construire une représentation du corps de ses racines, calculer des polynômes minimaux de certains endomorphismes multiplicatifs ou d'éléments algébriques et construire également des polynômes univariés de groupe de Galois donné (problème de Galois inverse).

Historiquement, J.L. Lagrange ([8]) a introduit la résolvante dite absolue d'un polynôme univarié f afin d'unifier les méthodes de résolution par radicaux jusqu'en degré 4 et tenter de prouver qu'à partir du degré 5 le phénomène d'abaissement du degré n'étant plus systématique, le polynôme f n'est alors pas nécessairement résoluble par radicaux. Alors que le calcul d'une résolvante absolue était somme toute assez simple, deux siècles plus tard, Stauduhar introduisit la résolvante relative à un sous-groupe L du groupe symétrique contenant le groupe de Galois G du polynôme f ([12]). Nous allons en voir l'intérêt mais auparavant nous devons apporter un éclairage essentiel sur l'effectivité des calculs algébriques dans le corps des racines de f.

Deux des théorèmes fondamentaux de l'algèbre sont le théorème fondamental des fonctions symétriques et le théorème de Galois ; sous leur aspect non effectif, ces théorèmes sont en fait les mêmes puisqu'ils expriment tous deux que si un polynôme est invariant par le groupe L contenant le groupe de Galois G alors son évaluation en les racines de f appartient au corps k des coefficients de f. Dans un cas $L = S_n$, le groupe symétrique de degré $n =\deg(f)$, et dans l'autre $L = G$, le groupe de Galois. Pour réaliser des calculs algébriques avec les racines de f, il s'agit de rendre effectif ce théorème.

Lorsque $L = S_n$, nous disposons de nombreuses méthodes effectives du théorème fondamental des fonctions symétriques. Ce sont ces diverses méthodes auxquelles viennent s'ajouter parfois des formules combinatoires qui sont utilisées pour calculer les résolvantes absolues (voir par exemple [8],[11], [13]). Une des raisons qui fait de la résolvante (absolue ou non) un élément central du calcul algébrique est que sa factorisation sur le corps k permet d'aboutir à une forme effective du théorème de Galois. Mais se pose alors le problème

Date: July 22, 2009.

[0]*AMS Subject Classification 2000:* 12F10 12Y05 11Y40.

Keywords: minimal polynomial, Galois group, galoisian polynomial system, triangular ideal, splitting field.

de sa factorisation lorsque son degré est élevé. L'exemple le plus édifiant étant celui de la résolvante absolue de Galois dont le degré est $\deg(f)!$; pourtant seul suffit un quelconque de ses facteurs irréductibles sur k, nécessairement de degré l'ordre du groupe de Galois. Lorsque $G \neq S_n$, le recours à la résolvante relative à un sous-groupe strict L de S_n est attractif pour les deux raisons suivantes : d'une part, elle est un facteur strict de la résolvante absolue (de degré le stabilisateur de l'invariant considéré dans L et non plus dans S_n) et, d'autre part, au regard de l'ordonnancement des racines, elle porte en elle des informations plus précises qu'en tant que facteur de la résolvante absolue. Par exemple, la résolvante de Galois relative au groupe de Galois (i.e. $L = G$) est de degré l'ordre du groupe de Galois et les conjugués d'une de ses racines sont obtenus par ses permutations par le groupe de Galois en tant que groupe de permutations dont l'action sur les racines est déterminée (et non à un isomorphisme près comme pour la résolvante absolue).

Toutefois, le calcul algébrique des résolvantes relatives a longtemps été considéré comme infaisable puis ardu (voir [1] où une méthode est proposée page 27). Hormis la méthode numérique restreinte au cas $k = \mathbb{Z}$ ([12]) et quelques méthodes combinatoires adaptées à des multi-résolvantes absolues particulières, jusqu'à notre algorithme proposé dans [2] aucune méthode algébrique générale ne venait concurrencer les diverses méthodes efficaces adaptées aux seules absolues.

Cet article propose un algorithme algébrique de calcul de résolvantes relatives améliorant l'efficacité de notre précédant algorithme et s'inspirant en partie de celui de Lehobey destiné aux résolvantes absolues (voir [9]). Nous montrons en quoi l'algorithme de Lehobey est cependant propre aux résolvantes absolues.

Dans tout cet article, nous fixons un corps parfait k et un polynôme f unitaire de degré n appartenant à $k[x]$. Nous fixons $\underline{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ un n-uplet des racines de f dans une clôture algébrique K de k. Nous supposons f sans racine multiple mais pas nécessairement irréductible. Enfin, nous noterons $\mathfrak{S}_n$ le groupe symétrique de degré n et $\sigma.r$ l'action naturelle d'une permutation σ de $\mathfrak{S}_n$ sur un polynôme r de $k[x_1, \ldots, x_n]$ (i.e. agissant par permutation sur ses indices).

2. Idéaux galoisiens et groupe de Galois

Dans ce paragraphe et le suivant constitués de rappels, la référence par défaut est [14].

Les idéaux galoisiens ont été introduits afin d'établir l'algorithme récursif `GaloisIdeal` construisant le corps $k(\underline{\alpha})$ des racines de f par isomorphisme avec l'anneau quotient

$$k[x_1, \ldots, x_n]/\mathfrak{M}$$

où

$$\mathfrak{M} = \{r \in k[x_1, \ldots, x_n] \mid r(\underline{\alpha}) = 0\}$$

est l'idéal maximal des $\underline{\alpha}$-relations. Les entrées initiales de cet algorithme sont au pire les représentants respectifs des classes de conjugaison des sous-groupes de $\mathfrak{S}_n$ et l'idéal

$$\mathcal{S} = \{r \in k[x_1, \ldots, x_n] \mid \forall \sigma \in \mathfrak{S}_n \quad \sigma.r(\underline{\alpha}) = 0\}$$

des relations symétriques engendré par les *modules de Cauchy* (voir la définition 9.5). L'appel récursif se réalise avec un idéal galoisien et une liste de groupes candidats à être le groupe de Galois (donnés à une certaine classe de conjugaison près). D'un appel à l'autre, il y a croissance des idéaux et décroissance de la liste des groupes candidats pour aboutir à l'idéal maximal $\mathfrak{M}$ et à un seul groupe candidat

$$G = \{\sigma \in \mathfrak{S}_n \mid \ \sigma.\mathfrak{M} \subset \mathfrak{M}\}$$

appelé *groupe de Galois de* $\underline{\alpha}$ *sur* k.

L'élément central de cet algorithme est la résolvante ; non seulement, elle réduit la liste des groupes candidats par identification des degrés et des groupes de Galois de ses facteurs avec une matrice dite de groupes mais, de surcroît, ces mêmes facteurs sont utilisés pour calculer un idéal galoisien incluant celui donné en argument.

Dans l'introduction, nous expliquons que les résolvantes absolues sont facilement calculables via les diverses formes effectives du théorème fondamental des fonctions symétriques. Pour calculer une résolvante non absolue, nous aurons recours à un système de générateurs d'un idéal galoisien I distinct de $\mathcal{S}$. L'algorithme `GaloisIdeal` et notre algorithme lui-même de calcul de résolvantes relatives rendent effective notre hypothèse de l'existence de ce système de générateurs.

Un idéal I de $k[x_1, \ldots, x_n]$ est dit *galoisien* associé à f s'il s'exprime sous la forme :

$$I \quad = \quad \{r \in k[x_1, \ldots, x_n] \mid \forall \sigma \in L \quad \sigma.r(\underline{\alpha}) = 0\}$$

où L est un sous-ensemble du groupe symétrique $\mathfrak{S}_n$.

L'*injecteur* de I dans un idéal J est l'ensemble des permutations envoyant globalement I dans J :

$$Inj(I, J) = \{\sigma \in \mathfrak{S}_n \mid \ \sigma.I \subset J\} \quad .$$

L'injecteur de l'idéal $\mathfrak{M}$ est le groupe de Galois G, celui de l'idéal $\mathcal{S}$ dans $\mathfrak{M}$ est $\mathfrak{S}_n$, et celui de I dans $\mathfrak{M}$ s'identifie à l'ensemble de permutations GL.

Sans perte de généralité, on peut supposer que L est l'injecteur de I dans $\mathfrak{M}$. La variété V des zéros de I vérifie

$$V = L.\alpha = \{(\alpha_{\sigma(1)}, \ldots, \alpha_{\sigma(n)}) \ , \ \sigma \in L\} \ .$$

Lorsque L est un groupe, l'idéal I est dit *pur* et il est triangulaire (voir [2]). De plus, L s'identifie à l'injecteur de I dans lui-même, le *groupe de décomposition* de I, qui, dans ce cas seulement, contient le groupe de Galois G. Il sera aussi appelé l'*injecteur de* I.

Lorsque L n'est pas un groupe, il est possible de remplacer l'idéal I par un idéal galoisien pur qui le contient strictement (voir [15]). Les calculs de résolvantes avec le nouvel idéal seront donc plus rapides qu'avec l'idéal I (voir à ce sujet notre introduction concernant l'intérêt des résolvantes relatives).

Exemple 2.1. *Prenons comme exemple le polynôme* $f = x^8 + x^6 + 2x^2 + 4$ *de groupe de Galois* $Aut_{\mathbb{Q}}(f) = 8T_{19}$ *sur* $\mathbb{Q}$ *(ce polynôme est pris dans la base de données de J. Klüners*

[6]*). Son idéal galoisien engendré par l'ensemble triangulaire :*

$$\left\{\begin{array}{l} x_8^8+x_8^6+2x_8^2+4,\\ x_7+x_8,\\ x_6^2+1/2x_8^6+1/2x_8^4+1,\\ x_5+x_6,\\ x_4^4+(-1/2x_8^6-1/2x_8^4+x_8^2)x_4^2+2,\\ x_3+x_4,\\ x_2^2+x_4^2-1/2x_8^6-1/2x_8^4+x_8^2,\\ x_1+x_2 \end{array}\right\} \quad .$$

possède pour injecteur le groupe $L =< (7,8),(1,3)(2,4),(1,5,3,8)(2,6,4,7) >$*, un conjugué de* $8T_{35}$*.*

Pour tout l'article, on considère que les variables sont ordonnées par $x_1 > \cdots > x_n$ et nous fixons un idéal galoisien pur I d'injecteur un groupe L, et un ensemble triangulaire séparable

$$T = \{f_1(x_1,\ldots,x_n), f_2(x_2,\ldots,x_n),\ldots,f_{n-1}(x_{n-1},x_n), f_n(x_n)\}$$

qui l'engendre.

L'ensemble T forme une base de Groebner minimale de I pour l'ordre lexicographique. Rappelons que, par définition des ensemble triangulaires séparables, chaque polynôme f_i est unitaire en la variable x_i.

3. Polynômes minimaux et résolvante

Cette partie s'applique à tout idéal galoisien, pur ou non. Ici et pour toute la suite, nous fixons un polynôme Θ appartenant à l'anneau $k[x_1,\ldots,x_n]$, nous posons

$$\theta = \Theta(\underline{\alpha}),$$

l'évaluation de Θ en $\underline{\alpha}$, et nous notons H le stabilisateur $Stab_L(\Theta)$ de Θ dans L.

Le groupe de Galois G est isomorphe au groupe des k-automorphismes du corps $k(\underline{\alpha})$ des racines de f. Pour cette raison, pour tout $g \in G$, nous pouvons noter

$$\theta^g = (g.\Theta)(\underline{\alpha}) \quad .$$

Rappelons que le groupe de Galois est le plus grand sous-ensemble de $\mathfrak{S}_n$ tel que cette notation ait un sens. Par la théorie de Galois classique, le *polynôme minimal de* θ *sur* k, polynôme unitaire irréductible sur k de racine θ, est donné par

$$Irr_{\theta,k} = \prod_{\psi\in\{\theta^g,\ g\in G\}} (x-\psi) \quad .$$

Soit par ailleurs, l'endomorphisme multiplicatif $\widehat{\Theta}$ de $k[x_1,\ldots,x_n]/I$ qui à P associe la classe de $\Theta.P$; le polynôme caractéristique de $\widehat{\Theta}$ est le polynôme de degré $\mathrm{card}(L)$ donné par

$$\chi_{\widehat{\Theta},I} = \prod_{\sigma\in L}(x-(\sigma.\Theta)(\underline{\alpha})) \quad ;$$

comme le corps k est parfait, le polynôme minimal de $\widehat{\Theta}$ est sa forme sans facteur carré :

$$Min_{\widehat{\Theta},I} = \prod_{\psi\in(L.\Theta)(\underline{\alpha})}(x-\psi) \quad .$$

Le groupe G étant un sous-groupe de L, il apparaît que $Irr_{\theta,k}$ est un facteur irréductible de $Min_{\widehat{\Theta},I}$ qui est le produit des polynômes minimaux des $\Theta(\underline{\beta})$ où $\underline{\beta}$ parcourt la variété V de I (sans duplication).

La *résolvante de* $\underline{\alpha}$ *par* Θ *relative à* L est le polynôme de $k[x]$

$$R_{\Theta,I} = \prod_{\Psi\in L.\Theta}(x-\Psi(\underline{\alpha})) \quad ;$$

l'idéal I suffit à la référencer car la résolvante ne dépend pas du choix de $\underline{\alpha}$ dans V. Le polynôme caractéristique est une puissance de la résolvante :

$$\chi_{\widehat{\Theta},I} = R_{\Theta,I}^{\mathrm{card}(H)} \quad ; \tag{1}$$

d'où, puisque k est parfait, l'appartenance de la résolvante à $k[x]$. Si la résolvante est sans facteur carré, elle s'identifie au polynôme $Min_{\widehat{\Theta},I}$.

Ainsi, le calcul d'une résolvante induit celui de $Min_{\widehat{\Theta},I}$ et de polynômes minimaux d'éléments de $k(\alpha_1,\ldots,\alpha_n)$.

4. Racine r-ième d'un polynôme univarié

Les algorithmes que nous allons établir pour le calcul de résolvantes font appel à des extractions d'une puissance d'un polynôme univarié. Nommons f ce polynôme et supposons-le unitaire de degré n dans $k[x]$. Plaçons nous dans l'hypothèse où il existe r un entier positif et h un polynôme de $k[x]$ de degré s tels que :

$$f = h^r \quad .$$

Le problème consiste à déterminer le polynôme h à partir de f et de r.

Différents algorithmes existent, qui sont efficaces pour les corps de caractéristique nulle ou lorsque la caractéristique $\mathrm{car}(k)$ ne divise pas r. Dans [5], Henrici propose un algorithme en $O(s^2)$ sous la condition additionnelle que $\mathrm{car}(k) > s$ dans le cas d'une caractéristique non nulle. Il nécessite uniquement la connaissance des $s+1$ premiers coefficients de plus hauts degrés de f pour obtenir h. Dans le cadre plus général de la décomposition fonctionnelle de polynômes, si le corps de base contient au moins n éléments, la méthode de Kozen-Landau a une complexité $O(n^2 r)$, ou $O(n^2)$ (voir [7]). Les méthodes à base d'itérations de Newton

améliorent la complexité en $O(M(n)\log r)$, où $M(n)$ est le coût en opérations arithmétiques de la multiplication de deux polynômes de $k[x]$ (voir [3] et [16]). Dans le cadre de nos calculs, cette meilleure complexité asymptotique intervient assez peu puisque les entiers r et s restent plutôt petits.

Le nouvel algorithme présenté ici dans la partie 4.2 intervient dans l'implantation du calcul de la résolvante par notre méthode. Il permet de calculer h sous les mêmes conditions que l'algorithme de [5], avec la même complexité $O(s^2)$, et s'avère très facile à implanter.

Notre algorithme étant basé sur les relations de Girard-Newton entre les fonctions symétriques élémentaires des racines de f et ses fonctions puissances, nous débutons par un rappel à ce sujet.

4.1. **polynômes symétriques.**

Un polynôme s de $k[x_1,\ldots,x_n]$ est dit *symétrique* (en $x_1,x_2,\ldots,x_n$) si $s=\sigma.s$ pour toute permutation $\sigma\in\mathfrak{S}_n$. Nous notons alors $s(f)$ l'évaluation de s en les n racines du polynôme f.

Deux bases importantes de l'anneau $k[x_1,\ldots,x_n]^{\mathfrak{S}_n}$ des polynômes symétriques sont rappelées ci-dessous :

- les *fonctions symétriques élémentaires* en $x_1,\ldots,x_n$, notées $e_0,e_1,e_2,\ldots,e_n,\ldots$, et définies par $e_0=1$, $e_r=0$ si $r>n$, et pour $r\in[\![1,n]\!]$
$$e_r=\sum_{m\in\mathfrak{S}_n.(x_1x_2\ldots x_r)} m\ ;$$
- les *fonctions puissances* (encore appelées fonctions de Newton), en $x_1,\ldots,x_n$, notées $p_0,p_1,p_2,\ldots,p_n,\ldots$, et données par
$$p_r=\sum_{i=1}^{n} x_i^r \quad .$$

Les formules de Girard-Newton forment un système triangulaire permettant de passer d'une base à une autre : pour tout entier $r>0$

$$p_re_0-p_{r-1}e_1+\cdots+(-1)^{r-1}p_1e_{r-1}+(-1)^r r.e_r=0 \quad . \tag{2}$$

Posons $a_i=a_i(f)=(-1)^ie_i(f)$. Alors le polynôme f s'exprime sous la forme

$$f=x^n+a_1x^{n-1}+a_2x^{n-2}+\cdots+a_n \tag{3}$$

et les relations de Girard-Newton nous donnent :

$$p_r(f)+p_{r-1}(f)a_1+\cdots+p_1(f)a_{r-1}+r.a_r=0 \quad . \tag{4}$$

4.2. **Algorithme calculant la racine r-ième.**

Si $\alpha_1, \alpha_2, \ldots, \alpha_s$ sont les s racines du polynôme h alors chacune d'elles est racine du polynôme f avec une multiplicité r, ce qui se traduit pour tout $i \in [\![1, n]\!]$ par

$$p_i(f) = \sum_{j=1}^{s} r\alpha_j^i = r\, p_i(h) \quad .$$

Ainsi, le calcul des fonctions puissances des racines de h se déduit trivialement des s premières fonctions puissances des racines de f.

A l'aide des relations ci-dessus, nous obtenons l'algorithme NthRoot suivant, qui détermine h à partir de la puissance r et du polynôme unitaire f. Pour une raison qui s'éclaircira lors de l'élaboration de notre dernier algorithme, nous surchargeons la fonction NthRoot comme suit :

```
NthRoot(f, r):
    %% Sortie : h tel que h^r = f
    NthRoot(f, r, deg(f)/r)
```

et

```
NthRoot(f, r, s):
    Extraire les coefficients a_1, ..., a_s de f à l'aide de la relation (3)
    Calculer successivement p_1(f), ..., p_s(f) à l'aide de la relation (4)
    Pour j de 1 à s Répéter p_j(h) := p_j(f)/r Fin Pour
    Calculer successivement e_1(h), ..., e_s(h) à l'aide de la relation (2)
    Renvoyer h = x^s - e_1(h)x^{s-1} + ... + (-1)^{s-1} e_{s-1}(h)x + (-1)^s e_s(h)
```

L'algorithme est valide lorsque la caractéristique du corps de base ne divise pas r. La résolution du système linéaire triangulaire des formules de Girard-Newton ne nécessite que la connaissance des $s+1$ premiers coefficients $1, a_1, \ldots, a_s$ de f et se réalise en $O(s^2)$. Notons cependant qu'avec la formule de Girard-Newton, pour obtenir $e_1(h), \ldots, e_s(h)$ à partir de $p_1(h), p_2(h), \ldots, p_s(h)$, on doit pouvoir diviser successivement par $2, 3, \ldots, s$, ce qui nous impose que la caractéristique du corps de base soit supérieure à s.

5. Calculer algébriquement les résolvantes

Fixons $H < L < \mathfrak{S}_n$ et $\Theta \in k[x_1, \ldots, x_n]$ tel que Θ possède H comme stabilisateur dans L. Nous cherchons à calculer la résolvante $R_{\Theta,I}$, I étant notre idéal galoisien d'injecteur L.

Introduisons pour L la notation suivante, qui s'appliquera à tout autre sous-groupe de $\mathfrak{S}_n$ et pour tout $i \in [\![1, n+1]\!]$:

$$L_i = \{\tau \in L \mid \forall k \in [\![i, n]\!],\ \tau(k) = k\}\,.$$

Inductivement, $L_{n+1} = L$ et L_i est le stabilisateur de i sous l'action de L_{i+1} ; d'où la chaîne d'inclusions suivante

$$\{Id\} = L_1 < L_2 < \ldots < L_n < L_{n+1} = L.$$

De plus, on a les inclusions suivantes :

$$\forall i \in [\![1, n]\!]\ ,\ H_i < L_i \qquad .$$

Pour finir, notons $h = \mathrm{card}(H)$, $\ell = \mathrm{card}(L)$, $h_i = \mathrm{card}(H_i)$, $\ell_i = \mathrm{card}(L_i)$, $m_i = \mathrm{card}(H_i)/\mathrm{card}(H_{i-1})$, et rappelons que (voir [2])

$$\deg_{x_i}(f_i) = \ell_{i+1}/\ell_i \qquad .$$

L'article [2] propose l'algorithme algébrique suivant afin de calculer la résolvante relative $R_{\Theta,I}$:

```
Resolvent(T, Θ, h):
    χ := x − Θ
    Pour i de 1 à n Faire
        χ := Res_{x_i}(f_i, χ)
        Fin Pour
    Retourner(χ^{1/h})
```

Cet algorithme calcule à l'aide de la boucle le polynôme caractéristique $\chi_{\widehat{\Theta},I}$, et la résolvante s'en déduit par une extraction de racine suivant l'identité (1). Les résultants successifs font cependant croître la taille des polynômes intermédiaires ; ils mènent au polynôme caractéristique qui est de degré ℓ alors que la résolvante est seulement de degré ℓ/h.

Dans le cas particulier du calcul de résolvantes absolues (i.e. lorsque $L = \mathfrak{S}_n$), où T est composé des modules de Cauchy de f, F. Lehobey a repris à la même époque un algorithme qui fait aussi appel à des calculs de résultants ([10]) en montrant qu'il est possible d'éliminer une partie de la puissance h superflue à chaque étape de la boucle ([9]). On peut ainsi calculer la résolvante absolue de la manière suivante en limitant l'explosion de la taille des données :

AbsoluteResolvent$(T, \Theta, [m_2, \ldots, m_{n+1}])$:
  $\mathcal{L} := x - \Theta$
  Pour i de 1 à n Faire
    $\gamma := \mathrm{Res}_{x_i}(f_i, \mathcal{L})$
    $\mathcal{L} := \gamma^{1/m_{i+1}}$
    Fin Pour
  Retourner($\mathcal{L}$)

Pour un contrôle encore plus efficace, F. Lehobey calcule en fait le polynôme réciproque de $\mathcal{L}$ avec lequel, à la i-ème étape, le résultant peut-être alors calculé modulo x^{s+1}, où $s = \frac{i!}{h_{i+1}}$ est le degré en x de $\mathcal{L}$ à cette étape (voir à ce sujet le corollaire 7.4).

6. Variétés galoisiennes

Dans cette partie, nous étudions plus particulièrement les projections de la variété $L.\underline{\alpha}$ de I dans K^n et les fibres au-dessus d'un de leurs points.

Soit L un sous-groupe de $\mathfrak{S}_n$ et $V = L.\underline{\alpha} \subseteq K^n$. Pour tout entier $i \in [\![1, n]\!]$ on note $\pi_i : K^n \longrightarrow K^{n-i+1}$ la projection sur les $n - i + 1$ dernières composantes, et $V_i = \pi_i(V)$.

Etant donné un sous-groupe H de L, on note L/H l'ensemble des classes à gauche de L modulo H, et l'expression $g \in L/H$ fera référence à un élément d'une transversale à gauche de L modulo H.

Proposition 6.1. *Soit $i \in [\![1, n]\!]$. Si L est un groupe alors la projection $\pi_i(V)$ de la variété $V = L.\underline{\alpha}$ est égale à*

$$V_i = \{(\alpha_{\tau(i)}, \ldots, \alpha_{\tau(n)}), \tau \in L/L_i\} \ . \tag{5}$$

De plus, la fibre des points de V au-dessus du point $M = (\alpha_{\tau(i)}, \ldots, \alpha_{\tau(n)})$ de V_i est l'ensemble

$$\tau L_i.\alpha = \{(\alpha_{\tau\sigma(1)}, \ldots, \alpha_{\tau\sigma(i-1)}, \alpha_{\tau(i)}, \ldots, \alpha_{\tau(n)}),\ \sigma \in L_i\} \ . \tag{6}$$

Preuve. Pour tout $\tau' \in L$ on a

$$\begin{array}{rcl} \tau' \in \tau L_i & \iff & \tau'(i) = \tau(i), \ldots, \tau'(n) = \tau(n) \\ & \iff & \pi_i(\tau'.\underline{\alpha}) = \pi_i(\tau.\underline{\alpha}), \end{array}$$

ce qui entraîne la relation (6).

La décomposition de L suivant ses classes à gauche modulo L_i fournit donc ci-dessous une décomposition en singletons distincts deux-à-deux et prouve la relation (5).

$$\begin{array}{rcl} \pi_i(V) & = & \pi_i(\displaystyle\bigcup_{\tau \in L/L_i} \tau L_i.\alpha) \\ & = & \displaystyle\bigcup_{\tau \in L/L_i} \{(\alpha_{\tau(i)}, \ldots, \alpha_{\tau(n)})\} \ . \end{array}$$

□

Ce qui précède entraîne immédiatement le corollaire suivant

Corollaire 6.2. *Avec les notations ci-dessus, nous avons*

- $\mathrm{card}(V_i) = \mathrm{card}(L)/\mathrm{card}(L_i)$,
- $\mathrm{card}(\pi_i^{-1}(M)) = \mathrm{card}(L_i)$ *pour tout point* M *de* V_i.

Pour $j \leq i$ on note

$$\pi_{j,i} : K^{n-j+1} \longrightarrow K^{n-i+1}$$

la projection sur les i dernières composantes.

Proposition 6.3. *Soient* i *et* j *deux entiers tels que* $1 \leq j \leq i \leq n$ *et* $\tau \in L/L_i$. *La fibre des points de* V_j *au-dessus du point* $M = (\alpha_{\tau(i)}, \ldots, \alpha_{\tau(n)}) \in V_i$ *est donnée par*

$$\pi_{j,i}^{-1}(M) = \{(\alpha_{\tau\sigma(j)}, \ldots, \alpha_{\tau\sigma(n)}), \sigma \in L_i/L_j\} \ , \tag{7}$$

et

$$\mathrm{card}(\pi_{j,i}^{-1}(M)) = \mathrm{card}(L_i)/\mathrm{card}(L_j) \ .$$

Preuve. Puisque $L_j < L_i < L$, les transversales à gauche vérifient

$$L/L_j = (L/L_i)\ (L_i/L_j)\,.$$

Le résultat se déduit alors des relations suivantes :

$$\begin{aligned} V_j &= \{\tau\sigma.(\alpha_j, \ldots, \alpha_n),\ \tau \in L/L_i, \sigma \in L_i/L_j\} \\ &= \{(\alpha_{\tau\sigma(j)}, \ldots, \alpha_{\tau\sigma(n)}),\ \tau \in L/L_i, \sigma \in L_i/L_j\} \\ &= \{(\alpha_{\tau\sigma(j)}, \ldots, \alpha_{\tau\sigma(i-1)}, \alpha_{\tau(i)}, \ldots, \alpha_{\tau(n)}),\ \tau \in L/L_i, \sigma \in L_i/L_j\} \end{aligned}$$

puisque $\sigma \in L_i$. En fixant τ, on obtient bien l'égalité (7) pour $\pi_{j,i}^{-1}(M)$ et, dans le dernier ensemble de cette suite d'égalités, les points $(\tau\sigma(j), \ldots, \tau\sigma(n))$ où σ parcourt L_i/L_j sont tous distincts. En effet, pour deux permutations σ et σ' d'une transversale de L_i modulo L_j, puisque $\sigma \notin \sigma' L_j$, on a $(\sigma(j), \ldots, \sigma(n)) \neq (\sigma'(j), \ldots, \sigma'(n))$. □

Corollaire 6.4. *Soit* $\{f_1, \ldots, f_n\}$ *l'ensemble triangulaire de* $k[x_1, \ldots, x_n]$ *générateur de l'idéal* I *de* $V = L.\underline{\alpha}$, *où* L *est un sous-groupe de* $\mathfrak{S}_n$ *contenant le groupe de Galois de* $\underline{\alpha}$. *Soit* $i \in [\![1, n]\!]$. *Pour tout* $\tau \in L$, *l'ensemble des zéros du polynôme univarié* $f_i(x, \alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)})$ *est égal à l'ensemble*

$$\{\alpha_{\tau\sigma(i)}, \sigma \in L_{i+1}/L_i\}.$$

Preuve. V_i est l'ensemble des zéros de $\{f_i, \ldots, f_n\}$. Soit $M = (\alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)})$ un point de V_{i+1}. Alors β est un zéro de $f_i(x, \alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)})$ si et seulement si $M' = (\beta, \alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)})$ est un zéro de $\{f_i, \ldots, f_n\}$, ce qui revient à dire que $M' \in \pi_{i,i+1}^{-1}(M)$. □

7. Eliminer les puissances superflues pour les résolvantes relatives

Dans cette section, la proposition 7.5 et son corollaire forment le fondement de l'algorithme RelativeResolvent1 donné à leur suite. Celui-ci améliore l'algorithme Resolvent de la section 5, qui se termine par l'extraction d'une racine h-ième du polynôme caractéristique. Nous montrons ici que cette puissance finale superflue h est le produit de puissances superflues qui peuvent être éliminées à chaque pas de la boucle.

Ces résultats découlent de propriétés relatives aux résultants que nous énonçons tout d'abord.

Lemme 7.1. *Soit A un anneau commutatif unitaire intègre. Soient f et g deux polynômes de $A[y]$ avec $f = a\prod_{i=1}^{d}(y-\alpha_i)$ et $g = b\prod_{j=1}^{e}(y-\beta_j)$ où les α_i et les β_j sont les racines respectivement de f et g dans une clôture algébrique du corps des fractions de A. Alors*

$$\mathrm{Res}_y(f,g) \quad = \quad a^e \prod_{i=1}^{d} g(\alpha_i)$$

Pour qu'un résultant ait de bonnes propriétés de spécialisation, il faut que le coefficient de tête de l'un au moins des polynômes f ou g ne se spécialise pas à 0. Les polynômes f_i avec lesquels nous calculons des résultants dans nos algorithmes sont unitaires en x_i. Dans ce cas le résultant des spécialisés est exactement le spécialisé du résultant.

Lemme 7.2. *Soit I un idéal de A. Pour tout $p \in A[y]$ on note $\overline{p}$ l'image de p modulo I. Si f est unitaire alors*

$$\overline{\mathrm{Res}_y(f,g)} = \mathrm{Res}_y(\overline{f},\overline{g}) \ ,$$

À la i-ème étape, nos calculs font intervenir la variable x de la résolvante $\mathcal{L}$ et $y = x_i$, la variable d'élimination du résultant. Le polynôme f_i est unitaire en x_i et ne contient pas x, tandis que la valeur initiale $x - \Theta$ de $\mathcal{L}$ est unitaire en x. Par conséquent, tous les résultants successifs resteront unitaires par rapport à la variable x, avec un degré déterminable comme suit :

Lemme 7.3. *Soit f et g deux polynômes de $A[y,x]$ vérifiant les deux propriétés suivantes:*

- *$g = x^m + g'(x,y)$ où g' est de degré au plus $m-1$ en x,*
- *f est unitaire par rapport à la variable y et $\deg_y(f) = d$.*

Alors $\mathrm{Res}_y(f,g)$ est un polynôme unitaire en la variable x, de degré md par rapport à cette variable.

Preuve. En effet, d'après les relations du lemme 7.1, le résultant par rapport à y peut s'écrire sous la forme

$$\mathrm{Res}_y(f,g) = \prod_{i=1}^{d} g(x,\alpha_i) = \prod_{i=1}^{d}(x^d + g'(x,\alpha_j)) \ ,$$

où les α_j désignent les racines de f dans une clôture du corps des fractions de l'anneau $A[y]$. □

Dans l'algorithme Resolvent de la section 5, le polynôme χ appartient à $k[x_1, \ldots, x_n, x]$ et il est de degré 1 en la variable x. Nous posons $\chi_1 = \chi$, et pour $i \in [\![2, n+1]\!]$ nous notons χ_i la valeur de χ obtenue à la $(i-1)$-ième étape de la boucle, de sorte que $\chi_i \in k[x_i, \ldots, x_n, x]$. Puisque les f_i sont unitaires en x_i, les degrés en x des résultants successifs se déduisent en appliquant inductivement le lemme 7.3, d'où la propriété suivante.

Corollaire 7.4. *Avec les notations ci-dessus, pour tout $i \in [\![2, n+1]\!]$ le polynôme χ_i est unitaire en x dans $k[x_i, \ldots, x_n][x]$ et*

$$\deg_x(\chi_i) = \deg_{x_1}(f_1) \deg_{x_2}(f_2) \ldots \deg_{x_{i-1}}(f_{i-1}) = \mathrm{card}(L_i) \ .$$

L'algorithme Resolvent est à base de résultants emboîtés par rapport aux polynômes f_i respectivement unitaires en la variable d'élimination et qui engendrent l'idéal triangulaire I. Les propriétés précédentes nous permettent de définir plus précisément les polynômes intermédiaires qu'il calcule, et de montrer qu'ils s'écrivent comme une puissance d'un autre polynôme dans l'anneau $k[x_1, \ldots, x_n, x]/I$.

Proposition 7.5. *Soit A un anneau intègre, et Ψ un polynôme de $A[x_1, \ldots, x_n]$ invariant par $H < \mathfrak{S}_n$. On définit inductivement $\Psi_1, \Psi_2, \ldots, \Psi_{n+1}$ comme suit:*

$$\begin{aligned} \Psi_1 &= \Psi \in A[x_1, \ldots, x_n] \,, \\ \Psi_{i+1} &= \mathrm{Res}_{x_i}(f_i(x_i, \ldots, x_n), \Psi_i(x_i, \ldots, x_n)) \in A[x_{i+1}, \ldots, x_n] \ . \end{aligned}$$

Si $\tau \in L$ alors, pour tout $i \in [\![1, n]\!]$, on a

$$\Psi_i(\alpha_{\tau(i)}, \ldots, \alpha_{\tau(n)}) = \prod_{\sigma \in L_i} \sigma.\Psi(\tau.\underline{\alpha}) \quad .$$

Preuve. La relation est évidente pour $i = 1$. Par induction, supposons que la relation de la proposition soit vérifiée pour $i \leq n$. Alors, en posant $U = \Psi_{i+1}(\alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)})$, nous

avons

$$\begin{aligned}
U &= \mathrm{Res}_{x_i}(f_i, \Psi_i)(\alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)}) \\
&= \mathrm{Res}_{x_i}(f_i(x_i, \alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)}), \Psi_i(x_i, \alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)})) \text{ (lemme 7.2)} \\
&= \prod_{\beta \in \mathcal{V}(f_i(x_i, \alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)}))} \Psi_i(\beta, \alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)}) \quad \text{(lemme 7.1)} \\
&= \prod_{\sigma \in L_{i+1}/L_i} \Psi_i(\alpha_{\tau\sigma(i)}, \alpha_{\tau(i+1)}, \ldots, \alpha_{\tau(n)}) \quad \text{(corollaire 6.4)} \\
&= \prod_{\sigma \in L_{i+1}/L_i} \Psi_i(\alpha_{\tau\sigma(i)}, \alpha_{\tau\sigma(i+1)}, \ldots, \alpha_{\tau\sigma(n)}) \\
&= \prod_{\sigma \in L_{i+1}/L_i} \prod_{\sigma' \in L_i} \Psi(\tau\sigma\sigma'.\underline{\alpha}) \quad \text{(hypothèse de récurrence)} \\
&= \prod_{\sigma \in L_{i+1}} \Psi(\tau\sigma.\underline{\alpha}) \\
&= \prod_{\sigma \in L_{i+1}} \sigma.\Psi(\tau.\underline{\alpha}) \ .
\end{aligned}$$

□

Corollaire 7.6. *Pour tout entier $i \in [\![1, n+1]\!]$, posons*

$$\mathcal{L}_i = \prod_{\sigma \in L_i/H_i} \sigma.\Psi \ .$$

Alors

$$\Psi_i = {\mathcal{L}_i}^{\mathrm{card}(H_i)} \mod I \ .$$

Preuve. Soit $\sigma_1, \ldots, \sigma_m$ une transversale à gauche de L_i modulo H_i. Puisque Ψ est invariant par H_i, l'orbite $L_i.\Psi = \cup_{j=1}^{m} \sigma_j H_i.\Psi$ se réduit à $\{\sigma_1.\Psi, \ldots, \sigma_m.\Psi\}$ et chaque valeur est atteinte h_i fois.

Pour tout point M de la variété $L.\underline{\alpha}$ de l'idéal I, on obtient alors

$$\Psi_i(M) = \left(\prod_{\sigma \in L_i/H_i} \sigma.\Psi(M) \right)^{h_i} ,$$

dont on déduit le résultat. □

L'idéal galoisien I étant engendré par l'ensemble triangulaire

$$T = \{f_1(x_1, \ldots, x_n), f_2(x_2, \ldots, x_n), \ldots, f_{n-1}(x_{n-1}, x_n), f_n(x_n)\},$$

on désigne par T_i l'ensemble triangulaire $\{f_i, f_{i+1}, \ldots, f_n\}$ pour tout i dans $[\![1, n]\!]$. Nous souhaitons pouvoir calculer inductivement les $\mathcal{L}_i$, ou plus plutôt leur valeur modulo I, sans passer par celle de Ψ_i.

Proposition 7.7. *Rappelons que $m_i = h_i/h_{i-1}$. Avec les notations précédentes, on a*

$$\mathcal{L}_{i+1} \equiv \mathrm{Res}_{x_i}(f_i, \mathcal{L}_i)^{1/m_{i+1}} \mod I \ .$$

Preuve. On sait (cela résulte par exemple du lemme 7.1) que si $g' \equiv g \mod f$ alors

$$\mathrm{Res}_y(f, g') = a^{\deg g' - \deg g}\ \mathrm{Res}_y(f, g) \ .$$

On peut supposer, sans restreindre le propos, que chaque f_i est réduit modulo T_{i+1}. Puisque le polynôme f_i est unitaire en x_i, on a

$$\Psi_{i+1} = \mathrm{Res}_{x_i}(f_i, \Psi_i) = \mathrm{Res}_{x_i}(f_i, \Psi_i \mod f_i) \quad .$$

Il s'ensuit à l'aide des propriétés des résultants et du corollaire 7.6:

$$\begin{aligned} \Psi_{i+1} \mod T_{i+1} &= \mathrm{Res}_{x_i}(f_i \mod T_{i+1}, \Psi_i \mod T_i) \\ &= \mathrm{Res}_{x_i}(f_i, \mathcal{L}_i^{h_i}) \\ &= \mathrm{Res}_{x_i}(f_i, \mathcal{L}_i)^{h_i} \quad . \end{aligned}$$

Remarquons que les relations modulo T_{i+1} sont identiques aux relation modulo I puisque les polynômes considérés sont dans $k[x_{i+1}, \ldots, x_n]$. Le résultat se déduit alors de l'égalité $\mathcal{L}_{i+1} \equiv (\Psi_{i+1})^{1/h_{i+1}}$ (corollaire 7.6). □

Réciproquement, en prenant pour premier terme $\mathcal{L}_1 = \Psi$, la relation de récurrence de la proposition 7.7 dans $k[x_1, \ldots, x_n, x]/I$ définit la suite des $\mathcal{L}_i$ du corollaire 7.6.

Choisissons $\Psi = x - \Theta$. Nous avons alors

$$\mathcal{L}_i = \prod_{\sigma \in L_i/H_i} (x - \sigma.\Theta) \quad . \tag{8}$$

En particulier le dernier terme de la suite est la résolvante de $\underline{\alpha}$ par Θ relative à L:

$$\mathcal{L}_{n+1} = \prod_{\sigma \in L/H} (x - \sigma.\Theta) = R_{\Theta,I} \quad .$$

Nous déduisons de ce qui précède un premier algorithme qui permet d'obtenir la résolvante relative $R_{\Theta,I}$ en évacuant à chaque étape une puissance m_i par rapport à notre algorithme initial Resolvent. Il suffit de calculer préalablement les m_i de la liste $[m_2, \ldots, m_{n+1}]$ passée en paramètre, à partir des ordres des groupes H_i. A la i-ème étape de la boucle est calculé le polynôme $\mathcal{L}_{i+1} \mod I$; les calculs d'extraction de racines et de résultants se réalisent au-dessus de l'anneau quotient $k[x_1, \ldots, x_n]/T_{i+1}$ d'après la preuve de la proposition 7.7. Nous utilisons l'algorithme NthRoot donné dans la section 4.

RelativeResolvent1$(T, \Theta, [m_2, \ldots, m_{n+1}])$:
  $\mathcal{L} := x - \Theta \mod T$
  Pour i de 1 à n Répéter
    $\gamma := \mathrm{Res}_{x_i}(f_i, \mathcal{L}) \mod T_{i+1}$
    $\mathcal{L} :=$ NthRoot$(\gamma, m_{i+1}) \mod T_{i+1}$
    Fin Pour
  Retourner $\mathcal{L}$

Remarque 7.8. *Lorsqu'à la i-ième étape de la boucle, le polynôme $\mathcal{L}$ ne dépend pas de x_i, le calcul du résultant est inutile. Il suffit alors de réaliser l'affectation*
$\mathcal{L} := \mathcal{L}^{d_i}$
où d_i est l'entier $deg_{x_i}(f_i)/m_{i+1} = (l_{i+1}/l_i)/(h_{i+1}/h_i)$. Le résultant sachant traiter les puissances, cette puissance d_i n'ajoute pas de complexité supplémentaire au calcul qui serait réalisé sans elle. Il est également possible de ne pas modifier $\mathcal{L}$ et de passer à l'étape suivante après l'affectation $d{:=}d * d_i$ (la variable d étant initialisée à 1) ; la sortie de l'algorithme sera alors $\mathcal{L}^d$.

8. Une amélioration de l' algorithme

Dans la boucle de l'algorithme RelativeResolvent1, l'extraction de racine d'un polynôme en la variable x suit un calcul de résultant. Or, si on extrait la racine à l'aide de l'algorithme NthRoot(f,r) de la section 4 calculant le polynôme h de degré s tel que $f = h^r$, seuls sont nécessaires les $s + 1$ premiers coefficients de plus hauts degrés en x de f. Afin d'éviter ces calculs inutiles, nous proposons de calculer les résultants sur des polynômes réciproques des $\mathcal{L}$ calculés au cours de l'algorithme RelativeResolvent1 ; les coefficients inutiles sont alors éliminables par le biais de réductions modulo la bonne puissance de x. Ceci est également applicable à tout autre algorithme d'extraction de racine r-ième ne nécessitant également que les $s + 1$ premiers coefficients de f.

Nous vérifions au début de cette section que dans l'algorithme RelativeResolvent1 les opérations commutent avec le passage au polynôme réciproque. Nous en déduisons ensuite un nouvel algorithme dans lequel tous les coefficients des polynômes calculés au cours de celui-ci sont exploités dans les étapes suivantes.

Etant donné un polynôme g en x de degré n, nous adoptons la notation suivante pour le polynôme réciproque de g

$$\widetilde{g} = x^n g(\frac{1}{x}) \ .$$

Remarque 8.1. *Soient f un polynôme univarié et unitaire de degré n et h de degré s tel que $f = h^r$. Soit f' un polynôme de degré n dont les $s+1$ termes de plus hauts degrés sont identiques à ceux de f. Alors l'exécution de NthRoot(f', r) renvoie le même polynôme h que NthRoot(f, r). La spécification de NthRoot s'élargit alors ainsi : si f' est un polynôme*

en x, unitaire et de degré rs tel que

$$\widetilde{f'} \equiv \widetilde{h^r} \mod x^{s+1}$$

alors $\mathsf{NthRoot}(f', r)$ *renvoie* h. *Précisons que si les* $s+1$ *premiers termes de* f' *ne correspondent pas à une puissance* r*-ième d'un polynôme de degré* s *alors le résultat renvoyé n'a aucune signification.*

Finalement, nous allons nous affranchir de la condition sur le degré de f' *puisque nous passerons* s *en argument avec l'appel à* $\mathsf{NthRoot}(f', r, s)$. *C'est ici que vient l'éclaircissement annoncé de la surcharge de la fonction* $\mathsf{NthRoot}$.

Les propriétés du lemme suivant se vérifient immédiatement.

Lemme 8.2. *Soit A un anneau. Si f et g sont deux polynômes unitaires de $A[x]$ alors $\widetilde{f}\,\widetilde{g} = \widetilde{fg}$.*

Soit $\alpha \in A$ et $f = \sum_{i=0}^{n} a_i(y)x^i$ un polynôme de $A[x, y]$. Notons $f_\alpha(x)$ le spécialisé de f en $y = \alpha$. Si $a_n(\alpha)$ est non nul alors la spécialisation et le passage au polynôme réciproque commutent, autrement dit $\widetilde{f}(x, \alpha) = \widetilde{f_\alpha}$.

Lemme 8.3. *Soit A un anneau commutatif intègre. Soient f et g deux polynômes de $A[y, x]$. Si f est unitaire en la variable y et si g est unitaire en x alors* $\mathrm{Res}_y(f, \widetilde{g})$ *est le polynôme réciproque de* $\mathrm{Res}_y(f, g)$.

Preuve. Posons $f = \prod_{i=1}^{d}(y - \alpha_i)$ où les α_i sont les racines de f dans une clôture algébrique du corps des fractions de A. Suivant les hypothèses, l'égalité du lemme 7.1 s'écrit

$$\mathrm{Res}_y(f, \widetilde{g}) = \prod_{j=1}^{d} \widetilde{g}(x, \alpha_j)$$

Puisque g est unitaire par rapport à x, le lemme 8.3 assure que la spécialisation en les α_j et le produit commutent avec le passage aux polynômes réciproques. □

Dans l'algorithme $\mathsf{RelativeResolvent1}$, pour chaque étape, le degré de $\mathcal{L}$ en x est précalculable. À la i-ème étape, on calcule le polynôme unitaire $\mathcal{L}_{i+1} \mod I$ de degré

$$s_{i+1} = \ell_{i+1}/h_{i+1}$$

en x, d'après l'expression (8). Conformément à la remarque 8.1, le polynôme $\mathcal{L}$ obtenu lors de cette i-ème étape peut être aussi calculé par l'appel $\mathsf{NthRoot}(\gamma', m_{i+1}, s_{i+1})$ avec γ' tel que

$$\widetilde{\gamma'} = \widetilde{\gamma} \mod x^{s_{i+1}+1} \quad .$$

D'après le lemme 8.3, on a

$$\widetilde{\gamma} = \mathrm{Res}_{x_i}(f_i, \widetilde{\mathcal{L}_i}) \mod T_{i+1} \quad .$$

Par conséquent, il suffit de prendre

$$\widetilde{\gamma'} = \mathrm{Res}_{x_i}(f_i, \widetilde{\mathcal{L}_i}) \mod T_{i+1} \cup \{x^{s_{i+1}+1}\}$$

en effectuant le calcul du résultant au-dessus de l'anneau $k[x_{i+1},\ldots,x_n,x]/\langle T_{i+1}\cup\{x^{s_{i+1}+1}\}\rangle$ (propriété de spécialisation du lemme 7.2), puis de passer au polynôme réciproque pour obtenir un polynôme γ' qui convient.

En prenant en compte ce qui précède, nous en déduisons l'algorithme RelativeResolvent2 qui améliore l'algorithme RelativeResolvent1 en évitant le calcul de coefficients inutiles dans les résultants.

On note ci-dessous Recip(p) la fonction qui renvoie le polynôme réciproque de p par rapport à la variable x.

```
RelativeResolvent2(T, Θ, [m_2, ..., m_{n+1}], [s_2, ..., s_{n+1}]):
    L := x − Θ  mod T
    Pour i de 1 à n Répéter
        Γ := Recip(L)
        T' := T_{i+1} ∪ {x^{s_{i+1}}}
        Γ := Res_{x_i}(f_i, Γ)  mod T'
        γ' := Recip(Γ)
        L := NthRoot(γ', m_{i+1}, s_{i+1})   mod T'
        Fin Pour
    Retourner L
```

9. Le cas particulier des résolvantes absolues

Dans cette partie, nous expliquons pourquoi dans le cas où $L=\mathfrak{S}_n$ il ne fut pas nécessaire à F. Lehobey de réaliser les calculs modulo I pour extraire les racines à chaque étape de la boucle de son algorithme AbsoluteResolvent. En fait, nous allons montrer que $L=\mathfrak{S}_n$ satisfait la condition d'un certain cas particulier. En dehors de ce cas particulier, les calculs doivent être réalisés modulo I pour être exacts. Pour présenter ce cas particulier, fixons un entier i entre 2 et n et posons

$$A = k[x_i,\ldots,x_n] \quad .$$

Notons W la variété de l'idéal engendré par $f_1,\ldots,f_{i-1}$ dans $A[x_1,\ldots,x_{i-1}]$ et posons

$$L'_i = \{\sigma'\in\mathfrak{S}_{i-1} \mid \exists\,\sigma\in L_i \text{ t.q. } \forall j<i \quad \sigma'(j)=\sigma(j)\}$$

pour désigner la restriction de l'action de L_i à $\{1,\ldots,i-1\}$. Ce premier lemme est évident :

Lemme 9.1. *Soit $\Theta\in A[x_1,\ldots,x_{i-1}]$ et soit le polynôme*

$$R=\prod_{\sigma\in L'_{(i)}}(x-\sigma.\Theta) \quad .$$

Pour tout $\beta\in W$, l'identité $W=L'_i.\beta$ est une condition nécessaire et suffisante pour que

$$R(\beta)=\prod_{\beta\in W}(x-\Theta(\beta)) \quad . \tag{9}$$

Remarque 9.2. *Soit $(\alpha_i, \ldots, \alpha_n)$ dans la variété de $< f_i, \ldots, f_n >$ et $\overline{A} = k(\alpha_i, \ldots, \alpha_n)$. Alors pour tout $(n-i+1)$-uplet $\overline{\beta}$ de la variété $\overline{W}$ de l'idéal de $\overline{A}[x_1, \ldots, x_{i-1}]$ engendré par $f_1(x_1, \ldots, x_{i-1}, \alpha_i, \ldots, \alpha_n), \ldots, f_{i-1}(x_{i-1}, \alpha_i, \ldots, \alpha_n)$, l'identité $\overline{W} = L'_i.\overline{\beta}$ est satisfaite.*

Le lemme suivant nous donne une condition suffisante à l'hypothèse du précédent lemme :

Lemme 9.3. *Si*

$$L_i. < f_1, \ldots, f_{i-1} > = < f_1, \ldots, f_{i-1} >$$

alors, pour tout $\beta \in W = V(< f_1, \ldots, f_{i-1} >)$, nous avons

$$W = L'_i.\beta \quad .$$

Preuve. Tout d'abord, le cardinal de W est identique à celui de L'_i. En effet, W est la variété d'un idéal radical, les polynômes $f_1, \ldots, f_{i-1}$ en forment une base de Gröbner et le produit de leurs degrés initiaux est identique au cardinal de L_i, identique quant à lui à celui de L'_i. Prenons $\sigma \in L'_i$; si $\beta \in W$ alors, par hypothèse, $\sigma.\beta \in W$ ce qui implique que $L'_i.\beta \in W$. D'où l'égalité entre $L'_i.\beta$ et W. □

Avec les deux lemmes précédents, nous établissons le théorème suivant qui, avec la remarque 9.2, nous explique pourquoi dans certains cas il est possible de s'affranchir des calculs modulo I qui apparaissent dans le corollaire 7.6 :

Théorème 9.4. *Soit H le stabilisateur de Θ dans L. Sous les hypothèses du théorème précédent et en reprenant les notations du paragraphe précédent, si le groupe L vérifie :*

$$L_{(i)}. < f_1, \ldots, f_{i-1} > = < f_1, \ldots, f_{i-1} >$$

alors

$$\Psi_i \quad = \quad \mathcal{L}_i^{\mathrm{card}(H_i)} \quad . \tag{10}$$

Vérifions maintenant que le groupe symétrique satisfait effectivement l'hypothèse du théorème 9.4. Auparavant rappelons la définition des modules de Cauchy :

Définition 9.5 ([4])**.** *Les* modules de Cauchy *de f sont les polynômes $C_1, \ldots, C_n$ de $k[x_1, \ldots, x_n]$ définis inductivement comme suit :*

- $C_n(x_n) = f(x_n)$,
- *pour $i = n-1, \ldots, 1$:*

$$C_i(x_i, \ldots, x_n) = \frac{C_{i+1}(x_i, x_{i+2}, \ldots, x_n) - C_{i+1}(x_{i+1}, x_{i+2}, \ldots, x_n)}{x_i - x_{i+1}}.$$

Théorème 9.6 ([10])**.** *Les modules de Cauchy engendrent l'idéal des relations symétriques $\mathcal{S}$.*

Théorème 9.7. *Pour tout $1 \leq j \leq n$, nous avons*

$$\mathfrak{S}_n.C_j \subset \langle C_1, \ldots, C_j \rangle \quad .$$

En d'autres termes, le groupe symétrique satisfait l'hypothèse du théorème 9.4.

Preuve. Le groupe symétrique $\mathfrak{S}_n$ est engendré par les permutations $t = (n, n-1)$ et $\tau = (n, n-1, \ldots, 2, 1)$; et pour tout $j \in [\![2, n-1]\!]$ la permutation τ est remplaçable par les permutations $\tau_1 = (n, n-1, \ldots, j)$ et $\tau_2 = (j, j-1, \ldots, 1)$ puisque $\tau = \tau_2\tau_1$. Il suffit de montrer le théorème sur les générateurs de $\mathfrak{S}_n$.

Pour $j = n$, le théorème est vérifié car $\mathfrak{S}_n.\langle C_1, \ldots, C_n\rangle = \langle C_1, \ldots, C_n\rangle$ (le groupe $\mathfrak{S}_n$ est l'injecteur de l'idéal dans lui-même); pour $j = 1$, il l'est également car $\mathfrak{S}_n.C_1 = \{C_1\}$ (le polynôme C_1 est symétrique en $x_1, \ldots, x_n$).

Prenons j dans l'intervalle $[\![2, n-1]\!]$. Comme C_j est symétrique en $x_j, \ldots, x_n$, nous avons $t.C_j = \tau_1.C_j = C_j$. Pour étudier l'action de τ_2, posons $A = k[x_{j+1}, \ldots, x_n]$ et

$$F(x) = C_j(x, x_{j+1}, \ldots, x_n) \in A[x] \quad \text{(de degré } j \text{ en } x\text{)}.$$

Notons $F_j = F(x_j), F_{j-1}(x_{j-1}, x_j), \ldots, F_1(x_1, \ldots, x_j)$, les modules de Cauchy du polynôme F. Nous avons donc $\tau_2.C_j = F_j(x_{j-1}) \in < F_1, \ldots, F_j >$ qui est un idéal invariant par toute permutation des variables $x_1, \ldots, x_j$; le résultat est démontré avec cette suite d'identités :

$$F_j = C_j, F_{j-1} = C_{j-1}, \ldots, F_1 = C_1 \quad .$$

□

La propriété que nous avons établie est en réalité plus forte que celle nécessaire. Nous savons désormais pourquoi l'algorithme de Lehobey n'est pas faux mais qu'il n'est pas généralisable au calcul de toute résolvante non absolue.

10. Conclusion

Nous avons présenté dans cet article un nouvel algorithme général et algébrique pour calculer les résolvantes relatives. Nous avons éclairci les raisons pour lesquelles l'algorithme de calcul de résolvantes absolues de [9] ne se transpose pas simplement à toutes les résolvantes.

Notre algorithme améliore nettement celui de [2] et permet de traiter des calculs qui n'étaient pas accessibles par ce dernier à cause de la croissance intermédiaire des données. Il nécessite d'extraire la racine m-ième d'un polynôme en une variable de degré n. A cet effet nous avons introduit une méthode compétitive basée sur le calcul des fonctions puissances des racines du polynôme dont la complexité est en $O((n/r)^2)$.

References

[1] J.M. Arnaudiès and A. Valibouze. Résolvantes de lagrange. Technical Report 93.61, LITP, 1993.

[2] P. Aubry and A. Valibouze. Using Galois ideals for computing relative resolvents. *J. Symbolic Comput.*, 30(6):635–651, 2000.

[3] R. P. Brent and H. T. Kung. Fast algorithms for manipulating formal power series. *J. ACM*, 25(4):581–595, 1978.

[4] A. Cauchy. Usage des fonctions interpolaires dans la détermination des fonctions symétriques des racines d'une équation algébrique donnée. *Oeuvres*, **5**:473 Extrait 108, 1840.

[5] Peter Henrici. Automatic computations with power series. *J. ACM*, 3(1):10–15, 1956.

[6] J. Klüners and G. Malle. A database for polynomials over the rationals. `(http://www.mathematik.uni-kassel.de/~klueners/minimum/)`.

[7] Dexter Kozen and Susan Landau. Polymonial decomposition algorithms. *J. Symb. Comput.*, 7(5):445–456, 1989.

[8] J.-L. Lagrange. Réflexions sur la résolution algébrique des équations. *Prussian Academy*, 1770.
[9] Frédéric Lehobey. Resolvent computations by resultants without extraneous powers. In *ISSAC*, pages 85–92, 1997.
[10] N. Rennert and A. Valibouze. Calcul de résolvantes avec les modules de Cauchy. *Experiment. Math.*, 8(4):351–366, 1999.
[11] L. Soicher and J. McKay. Computing Galois groups over the rationals. *J. Number Theory*, 20(3):273–281, 1985.
[12] R.P. Stauduhar. The determination of Galois groups. *Math. Comp.*, 27:981–996, 1973.
[13] A. Valibouze. Library SYM of Maxima.
(`http://maxima.sourceforge.net/docs/manual/en/maxima_32.html#SEC125`).
[14] A. Valibouze. Étude des relations algébriques entre les racines d'un polynôme d'une variable. *Bull. Belg. Math. Soc. Simon Stevin*, 6(4):507–535, 1999. (Version longue du rapport LIP6 1997/014).
[15] A. Valibouze. Sur les relations entre les racines d'un polynôme. *Acta Arithmetica*, 131.1:1–27, 2008.
[16] Joachim von zur Gathen. Functional decomposition of polynomials: the tame case. *J. Symbolic Comput.*, 9(3):281–299, 1990.

LIP6,UPMC, 4, place Jussieu, F-75252 Paris Cedex 05
E-mail address: `nom.prenom@upmc.fr`