Calcul rapide de résolvantes de Lagrange absolues*

Romain Lebreton
Laboratoire d'Informatique
École Polytechnique
Paris

Éric Schost
Computer Science Department
University of Western Ontario
London

Deuxième colloque franco-maghrébin de calcul formel

Îles de Kerkennah

1 octobre 2011

∗. This document has been written using the GNU TEXMACS text editor (see `www.texmacs.org`).

Présentation

Fixons k un corps effectif de caractéristique 0 ou suffisamment grande.

Fixons $f = X^n + \sum_{i=1}^{n} (-1)^i f_i X^{n-i} \in k[X]$ séparable de degré n.

Notons $\alpha_1, ..., \alpha_n$ ses racines.

Relations symétriques :

- Définition : $P \in k[X_1, ..., X_n]$ tels que $\forall \sigma \in \mathfrak{S}_n, P(\alpha_{\sigma(1)}, ..., \alpha_{\sigma(n)}) = 0$;
- Elles forment un idéal $\mathcal{I} \subseteq k[X_1, ..., X_n]$ engendré par les $(E_i(X_1, ..., X_n) - f_i)_{i=1,...,n}$.

L'algèbre de décomposition universelle est $\mathbb{A} := k[X_1, ..., X_n]/\mathcal{I}$, son degré est $\delta := n!$.

Pour tout $P \in \mathbb{A}$, notons son polynôme charactéristique

$$\mathcal{X}_{P,\mathbb{A}}(T) := \prod_{\sigma \in \mathfrak{S}_n} (T - P(\alpha_{\sigma(1)}, ..., \alpha_{\sigma(n)})) \in k[T].$$

État de l'art : calcul d'une résolvante absolue

Résolvante de Lagrange absolue :

$$L_P(T) := \prod_{\sigma \in \mathfrak{S}_n // \mathrm{Stab}\, P} (T - P(\alpha_{\sigma(1)}, ..., \alpha_{\sigma(n)})) \in k[T].$$

On a la relation $\exists r \in \mathbb{N}^*, \mathcal{X}_P = (L_P)^r$.

Méthodes symboliques pour le calcul de résolvantes absolues :

- par les résultants *cf.* [Lagrange], [Soicher, 1981], [Giusti *et al.*, 1988], [Lehobey,1997];
- par les fonctions symétriques *cf.* [Lagrange], [Valibouze, 1988], [Caperson, McKay, 1994];
- par les bases standards *cf.* [Giusti *et al.*, 1988], [Arnaudiès, Valibouze, 1993];
- par des invariants *cf.* [Berwick, 1929], [Foulkes, 1931].

$\rightsquigarrow$ Pas d'étude de complexité. Algorithmes non optimaux en $\Omega(\delta^2)$.

État de l'art : calcul dans l'algèbre de décomposition universelle

Représentation triangulaire	Représentation à une variable
Modules de Cauchy $\mathrm{MC}_i \in k[X_1, ..., X_i]$ pour $1 \leqslant i \leqslant n$.	Polynôme minimal Q d'une forme linéaire primitive Λ. Paramétrisations $(S_i(T))_{1 \leqslant i \leqslant n}$.
$\mathbb{A}_1 = k[X_1]/(\mathrm{MC}_1)$ $\vdots$ $\mathbb{A}_j = k[X_1, ..., X_j]/(\mathrm{MC}_1, ..., \mathrm{MC}_j)$ $\vdots$ $\mathbb{A} = \mathbb{A}_n = k[X_1, ..., X_n]/(\mathrm{MC}_1, ..., \mathrm{MC}_n)$	$\mathbb{A} \simeq k[T]/(Q)$ $X_i \longmapsto S_i(T)$ $\Lambda \longleftarrow\!\shortmid T$

Coût du calcul de la représentation :

$o(\delta)$ par la formule récursive : $\mathrm{MC}_{i+1} = \frac{(\mathrm{MC}_i(X_1, ..., X_i) - \mathrm{MC}_i(X_1, ..., X_{i+1}))}{X_i - X_{i+1}}$	$\tilde{\mathcal{O}}(\delta^\omega)$ par l'algorithme de base de Gröbner F4 *cf.* [Faugère, 1999]. $\tilde{\mathcal{O}}(\delta^2)$ par l'algorithme de résolution géométrique *cf.* [Giusti, Lecerf, Salvy, 2001] *cf.* [Heintz *et al.*, 2000]

État de l'art : calcul dans l'algèbre de décomposition universelle

Représentation triangulaire	Représentation à une variable
Modules de Cauchy $\mathrm{MC}_i \in k[X_1, ..., X_i]$ pour $1 \leqslant i \leqslant n$.	Polynôme minimal Q d'une forme linéaire primitive Λ. Paramétrisations $(S_i(T))_{1 \leqslant i \leqslant n}$.
$\mathbb{A}_1 = k[X_1]/(\mathrm{MC}_1)$ $\vdots$ $\mathbb{A}_j = k[X_1, ..., X_j]/(\mathrm{MC}_1, ..., \mathrm{MC}_j)$ $\vdots$ $\mathbb{A} = \mathbb{A}_n = k[X_1, ..., X_n]/(\mathrm{MC}_1, ..., \mathrm{MC}_n)$	$\mathbb{A} \simeq k[T]/(Q)$ $X_i \longmapsto S_i(T)$ $\Lambda \longleftarrow\!\shortmid T$

Coût des opérations de base :

- multiplication $\tilde{\mathcal{O}}(\delta)$ [Bostan et *al.*, 2009], MAIS non implémenté, grande constante	$\tilde{\mathcal{O}}(\delta)$, simple et efficace
- division (si possible) pas d'algorithme quasi-optimal	$\tilde{\mathcal{O}}(\delta)$, simple et efficace
	+ calcul facile de forme normale, trace ...

Résultats

Théorème. *Pour tout $P \in \mathbb{A}$, le polynôme charactéristique $\mathcal{X}_P \in k[T]$ peut être calculé en*

$$\mathcal{O}(n^2\,\mathrm{M}(\delta)) = \tilde{\mathcal{O}}(\delta)$$

opérations arithmétiques dans k.

Théorème. *Supposons donnée une forme linéaire primitive Λ. Alors une représentation à une variable $\mathfrak{P} = (\Lambda, Q, S_1, ..., S_n) \in k[X_1, ..., X_n] \times k[T]^{n+1}$ où*

$$\begin{array}{ccc} \mathbb{A} = k[X_1, ..., X_n]/\mathcal{I} & \simeq & k[T]/Q(T) \\ X_i & \mapsto & S_i(T) \\ \Lambda(X_1, ..., X_n) & \leftarrow\!\shortmid & T \end{array}$$

se calcule en complexité arithmétique $\mathcal{O}(n^3\,\mathrm{M}(\delta)) = \tilde{\mathcal{O}}(\delta)$.

Remarque 1. L'article [Colin, Giusti, *en cours*] nous permet de rendre le choix de Λ déterministe non-uniforme.

Applications

- Calcul du polynôme charactéristique $\mathcal{X}_P$
  - $\rightsquigarrow$ Calcul symbolique de résolvantes de Lagrange absolues en complexité $\tilde{\mathcal{O}}(\delta)$

- Calcul d'une représentation $\mathbb{A} \simeq k[T]/Q(T)$ à une variable
  - $\rightsquigarrow$ Algorithmes pour les opérations arithmétiques en $\tilde{\mathcal{O}}(\delta)$ opérations dans k.
  - $\rightsquigarrow$ Algorithmes efficaces pour le calcul de trace, polynôme minimum...
  - $\rightsquigarrow$ Corps de décomposition dynamique, *cf.* [Della Dora *et al.*, 1985]
  - $\rightsquigarrow$ Calcul d'une représentation de l'idéal galoisien alterné, *cf.* [Valibouze, 2010]

Algorithme 1 - PolynomeCharacteristique :

Entrée :

- un polynôme $f \in k[T]$;
- un polynôme $P \in k[X_1, ..., X_n]$ réduit;
- pour tout $1 \leqslant i \leqslant n$, une représentation à une variable $\mathfrak{P}_i = (\Lambda_i, Q_i, S_{i,1}, ..., S_{i,n})$ de $\mathbb{A}_i := \mathbb{A} \cap k[X_1, ..., X_i]$ tel que

$$\forall i \in \{1, ..., n\}, \exists \lambda_{i+1} \in k, \quad \Lambda_{i+1} = \Lambda_i + \lambda_{i+1} X_{i+1}. \tag{1}$$

Sortie :

- Le polynôme charactéristique

$$\mathcal{X}_P(T) := \prod_{\sigma \in \mathfrak{S}_n} (T - P(\alpha_{\sigma(1)}, ..., \alpha_{\sigma(n)})).$$

Algorithme `PolynomeCharacteristique` - Principe

Basé sur l'approche par résultants du calcul de résolvantes :

- $R_{n+1} := T - P(X_1, ..., X_n) \in k[X_1, ..., X_n, T]$
- $\forall i \in \{1, ..., n\}, \quad R_i := \mathrm{Res}_{X_{i+1}}(\mathrm{MC}_{i+1}, R_{i+1}) \in k[X_1, ..., X_i, T]$
- $\mathcal{X}_P = R_0 \in k[T]$.

Mathématiquement,

$$\mathrm{Res}_{X_{i+1}} : \quad \mathbb{A}_i[T][X_{i+1}] \times \mathbb{A}_i[T][X_{i+1}] \longrightarrow \mathbb{A}_i[T].$$

Algorithmiquement,

$$\mathrm{Res}_{X_{i+1}} : \quad (k[Z_i]/Q_i)[T][X_{i+1}] \times (k[Z_i]/Q_i)[T][X_{i+1}] \longrightarrow (k[Z_i]/Q_i)[T].$$

Lemme. *Chaque calcul de résultant coûte* $\tilde{\mathcal{O}}(\delta)$ *opérations arithmétiques dans* k.

Algorithme `PolynomeCharacteristique` - Changement de représentation

Rappel :

- $R_i = \mathrm{Res}_{X_{i+1}}(\mathrm{MC}_{i+1}, R_{i+1})$
- $R_{i+1} \in \mathbb{A}_{i+1}[T]$
- $\mathrm{Res}_{X_{i+1}} : \quad \mathbb{A}_i[T][X_{i+1}] \times \mathbb{A}_i[T][X_{i+1}] \longrightarrow \mathbb{A}_i[T]$

Changement de représentation :

$$\begin{array}{rccc} & \mathbb{A}_{i+1} & \longrightarrow & \mathbb{A}_i[X_{i+1}]/(\mathrm{MC}_{i+1}) \\ & \downarrow & & \downarrow \\ \mathrm{desc}_i : & k[Z_{i+1}]/(Q_{i+1}) & \longrightarrow & k[Z_i, X_{i+1}]/(Q_i(Z_i), Q_{i,i+1}) \ . \\ & Z_{i+1} & \longmapsto & Z_i + \lambda_{i+1} X_{i+1} \end{array}$$

Lemme. *Le calcul de* $\mathrm{desc}_i(R_{i+1})$ *se fait en complexité* $\tilde{\mathcal{O}}(\delta)$.

Proposition. *L'algorithme* `PolynomeCharacteristique` *a une complexité arithmétique quasi-optimale* $\tilde{\mathcal{O}}(\delta)$.

Algorithme 2 - `EtapeRepresentation` :

Entrée :

- un polynôme $f \in k[T]$;
- une forme linéaire primitive Λ_j de $\mathbb{A}_j$;
- pour tout $1 \leqslant i \leqslant j-1$, une représentation à une variable $\mathfrak{P}_i = (\Lambda_i, Q_i, S_{i,1}, ..., S_{i,n})$ de $\mathbb{A}_i$ tel que

$$\forall i \in \{1, ..., j-1\}, \exists \lambda_{i+1} \in k, \quad \Lambda_{i+1} = \Lambda_i + \lambda_{i+1} X_{i+1}. \tag{2}$$

Sortie :

- une représentation à une variable $\mathfrak{P}_j = (\Lambda_j, Q_j, S_{j,1}, ..., S_{j,n})$ de $\mathbb{A}_j$.

Algorithme `EtapeRepresentation` - Calcul d'une représentation à une variable

Lien entre polynôme charactéristique et représentation à une variable :

- polynôme minimal : $Q_j(T) = \mathcal{X}_{\Lambda_j}(T)$;
- paramétrisations :

Lemme. *Si* $K := k[T_1, ..., T_n]$ *et* $\Lambda := T_1 X_1 + \cdots + T_n X_n \in K[X_1, ..., X_n]$. *Ainsi* $\mathcal{X}_\Lambda \in k[T, T_1, ..., T_n]$ *et*

$$X_i \frac{\partial \mathcal{X}_\Lambda}{\partial T} + \frac{\partial \mathcal{X}_\Lambda}{\partial T_i} = 0 \text{ dans } \mathbb{A}.$$

En pratique, on utilise les nombres tangents $K := k[\varepsilon]/(\varepsilon^2)$ pour calculer les dérivées.

Proposition. *Étant donné un élément primitif* $\Lambda \in \mathbb{A}$, *le coût du calcul d'une représentation à une variable de* $\mathbb{A}$ *est* $\mathcal{O}(n\,\mathcal{C})$ *où* $\mathcal{C}$ *est le coût d'un polynôme charactéristique.*

Algorithme `EtapeRepresentation` - Formule récursive

Définition. *Soient* $f, g \in k[T]$ *qui s'écrivent* $f = \prod_{i=1,\dots,n} (T - \alpha_i)$, $g = \prod_{j=1,\dots,m} (T - \beta_j)$ *dans* $\bar{k}$. *Alors on définit* $f \oplus g \in k[T]$ *(resp.* $f \otimes g \in k[T]$*) par*

$$f \oplus g := \prod_{1 \leqslant i \leqslant n, 1 \leqslant j \leqslant m} (T - (\alpha_i + \beta_j));$$
$$(\text{resp.}) \quad f \otimes g := \prod_{1 \leqslant i \leqslant n, 1 \leqslant j \leqslant m} (T - (\alpha_i \cdot \beta_j)).$$

Proposition. (Formule récursive) *Nous avons la relation*

$$\mathcal{X}_{\Lambda_j, \mathbb{A}_j}(T) = \frac{\mathcal{X}_{\Lambda_{j-1}, \mathbb{A}_{j-1}}(T) \oplus (f \otimes (T - \lambda_j))}{\prod_{i=1}^{j-1} \mathcal{X}_{\Lambda_{j-1} + \lambda_j X_i, \mathbb{A}_{j-1}}(T)}. \tag{3}$$

Proposition. *L'algorithme* `EtapeRepresentation` *a une complexité quasi-optimale* $\tilde{\mathcal{O}}(\delta)$.

Algorithme 3 - Representation :

Entrée :

- un polynôme $f \in k[T]$;
- une forme linéaire primitive $\Lambda := X_1 + \lambda_2 X_2 + \cdots + \lambda_n X_n$ de $\mathbb{A}$;

Sortie :

- une représentation à une variable $\mathfrak{P}_i = (\Lambda_i, Q_i, S_{i,1}, ..., S_{i,n})$ de $\mathbb{A}$.

Algorithme :

- Représentation $\mathfrak{P}_1 = (X_1, f(T), T)$ de $\mathbb{A}_1$;
- `for i=2..n do`

  $\mathfrak{P}_i \longleftarrow$ `EtapeRepresentation`$(f, X_1 + \lambda_2 X_2 + \cdots + \lambda_i X_i, \mathfrak{P}_{i-1})$
- `return` $\mathfrak{P}_n$

$\rightsquigarrow$ Complexité $\tilde{\mathcal{O}}(\delta)$

Conclusion

Résultats théoriques :

- algorithme efficace pour les calculs dans l'algèbre de décomposition universelle;
- algorithme efficace pour le calcul de polynôme charactéristique;
- amélioration de complexité pour le calcul symbolique de résolvantes absolues.

En pratique :

- Code MAGMA en cours;
- algorithme différent pour la représentation à une variable de $\mathbb{A}$.

Merci pour votre attention ;-)