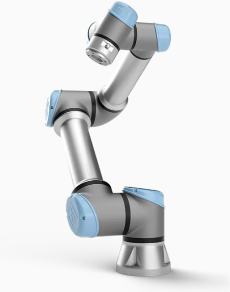
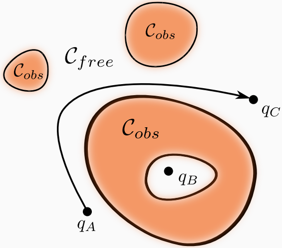
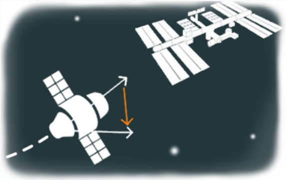
Rémi Prébet

Équipe-projet : AriC






```
@real: i, s, n;
@pre: n ≥ 1;
i = 0; s = 0;
@invariant: [[{i,s},2]] ≥ 0 ∧ [[{i,s},2]] ≥ 0
while( [[{i,n},1]] )
{
        i = i+1;
        s = [[{i,s},1]];
}
@post: (n − 1) · n/2 ≤ s ≤ n · (n+1)/2;
```



systèmes polynomiaux sur $\mathbb{R}$ - planification de mouvement - calculs certifiés

Positionnement scientifique : le calcul formel

Chaîne de résolution du calcul formel

$\Longrightarrow$ Modé-lisation

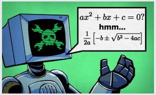
Problèmes mathématiques

$\Longrightarrow$ **Expertise**

Algorithmes

$\Longrightarrow$ **Logiciel**

Solutions exactes



Systèmes polynomiaux	**Systèmes différentiels**	**Algorithmique socle**
cryptologie, robotique	physique, preuves math.	algèbre linéaire, arithmétique
Vision par ordinateur	Conjecture de Kepler	Produit entier en $O(n \log n)$

Numérique

- Efficacité
- Approximations
- Temps réel

Symbolique

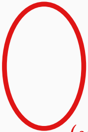
- Réponses exactes
- Coût intrinsèque
- Hors-ligne

Algorithmique des systèmes polynomiaux à variables réelles

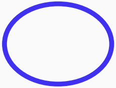
Ensembles semi-algébriques

Solutions **réelles** de systèmes d'**équations** et d'**inégalités polynomiales**

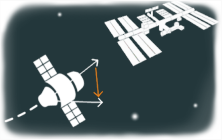
$$\begin{cases} 4y + x^3 - 4x^2 - 2x - 8 = 0 \\ -2 \leq x \leq 0 \end{cases}$$



$$\frac{x^2}{4} + y^2 - 1 = 0$$



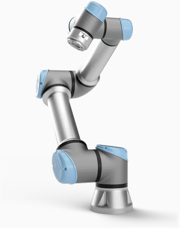
$$(x-1)^2 + \frac{(y-1)^2}{9} - 1 = 0$$



Calcul numérique certifié

```
@real: i, s, n;
@pre: n ≥ 1;
i = 0; s = 0;
@invariant: [[{i,s},2]] ≥ 0 ∧ [[{i,s},2]] ≥ 0
while( [[{i,n},1]] )
{
        i = i+1;
        s = [[{i,s},1]];
}
@post: (n−1)·n/2 ≤ s ≤ n·(n+1)/2;
```

Vérification de programmes

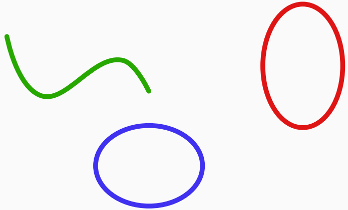


Robotique

Algorithmique des systèmes polynomiaux à variables réelles

Ensembles semi-algébriques

Solutions **réelles** de systèmes d'<u>**équations**</u> et d'<u>**inégalités polynomiales**</u>



Problèmes algorithmiques fondamentaux

Projeter : quel est l'ensemble des valeurs possibles ?

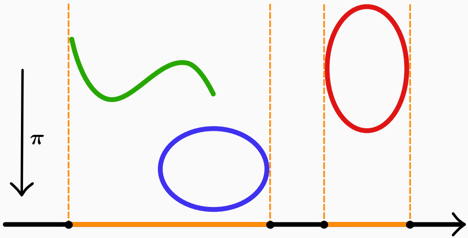
Échantillonner : existe-t-il des solutions ?

Connecter : deux points sont-ils connectés ?

Algorithmique des systèmes polynomiaux à variables réelles

Ensembles semi-algébriques

Solutions **réelles** de systèmes d'**équations** et d'**inégalités polynomiales**

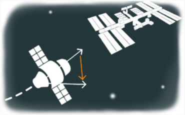


Problèmes algorithmiques fondamentaux

Projeter : quel est l'ensemble des valeurs possibles ?

Échantillonner : existe-t-il des solutions ?

Connecter : deux points sont-ils connectés ?

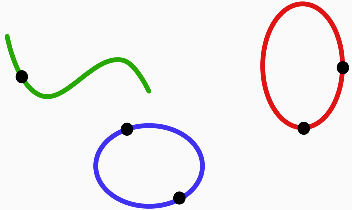


Calcul numérique certifié

Algorithmique des systèmes polynomiaux à variables réelles

Ensembles semi-algébriques

Solutions **réelles** de systèmes d'**équations** et d'**inégalités polynomiales**



Problèmes algorithmiques fondamentaux

Projeter : quel est l'ensemble des valeurs possibles ?

Échantillonner : existe-t-il des solutions ?

Connecter : deux points sont-ils connectés ?

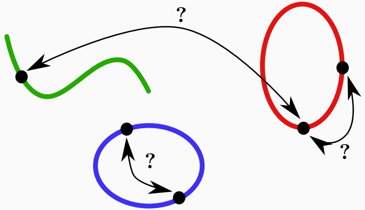
```
@real: i, s, n;
@pre: n ≥ 1;
i = 0; s = 0;
@invariant: [[{i,s},2]] ≥ 0  ∧  [[{i,s},2]] ≥ 0
while( [[{i,n},1]] )
{
        i = i+1;
        s = [[{i,s},1]];
}
@post: (n−1)·n/2 ≤ s ≤ n·(n+1)/2;
```

Vérification de programmes

Algorithmique des systèmes polynomiaux à variables réelles

Ensembles semi-algébriques

Solutions **réelles** de systèmes d'**équations** et d'**inégalités polynomiales**

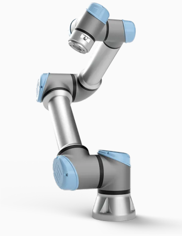


Problèmes algorithmiques fondamentaux

Projeter : quel est l'ensemble des valeurs possibles ?

Échantillonner : existe-t-il des solutions ?

Connecter : deux points sont-ils connectés ?

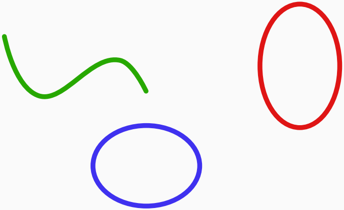


Robotique

Algorithmique des systèmes polynomiaux à variables réelles

Ensembles semi-algébriques

Solutions **réelles** de systèmes d'**équations** et d'**inégalités polynomiales**

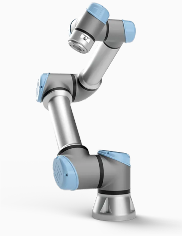


Problèmes algorithmiques fondamentaux

Projeter : quel est l'ensemble des valeurs possibles ?

Échantillonner : existe-t-il des solutions ?

Connecter : deux points sont-ils connectés ?



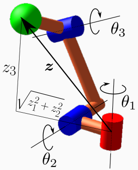
Robotique

I. Contributions

Contribution 1 - Robotique et cuspidalité

Application cinématique

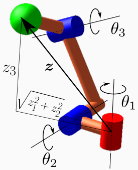
$$\begin{array}{rccc} \mathcal{R}: & \mathbb{R}^d & \to & \mathbb{R}^d \\ & (\boldsymbol{\ell}, \boldsymbol{\theta}) & \mapsto & \Big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\Big) \end{array}$$



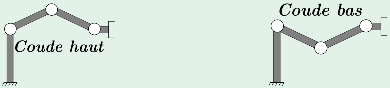
Contribution 1 - Robotique et cuspidalité

Application cinématique

$$\mathcal{R}: \quad \mathbb{R}^d \to \mathbb{R}^d, \quad (\boldsymbol{\ell}, \boldsymbol{\theta}) \mapsto \Big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \dots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\Big)$$

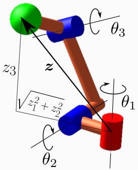


Non-cuspidal : posture contrôlée

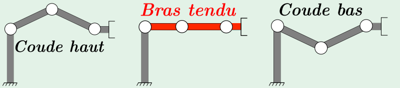


Application cinématique

$$\mathcal{R}: \begin{array}{ccc} \mathbb{R}^d & \to & \mathbb{R}^d \\ (\boldsymbol{\ell}, \boldsymbol{\theta}) & \mapsto & \big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\big) \end{array}$$



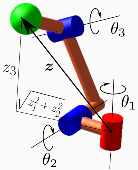
Non-cuspidal : posture contrôlée



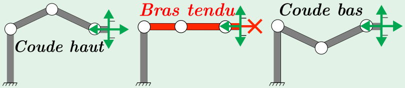
Contribution 1 - Robotique et cuspidalité

Application cinématique

$$\mathcal{R}: \quad \mathbb{R}^d \to \mathbb{R}^d, \qquad (\boldsymbol{\ell}, \boldsymbol{\theta}) \mapsto \big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\big)$$



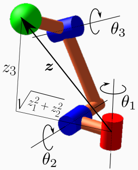
Non-cuspidal : posture contrôlée



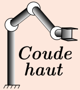
Contribution 1 - Robotique et cuspidalité

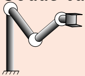
Application cinématique

$$\mathcal{R}: \quad \mathbb{R}^d \to \mathbb{R}^d$$
$$(\boldsymbol{\ell}, \boldsymbol{\theta}) \mapsto \Big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\Big)$$



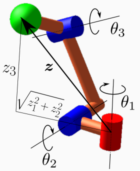
Cuspidal : posture non-contrôlée



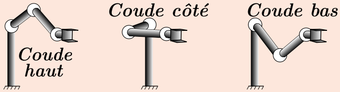


Application cinématique

$$\begin{array}{rccc} \mathcal{R}: & \mathbb{R}^d & \to & \mathbb{R}^d \\ & (\boldsymbol{\ell}, \boldsymbol{\theta}) & \mapsto & \Big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\Big) \end{array}$$

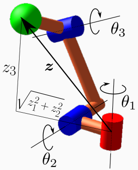


Cuspidal : posture non-contrôlée

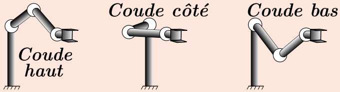


Application cinématique

$$\mathcal{R}: \quad \mathbb{R}^d \to \mathbb{R}^d, \quad (\boldsymbol{\ell}, \boldsymbol{\theta}) \mapsto \big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\big)$$



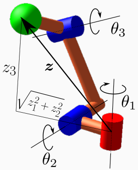
Cuspidal : posture non-contrôlée



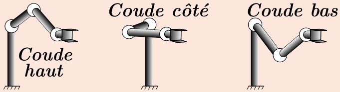
Pb ouvert (1983→2022) : test de cuspidalité

Application cinématique

$$\begin{array}{rccc} \mathcal{R}: & \mathbb{R}^d & \rightarrow & \mathbb{R}^d \\ & (\boldsymbol{\ell}, \boldsymbol{\theta}) & \mapsto & \left(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\right) \end{array}$$



Cuspidal : posture non-contrôlée



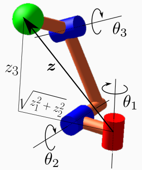
Pb ouvert (1983→2022) : test de cuspidalité

- Premier algorithme général NEW!
- Publi. ISSAC'22 (1ère conf. calcul formel)

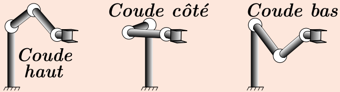
Contribution 1 - Robotique et cuspidalité

Application cinématique

$$\mathcal{R}: \quad \begin{array}{ccc} \mathbb{R}^d & \rightarrow & \mathbb{R}^d \\ (\boldsymbol{\ell}, \boldsymbol{\theta}) & \mapsto & \Big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta}) \Big) \end{array}$$



Cuspidal : posture non-contrôlée



Pb ouvert (1983→2022) : test de cuspidalité

Premier algorithme général NEW!

Publi. ISSAC'22 (1$^{\text{ère}}$conf. calcul formel)

Logiciel

Implantation prototype

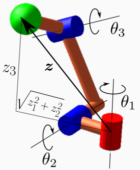
Impact

Citation ICRA & IROS (1$^{\text{ères}}$confs. robotique)

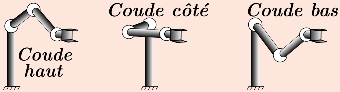
Contribution 1 - Robotique et cuspidalité

Application cinématique

$$\mathcal{R}: \quad \begin{array}{ccc} \mathbb{R}^d & \rightarrow & \mathbb{R}^d \\ (\boldsymbol{\ell}, \boldsymbol{\theta}) & \mapsto & \big(z_1(\boldsymbol{\ell}, \boldsymbol{\theta}), \ldots, z_d(\boldsymbol{\ell}, \boldsymbol{\theta})\big) \end{array}$$



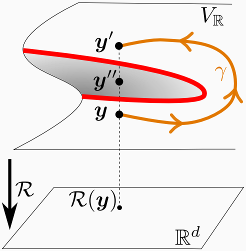
Cuspidal : posture non-contrôlée



Pb ouvert (1983→2022) : test de cuspidalité

Premier algorithme général NEW!

Publi. ISSAC'22 ($1^{\text{ère}}$ conf. calcul formel)



Logiciel

Implantation prototype

Impact

Citation ICRA & IROS ($1^{\text{ères}}$ confs. robotique)

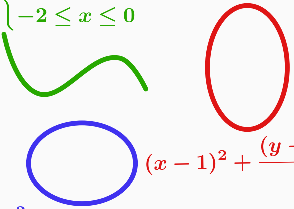
Goulot d'étranglement

Tests de connexité

Ensemble semi-algébrique S

Défini par s polynômes (équations+inégalités) à n variables de deg $\leq D$

$$\begin{cases} 4y + x^3 - 4x^2 - 2x - 8 = 0 \\ -2 \leq x \leq 0 \end{cases}$$



$$(x-1)^2 + \frac{(y-1)^2}{9} - 1 = 0$$

$$\frac{x^2}{4} + y^2 - 1 = 0$$

Algorithme généraliste [Collins; '75]

Complexité : $(sD)^{2^{O(n)}}$

Problèmes algorithmiques fondamentaux

Projeter S sur t coordonnées $\rightsquigarrow (sD)^{O(nt)}$ [Renegar, '92] **Optimal !**

Échantillonner des points de S $\rightsquigarrow (sD)^{O(n)}$ [Basu-Pollack-Roy, '98] **Optimal !**

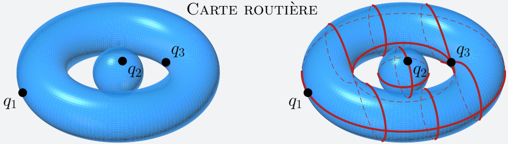
Connecter deux points dans S $\rightsquigarrow (sD)^{O(n^2)}$ [Canny, '88] **Pas optimal !**

Contribution 2 - Requêtes de connexité, cartes routières

[Canny, 1988] **Carte routière** : courbe capturant la connexité de l'ensemble

Chaîne de réduction

Dimension arbitraire $\xrightarrow[\textsc{Carte routière}]{}$ Dimension 1



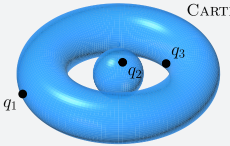
Contribution 2 - Requêtes de connexité, cartes routières

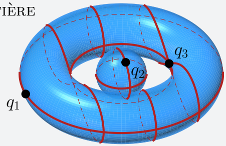
[Canny, 1988] **Carte routière** : courbe capturant la connexité de l'ensemble

Chaîne de réduction

Dimension arbitraire $\Longrightarrow$ Dimension 1

Carte routière





Vers l'optimalité

1982 $D^{2^{O(n)}}$

$\downarrow$

1988 $(nD)^{O(n^2)}$

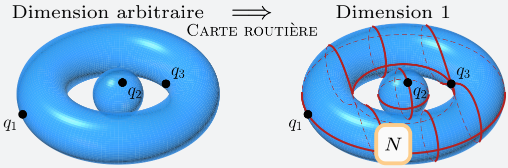
$\downarrow$

$(nD)^{O(n)}$

Contribution 2 - Requêtes de connexité, cartes routières

[Canny, 1988] **Carte routière** : courbe capturant la connexité de l'ensemble

Chaîne de réduction



Vers l'optimalité

1982 $D^{2^{O(n)}}$

↓

1988 $(nD)^{O(n^2)}$

↓

$(nD)^{O(n)}$

Résultat de complexité (2024) NEW! (cas lisse et **non-borné**)

On peut calculer une carte routière de taille $N = (n^2 D)^{4n \log_2 n + O(n)}$ en $O(N^{1.5})$ ops

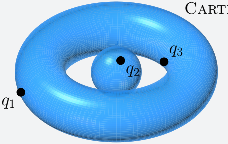
Contribution 2 - Requêtes de connexité, cartes routières

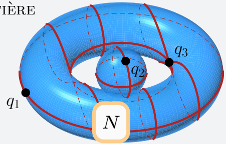
[Canny, 1988] **Carte routière** : courbe capturant la connexité de l'ensemble

Chaîne de réduction

Dimension arbitraire $\Longrightarrow$ Dimension 1

Carte routière





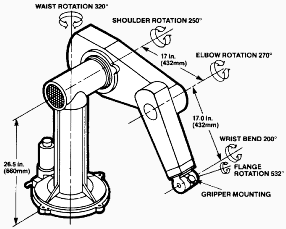
Vers l'optimalité

1982	$D^{2^{O(n)}}$
1988	$(nD)^{O(n^2)}$
	$(nD)^{O(n)}$

Résultat de complexité (2024) NEW!

(cas lisse et **non-borné**)

On peut calculer une carte routière de taille $N = (n^2D)^{\mathbf{4n\log_2 n}+O(n)}$ en $O(N^{1.5})$ ops



Puma 560 [Unimation, 1984]

1 Publication (2024) + 1 Soumission (2024)

JSC (1$^{\text{er}}$ journal calcul formel)

Calcul d'une carte routière NEW!

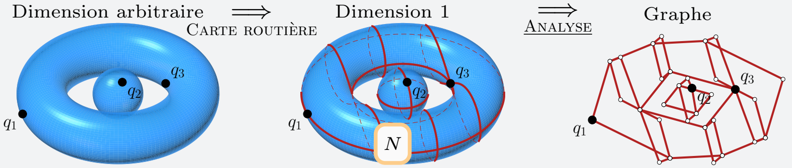
Sortie : polynômes de deg max **200**
1 polynôme ≈ 20 000 coeffs
1 coeff ≈ 3000 chiffres

Temps : **4h10**

Contribution 3 - Requêtes de connexité, courbes algébriques

Graphe de connexité : graphe capturant la connexité de la carte routière

Chaîne de réduction

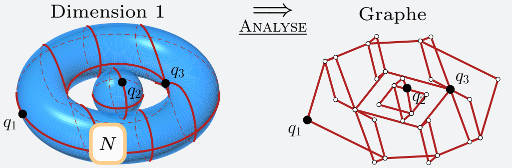


Contribution 3 - Requêtes de connexité, courbes algébriques

Graphe de connexité : graphe capturant la connexité de la carte routière

État de l'art : géométrie

Espace	Complexité
$\mathbb{R}^2$	$\tilde{O}(N^3)$
$\mathbb{R}^3$	$\tilde{O}(N^9)$
$\mathbb{R}^n$	$N^{O(1)}$

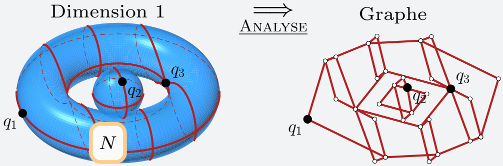


Contribution 3 - Requêtes de connexité, courbes algébriques

Graphe de connexité : graphe capturant la connexité de la carte routière

État de l'art : géométrie

Espace	Complexité
$\mathbb{R}^2$	$\tilde{O}(N^3)$
$\mathbb{R}^3$	$\tilde{O}(N^9)$
$\mathbb{R}^n$	$N^{O(1)}$



Problème ouvert (1988 → 2023) : éviter le calcul exhaustif de la géométrie NEW!

On peut calculer un graphe de connexité d'une courbe de taille N en $\tilde{O}(N^3)$ opérations

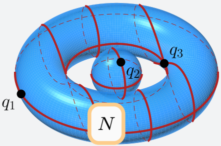
Contribution 3 - Requêtes de connexité, courbes algébriques

Graphe de connexité : graphe capturant la connexité de la carte routière

État de l'art : géométrie

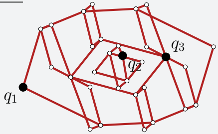
Espace	Complexité
$\mathbb{R}^2$	$\tilde{O}(N^3)$
$\mathbb{R}^3$	$\tilde{O}(N^9)$
$\mathbb{R}^n$	$N^{O(1)}$

Dimension 1



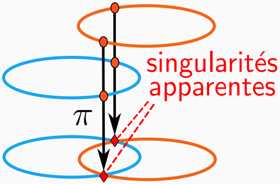
$\Longrightarrow$ ANALYSE

Graphe



Problème ouvert (1988 → 2023) : éviter le calcul exhaustif de la géométrie NEW!

On peut calculer un graphe de connexité d'une courbe de taille N en $\tilde{O}(N^3)$ opérations



Publication (2023)

ISSAC ($1^{\text{ère}}$ conférence calcul formel)

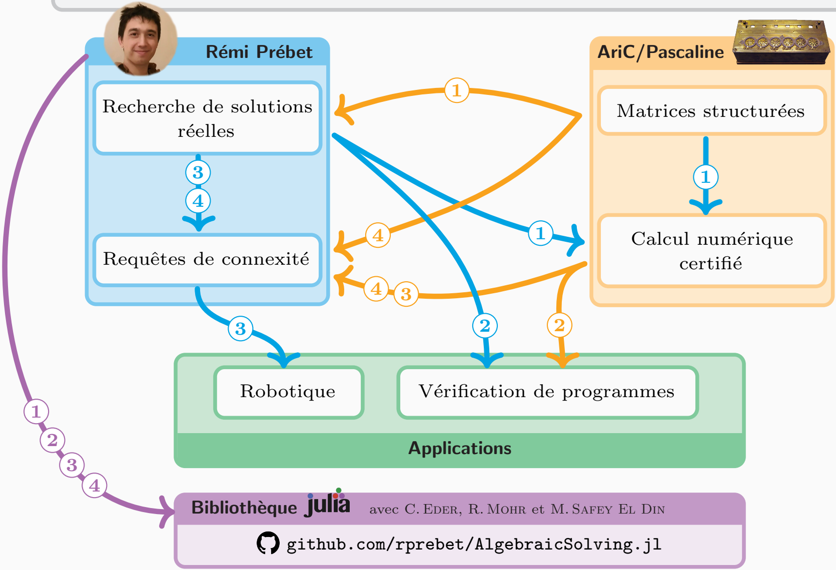
Logiciel (auparavant deg≈20) NEW!

Degré 100 --→ Degré 200

rprebet/AlgebraicSolving.jl

II. Programme de recherche

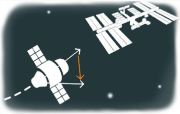
Calcul numérique certifié et robotique par la géométrie et l’algorithmique des systèmes structurés



Axe 1 - Analyse d'erreur flottante par l'optimisation polynomiale

Évaluation de fonction par un programme

Borne <u>fine</u> sur l'erreur pire-cas en fonction de la précision p

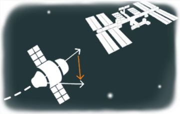


- Garanties pour applications critiques
- Choix éclairé : précision, programme
- Preuves mathématiques (conjecture de Kepler)

$+,\ -,\ \times,\ \div,\ \sqrt{\cdot}$

$\exp,\ \sqrt{x^2 + y^2},\ \ldots$

Axe 1 - Analyse d'erreur flottante par l'optimisation polynomiale



Évaluation de fonction par un programme

Borne <u>fine</u> sur l'erreur pire-cas en fonction de la précision p

- Garanties pour applications critiques
- Choix éclairé : précision, programme
- Preuves mathématiques (conjecture de Kepler)

$+,\ -,\ \times,\ \div,\ \sqrt{\cdot}$

exp, $\sqrt{x^2+y^2}$, ...

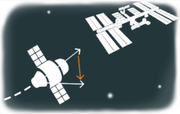
Prog. d'évaluation de $\sqrt{x^2+y^2}$

```
sx = x*x;
sy = y*y;
sigma = sx + sy;
rho = sqrt(sigma);
```

Axe 1 - Analyse d’erreur flottante par l’optimisation polynomiale

Évaluation de fonction par un programme

Borne <u>fine</u> sur l’erreur pire-cas en fonction de la précision p



- Garanties pour applications critiques
- Choix éclairé : précision, programme
- Preuves mathématiques (conjecture de Kepler)

$+,\ -,\ \times,\ \div,\ \sqrt{\cdot}$

$\exp,\ \sqrt{x^2+y^2},\ \ldots$??

Prog. d’évaluation de $\sqrt{x^2+y^2}$

```
sx = x*x;
sy = y*y;
sigma = sx + sy;
rho = sqrt(sigma);
```

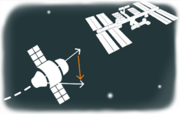
Modélisation algébrique de l’erreur d’arrondi

$$
\begin{array}{lll}
1: & s_x = x^2(1+u\varepsilon_{s_x}) & |\varepsilon_{s_x}| \le b_\times \\
2: & s_y = y^2(1+u\varepsilon_{s_y}) & |\varepsilon_{s_y}| \le b_\times \\
3: & \sigma = (s_x+s_y)(1+u\varepsilon_\sigma) & |\varepsilon_{s_\sigma}| \le b_+ \\
4: & \rho^2 = \sigma(1+u\varepsilon_\rho)^2 & \rho \ge 0,\ |\varepsilon_\rho| \le b_\surd
\end{array}
$$

Axe 1 - Analyse d'erreur flottante par l'optimisation polynomiale

Évaluation de fonction par un programme

Borne fine sur l'erreur pire-cas en fonction de la précision p



- Garanties pour applications critiques
- Choix éclairé : précision, programme
- Preuves mathématiques (conjecture de Kepler)

$+,\ -,\ \times,\ \div,\ \sqrt{\cdot}$

$\exp,\ \sqrt{x^2+y^2},\ \ldots$??

Prog. d'évaluation de $\sqrt{x^2+y^2}$

```
1: sx = x*x;
2: sy = y*y;
3: sigma = sx + sy;
4: rho = sqrt(sigma);
```

→

Modélisation algébrique de l'erreur d'arrondi

1: $s_x = x^2(1+u\varepsilon_{s_x})$ $\quad |\varepsilon_{s_x}| \le b_\times$

2: $s_y = y^2(1+u\varepsilon_{s_y})$ $\quad |\varepsilon_{s_y}| \le b_\times$

3: $\sigma = (s_x+s_y)(1+u\varepsilon_\sigma)$ $\quad |\varepsilon_{s_\sigma}| \le b_+$

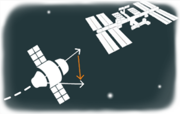
4: $\rho^2 = \sigma(1+u\varepsilon_\rho)^2$ $\quad \rho \ge 0,\ |\varepsilon_\rho| \le b_{\sqrt{}}$

Optimisation polynomiale structurée
[Muller & Salvy, 2024]

$$(a,b) \in \mathbb{R}^2 \text{ t.q. } \left|\frac{\rho}{\sqrt{x^2+y^2}}\right| \le au + bu^2$$

$(u = 2^{-p})$

Axe 1 - Analyse d'erreur flottante par l'optimisation polynomiale



Évaluation de fonction par un programme

Borne <u>fine</u> sur l'erreur pire-cas en fonction de la précision p

- Garanties pour applications critiques
- Choix éclairé : précision, programme
- Preuves mathématiques (conjecture de Kepler)

$+,\ -,\ \times,\ \div,\ \sqrt{\cdot}$

$\exp,\ \sqrt{x^2+y^2},\ \ldots$

Prog. d'évaluation de $\sqrt{x^2+y^2}$

```
1: sx = x*x;
2: sy = y*y;
3: sigma = sx + sy;
4: rho = sqrt(sigma);
```

Modélisation algébrique de l'erreur d'arrondi

$$
\begin{array}{lll}
1: & s_x = x^2(1+u\varepsilon_{s_x}) & |\varepsilon_{s_x}| \le b_\times \\
2: & s_y = y^2(1+u\varepsilon_{s_y}) & |\varepsilon_{s_y}| \le b_\times \\
3: & \sigma = (s_x+s_y)(1+u\varepsilon_\sigma) & |\varepsilon_{s_\sigma}| \le b_+ \\
4: & \rho^2 = \sigma(1+u\varepsilon_\rho)^2 & \rho \ge 0,\ |\varepsilon_\rho| \le b_{\sqrt{}}
\end{array}
$$

Optimisation polynomiale <u>structurée</u>
[Muller & Salvy, 2024]

$$(a,b) \in \mathbb{R}^2 \text{ t.q. } \left|\frac{\rho}{\sqrt{x^2+y^2}}\right| \le au + bu^2$$

$(u = 2^{-p})$

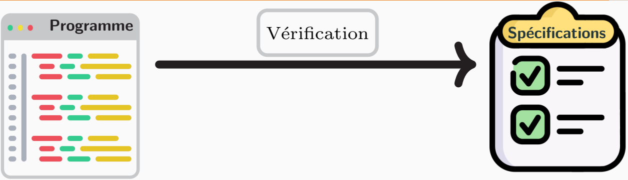
Projection : comptage de racines réelles

Matrices polynomiales <u>structurées</u>

Jeannerod & Villard

Axe 2 - Vérification de programmes et systèmes structurés

Axe 2 - Vérification de programmes et systèmes structurés



Axe 2 - Vérification de programmes et systèmes structurés

Vérification

Terminaison de boucle ?

$(\mathtt{y1}, \mathtt{y0}, \mathtt{n}) = (y_1, y_0, 0);$

$\mathtt{while}\ \mathtt{y1} \geq 0\ \mathtt{do}$

$\quad \mathtt{y1},\ \mathtt{y0} = \mathtt{n} * \mathtt{y1} - \mathtt{y0},\ \mathtt{y1};$

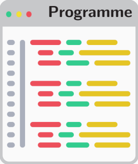
$\quad \mathtt{n}{+}{+};$

$\mathtt{end\ while}$

Suite récurrente positive ?

$$y_{n+2} = n \cdot y_{n+1} - y_n$$

A.Ibrahim & B.Salvy
2024



Vérification

Terminaison de boucle ?

$(\mathtt{y1}, \mathtt{y0}, \mathtt{n}) = (y_1, y_0, 0);$

$\mathtt{while}\ \mathtt{y1} \geq 0\ \mathtt{do}$

$\quad \mathtt{y1},\ \mathtt{y0} = \mathtt{n} * \mathtt{y1} - \mathtt{y0},\ \mathtt{y1};$

$\quad \mathtt{n}{+}{+};$

$\mathtt{end\ while}$

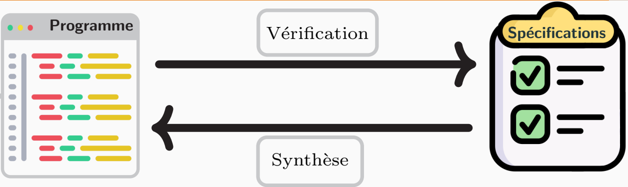
Suite récurrente positive ?

$$y_{n+2} = n \cdot y_{n+1} - y_n$$

A.Ibrahim & B.Salvy
2024

Résoudre un système polynomial **structuré**

Axe 2 - Vérification de programmes et systèmes structurés



Terminaison de boucle ?

$(\mathtt{y1}, \mathtt{y0}, \mathtt{n}) = (y_1, y_0, 0);$
$\mathtt{while}\ \mathtt{y1} \geq 0\ \mathtt{do}$
$\quad \mathtt{y1},\ \mathtt{y0} = \mathtt{n} * \mathtt{y1} - \mathtt{y0},\ \mathtt{y1};$
$\quad \mathtt{n{+}{+}};$
$\mathtt{end\ while}$

Suite récurrente positive ?

$$y_{n+2} = n \cdot y_{n+1} - y_n$$

A.Ibrahim & B.Salvy
2024

Résoudre un système polynomial **structuré**

Axe 2 - Vérification de programmes et systèmes structurés

Vérification

Synthèse

Terminaison de boucle ?

$(\mathtt{y1}, \mathtt{y0}, \mathtt{n}) = (y_1, y_0, 0);$

while $\mathtt{y1} \geq 0$ **do**

$\quad \mathtt{y1},\ \mathtt{y0} = \mathtt{n} * \mathtt{y1} - \mathtt{y0},\ \mathtt{y1};$

$\quad \mathtt{n{+}{+}};$

end while

Suite récurrente positive ?

$$y_{n+2} = n \cdot y_{n+1} - y_n$$

A.Ibrahim & B.Salvy
2024

Ébauche de prog. **[F.Mohammadi]**

@real: $i,\ s,\ n;$
@pre: $n \geq 1;$
$i = 0;\ s = 0;$
@invariant: $\boxed{[\{i,s\},2]} \geq 0 \ \wedge\ \boxed{[\{i,s\},2]} \geq 0$
while($\boxed{[\{i,n\},1]}$)
{
$\qquad i = i+1;$
$\qquad s = \boxed{[\{i,s\},1]};$
}
@post: $(n-1) \cdot n/2 \leq s \leq n \cdot (n+1)/2;$

Résoudre un système polynomial **structuré**

Axe 2 - Vérification de programmes et systèmes structurés

Vérification

Synthèse

Terminaison de boucle ?

$(\mathtt{y1}, \mathtt{y0}, \mathtt{n}) = (y_1, y_0, 0);$

```
while y1 ≥ 0 do
  y1, y0 = n * y1 − y0, y1;
  n++;
end while
```

Suite récurrente positive ?

$$y_{n+2} = n \cdot y_{n+1} - y_n$$

A.Ibrahim & B.Salvy
2024

Ébauche de prog. [F.Mohammadi]

```
@real: i, s, n;
@pre: n ≥ 1;
i = 0; s = 0;
@invariant: [[{i,s},2]] ≥ 0 ∧ [[{i,s},2]] ≥ 0
while( [[{i,n},1]] )
{
        i = i + 1;
        s = [[{i,s},1]];
}
@post: (n − 1) · n/2 ≤ s ≤ n · (n + 1)/2;
```

Échantillonnage d'un ensemble semi-algébrique

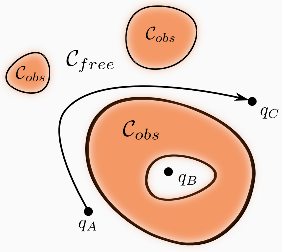
Bornes de complexité adaptées :

~~exponentielle~~ $(n_{var}) \rightarrow$ polynomiale(n_{var}) & exponentielle(n_{equ})

Résoudre un système polynomial **structuré**

Axe 3 - Planification de mouvement de robot à paramètres avec obstacles

Obstacles/Auto intersections

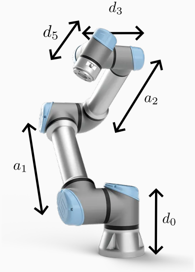


Inégalités $\Rightarrow$ semi-algébriques

Cuspidalité

Classification paramétrique
$\leftrightarrow$ directives de conception

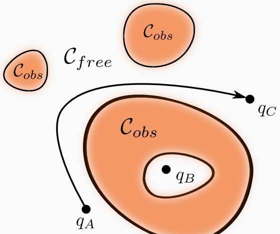
Famille de robots



Systèmes paramétrés

Axe 3 - Planification de mouvement de robot à paramètres avec obstacles

Obstacles/Auto intersections



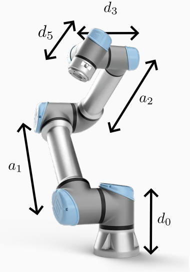
Inégalités $\Rightarrow$ semi-algébriques

Cuspidalité

Classification paramétrique
$\looparrowright$directives de conception

Sous-routines
efficaces en pratique

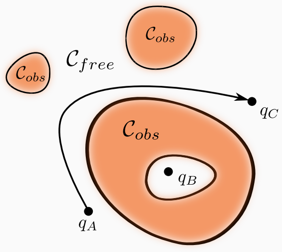
Famille de robots



Systèmes paramétrés

Axe 3 - Planification de mouvement de robot à paramètres avec obstacles

Obstacles/Auto intersections



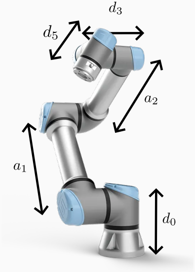
Inégalités $\Rightarrow$ semi-algébriques

Cuspidalité

Classification paramétrique
$\hookrightarrow$ directives de conception

Sous-routines
efficaces en **pratique**

Famille de robots



Systèmes paramétrés

Cartes routières : cas semi-algébrique structuré

$$f_1 = \cdots = f_p = 0, \quad g_1 > 0, \ldots, g_s > 0$$

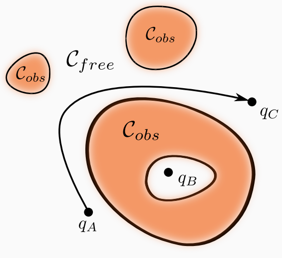
Structure des systèmes $\rightarrow$ Axe précédent

<u>État de l'art</u> : $s^n(nD)^{O(n^2)}$ [Basu, Pollack, Roy ; 2000]

<u>Objectif</u> : $\boldsymbol{s^n}(n^2D)^{4n\log_2 n + O(n)}$

Axe 3 - Planification de mouvement de robot à paramètres avec obstacles

Obstacles/Auto intersections



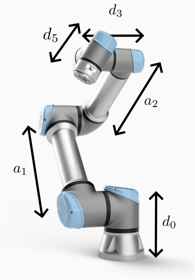
Inégalités $\Rightarrow$ semi-algébriques

Cuspidalité

Classification paramétrique
$\looparrowright$ directives de conception

Sous-routines
efficaces en pratique

Famille de robots



Systèmes paramétrés

Cartes routières : cas semi-algébrique structuré

$$f_1 = \cdots = f_p = 0, \quad g_1 > 0, \ldots, g_s > 0$$

Structure des systèmes $\rightarrow$ Axe précédent

État de l'art : $s^n (nD)^{O(n^2)}$ [Basu, Pollack, Roy ; 2000]

Objectif : $s^n (n^2 D)^{4n \log_2 n + O(n)}$

Implantations efficaces

AlgebraicSolving.jl

$\swarrow$ `msolve` $\searrow$

Isolation $\mathbb{R}$ Élimination

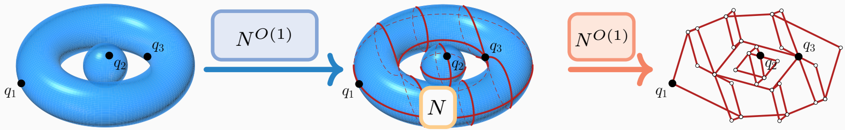
Hybride symbolique-numérique

Routines numériques certifiées

Nouvelle situation

L'analyse de la carte routière devient l'étape la plus coûteuse

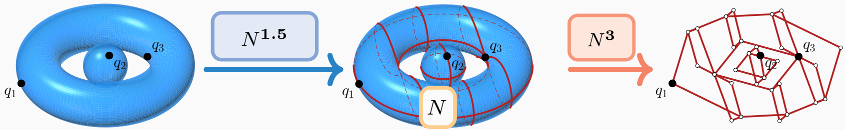
Réduction de dimension



Nouvelle situation

L'analyse de la carte routière devient l'étape la plus coûteuse

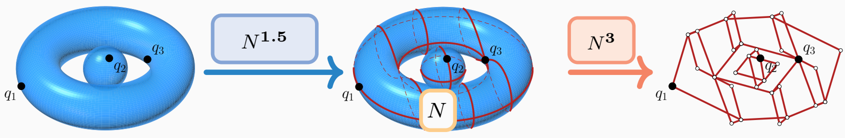
Réduction de dimension



Nouvelle situation

L'analyse de la carte routière devient l'étape la plus coûteuse

Réduction de dimension



Goulot d'étranglement

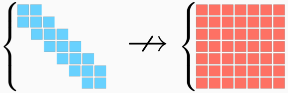
Élimination algébrique
(résultant)

Expertise mondiale
Villard

Structures des cartes routières

Algébrique

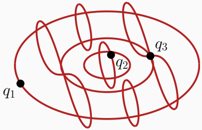
Éviter un changement de variables :



avec Bréhard, Poteaux

Université de Lille

Géométrique



Rémi Prébet

Postdoc (2024) : Vérification de programmes par les systèmes polynomiaux

Thèse (2023) : Connexité dans les ensembles algébriques réels

ÉNS Cachan/Paris-Saclay (2015-2020)

Mobilité de 3 mois, *U. Waterloo*, Canada

Qualifié CNU 25, 26, 27 ; Agrégation math.

Encadrement **(post-doc)** : M1, M2, **Thèse**

Contributions et publications

- Algorithmes efficaces en théorie & pratique pour les requêtes de connexité
- Résultats expérimentaux, prototypes
- Applications : robotique (LS2N), vérification de programme (post-doc)

2 journaux : JSC, TGRS*
4 confs : ISSAC×(2+1†), IGARSS*
1 soumission journal : JSC

†Publication sujet post-doc (Jul. '24)

**Contributions en traitement du signal* (LISTIC)

Projet de recherche

Axe 1-2 Calcul **numérique** certifié par la résolution de systèmes structurés

Axe 3 Certificats pour la planification de mouvement de **robots** réalistes

Axe 3-4 Développement de l'**algorithmique** de la géométrie algébrique réelle :

- *(théorie)* amélioration des complexités : généralisations, structures
- *(pratique)* calcul numérique certifié, librairie Julia AlgebraicSolving.jl