Hybrid Symbolic-Numeric Methods in Polynomial Algebra

Mădălina Hodorog

Johann Radon Institute for Computational and Applied Mathematics,
Doctoral Program "Computational Mathematics"
Johannes Kepler University Linz, Austria
Research Institute for Symbolic Computation

Research Seminar, Krems-Austria
November 26-27, 2010

Table of contents

Doctoral Program Computational Mathematics Numerical Analysis and Symbolic Computation

1. Why hybrid symbolic-numeric methods?
2. What are hybrid symbolic-numeric methods?
3. Different approaches for hybrid symbolic-numeric methods
   - Z. Zeng's (H. J. Stetter's) approach
   - E. Kaltofen's approach
   - C. Yap's approach
   - Shortly about our approach

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

Why hybrid symbolic-numeric methods?

In general, in polynomial algebra one is interested in solving problems whose input are polynomials with complex coefficients, i.e. the coefficients are only imperfectly known (i.e floating point numbers).

Example: **Given** $p(x) = x^2 + 1.99x + 1.00, q(x) = x + 1.00$ and a tolerance $\delta = 0.01$, **compute** the greatest common divisor of p, q, i.e. $gcd(p, q)$! The tolerance $\delta = 0.01$ means that the third and subsequent decimals of the coefficients are unknown!

In particular, we[1] address a similar problem in polynomial algebra!

[1] Mădălina Hodorog, Bernard Mourrain, Josef Schicho

Why hybrid symbolic-numeric methods?

Example: **Given** $f(x) = x^2 - y^3 - 0.01 \in \mathbb{C}[x, y]$ squarefree, $\mathcal{C} = \{(x, y \in \mathbb{C}^2 | f(x, y) = 0)\}$ plane complex algebraic curve and a tolerance $\delta = 0.01$,
compute a set of δ-invariants of $\mathcal{C}$ (i.e. **genus**, etc) and its singularities (i.e. algebraic link, Alexander polynomial, etc).

We developed[1] several symbolic-numeric algorithms for computing all these δ-invariants. We presented them also in january at the Research Seminar in Rastenfeld.

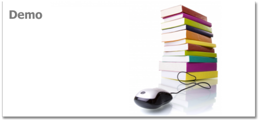
[1] Mădălina Hodorog, Bernard Mourrain, Josef Schicho

Why hybrid symbolic-numeric methods?

Example: **Given** $f(x) = x^2 - y^3 - 0.01 \in \mathbb{C}[x, y]$ squarefree, $\mathcal{C} = \{(x, y \in \mathbb{C}^2 | f(x, y) = 0)\}$ plane complex algebraic curve and a tolerance $\delta = 0.01$,
compute a set of δ-invariants of $\mathcal{C}$ (i.e. **genus**, etc) and its singularities (i.e. algebraic link, Alexander polynomial, etc).

We implemented[1] the algorithms in our library GENOM3CK using Axel.
Support: `http://people.ricam.oeaw.ac.at/m.hodorog/software.html`

M. Hodorog, B. Mourrain, J. Schicho. *GENOM3CK - A library for GENus cOMputation of plane Complex algebraiC Curves using Knot theory*. International Symposium on Symbolic and Algebraic Computation. Münich, Germany, 2010.



[1] Mădălina Hodorog, Bernard Mourrain, Josef Schicho

Why hybrid symbolic-numeric methods?

Example: **Given** $f(x) = x^2 - y^3 - 0.01 \in \mathbb{C}[x, y]$ squarefree, $\mathcal{C} = \{(x, y \in \mathbb{C}^2 | f(x, y) = 0)\}$ plane complex algebraic curve and a tolerance $\delta = 0.01$,
compute a set of δ-invariants of $\mathcal{C}$ (i.e. **genus**, etc) and its singularities (i.e. algebraic link, Alexander polynomial, etc).

Question: What does our algorithm really computes? What can we certify about the computed output? What do we mean by δ-invariants?

Why hybrid symbolic-numeric methods?

Example: **Given** $f(x) = x^2 - y^3 - 0.01 \in \mathbb{C}[x,y]$ squarefree, $\mathcal{C} = \{(x, y \in \mathbb{C}^2 | f(x,y) = 0)\}$ plane complex algebraic curve and a tolerance $\delta = 0.01$,
compute a set of δ-invariants of $\mathcal{C}$ (i.e. **genus**, etc) and its singularities (i.e. algebraic link, Alexander polynomial, etc).

Question: What does our algorithm really computes? What can we certify about the computed output? What do we mean by δ-invariants?
Answer: In order to provide our solution to these problems, we study different approaches for hybrid symbolic-numeric methods!

1. Why hybrid symbolic-numeric methods?

2. What are hybrid symbolic-numeric methods?

3. Different approaches for hybrid symbolic-numeric methods
   - Z. Zeng's (H. J. Stetter's) approach
   - E. Kaltofen's approach
   - C. Yap's approach
   - Shortly about our approach

What are hybrid symbolic-numeric methods?

- We use the name "hybrid symbolic-numeric methods" as in the book *"Computer Algebra Handbook"* (Editors: J. Grabmeier, E. Kaltofen, V. Weispfenning).
- The objects of study are polynomials with both:
  - ☛ exact coefficients, i.e. integer and rational numbers: $1, -2, \frac{1}{2}$.
  - ☛ and inexact coefficients, i.e. numerical values. For 1.001 we associate a tolerance of 10^{-3}, i.e. the last digit is uncertain.

What are hybrid symbolic-numeric methods?

- Numerical/approximate polynomial algebra

Basic Notions
symbolic-numeric methods
approximate polynomials
ill-posed problems

What are hybrid symbolic-numeric methods?

- Numerical/approximate polynomial algebra

Basic Notions
symbolic-numeric methods
approximate polynomials
ill-posed problems

Intuition. A symbolic-numeric method is similar to what Knuth calls a *seminumerical* algorithm, one that lies ”*on the borderline between numeric and symbolic computation.*”

What are hybrid symbolic-numeric methods?

- Numerical/approximate polynomial algebra

Basic Notions

symbolic-numeric methods

approximate polynomials

ill-posed problems

Remark. Consider $\mathbb{R}[x](\mathbb{C}[x])$ with the metric given by the euclidean norm $||\cdot||$. Given $f \in \mathbb{R}[x]$ and $\delta \in \mathbb{R}_+$, define an δ-neighborhood of f as:

$$N_{f,\delta} = \{g \in \mathbb{R}[x] : ||f - g|| \leq \delta\}.$$

What are hybrid symbolic-numeric methods?

- Numerical/approximate polynomial algebra

Basic Notions
symbolic-numeric methods
approximate polynomials
ill-posed problems

Definition. An ill-posed problem is a problem which does not fulfill Hadamard's definition of well-posedness:

- For all data, a solution exists.
- For all data, the solution is unique.
- The solution depends continously on the data $(*)$.

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

What are hybrid symbolic-numeric methods?

- Numerical/approximate polynomial algebra

Basic Notions
symbolic-numeric methods
approximate polynomials
ill-posed problems

Example 1 (ill-posed problem which does not fulfill $(*)$).
For $f(x) = x^4 - 1, g(x) = x^2 + x - 2$, get $gcd(f, g) = x - 1$.

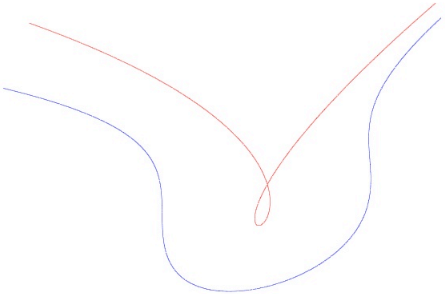
For $\tilde{f}(x) = x^4 - 1 - 0.0001, \tilde{g}(x) = x^2 + x - 2 - 0.0001$, get $gcd(\tilde{f}, \tilde{g}) = 1$.

What are hybrid symbolic-numeric methods?

- Numerical/approximate polynomial algebra

Basic Notions
symbolic-numeric methods
approximate polynomials
ill-posed problems

Example 2 (ill-posed problem which does not fulfill $(*)$).
Let $s = (0,0)$ singularity of $\mathcal{C} = \{(x,y) \in \mathbb{R}^2 | -x^3 - xy + y^2 = 0\}$, and
$\mathcal{D} = \{(x,y) \in \mathbb{R}^2 | -x^3 - xy + y^2 - 0.01 = 0\}$!



Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

Z. Zeng's approach

We distinguish between the theoretical part and the practical part.
Theoretical part. Principles.

Z. Zeng's approach

We distinguish between the theoretical part and the practical part.
Theoretical part. Principles.

- the set of problem instances with certain solution structure forms a pejorative manifold.
  - For $P : D \rightarrow S$, we partition the set of input data D in pejorative manifolds M_i depending on the structure of the solution. M_i form a stratification structure for D!
  - Tiny arbitrary perturbation pushes a problem instance away from its residing manifold, losing the structure of the solution.

Z. Zeng's approach

Theoretical part. Example.

- Let $E : \mathbb{C}[x] \times \mathbb{C}[x] \to \mathbb{C}[x]$ an exact algorithm which Given (f, g) assigns $E(f, g) := gcd(f, g)$!
  This problem is ill-posed!
  The stratification structure of $\mathbb{C}[x] \times \mathbb{C}[x]$ consists of:

$$\begin{cases} \mathcal{P}_k^{m,n} = \{(f,g) \in \mathbb{C}^2[x] : \mathtt{deg}(f) = m \geq \mathtt{deg}(g) = n, \mathtt{deg}(gcd(f,g)) = k\} \\ \\ \mathtt{codim}\ \mathcal{P}_k^{m,n} = k \in \mathbb{Z}_+ \\ \\ \overline{\mathcal{P}_n^{m,n}} \subset \overline{\mathcal{P}_{n-1}^{m,n}} \subset \ldots \subset \overline{\mathcal{P}_0^{m,n}} \end{cases}$$

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

Z. Zeng's approach

Theoretical part. Example.

- Let $E : \mathbb{C}[x] \times \mathbb{C}[x] \to \mathbb{C}[x]$ an exact algorithm which
  Given (f, g) assigns $E(f, g) := gcd(f, g)$!
  This problem is ill-posed!
  The stratification structure of $\mathbb{C}[x] \times \mathbb{C}[x]$ consists of:

$$\begin{cases} \mathcal{P}_k^{m,n} = \{(f,g) \in \mathbb{C}^2[x] : \mathtt{deg}(f) = m \geq \mathtt{deg}(g) = n, \mathtt{deg}(gcd(f,g)) = k\} \\ \mathtt{codim}\ \mathcal{P}_k^{m,n} = k \in \mathbb{Z}_+ \\ \overline{\mathcal{P}_n^{m,n}} \subset \overline{\mathcal{P}_{n-1}^{m,n}} \subset \ldots \subset \overline{\mathcal{P}_0^{m,n}} \end{cases}$$

- $(f, g) \in \mathcal{P}_k^{m,n} : f = uv, g = uw, gcd(f, g) = u, deg(u) = k, lc(u) = 1.$
  Each $\mathcal{P}_k^{m,n}$ is parametrized as $F(\mathbf{u}, \mathbf{v}, \mathbf{w}) = \mathbf{z}$, where $\mathbf{u}, \mathbf{v}, \mathbf{w}$, respectively $\mathbf{z}$ coefficients vectors of u, v, w, respectively f, g.

Z. Zeng's approach

Theoretical part. Example.

- $E : \mathbb{C}^2[x] \to \mathbb{C}[x]$ exact algorithm s.t. $(f, g) \mapsto E(f, g) := gcd(f, g)$.
- $f(x) = x^4 - 1, g(x) = x^2 + x - 2, gcd(f, g) = x - 1$
- $\tilde{f}(x) = x^4 - 1.0001, \tilde{g}(x) = x^2 + x - 2.0001, \delta = 10^{-4}, gcd(\tilde{f}, \tilde{g}) = 1.$

Z. Zeng's approach

Theoretical part. Example.

- $E : \mathbb{C}^2[x] \to \mathbb{C}[x]$ exact algorithm s.t. $(f, g) \mapsto E(f, g) := gcd(f, g)$.
- $f(x) = x^4 - 1, g(x) = x^2 + x - 2, gcd(f, g) = x - 1$
- $\tilde{f}(x) = x^4 - 1.0001, \tilde{g}(x) = x^2 + x - 2.0001, \delta = 10^{-4}, gcd(\tilde{f}, \tilde{g}) = 1.$
- $\mathbb{C}^2[x]$ has the stratification structure: $\overline{\mathcal{P}_2^{4,2}} \subset \overline{\mathcal{P}_1^{4,2}} \subset \overline{\mathcal{P}_0^{4,2}}$, where

  $\mathcal{P}_i^{4,2} = \{(f, g) : \mathtt{deg}(f) = 4, \mathtt{deg}(g) = 2, \mathtt{deg}(gcd(f, g)) = i\}, i = 0, 1, 2.$

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

Z. Zeng's approach

Theoretical part. Example.

- $E : \mathbb{C}^2[x] \to \mathbb{C}[x]$ exact algorithm s.t. $(f, g) \mapsto E(f, g) := gcd(f, g)$.
- $f(x) = x^4 - 1, g(x) = x^2 + x - 2, gcd(f, g) = x - 1$
- $\tilde{f}(x) = x^4 - 1.0001, \tilde{g}(x) = x^2 + x - 2.0001, \delta = 10^{-4}, gcd(\tilde{f}, \tilde{g}) = 1.$
- $\mathbb{C}^2[x]$ has the stratification structure: $\overline{\mathcal{P}_2^{4,2}} \subset \overline{\mathcal{P}_1^{4,2}} \subset \overline{\mathcal{P}_0^{4,2}}$, where

  $$\mathcal{P}_i^{4,2} = \{(f, g) : \mathtt{deg}(f) = 4, \mathtt{deg}(g) = 2, \mathtt{deg}(gcd(f, g)) = i\}, i = 0, 1, 2.$$
- $gcd(f, g) = x - 1 \in \mathcal{P}_1^{4,2}$, $gcd(\tilde{f}, \tilde{g}) = 1 \in \mathcal{P}_0^{4,2}$.

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

Z. Zeng's approach

Theoretical part. Example.

- $E : \mathbb{C}^2[x] \to \mathbb{C}[x]$ exact algorithm s.t. $(f, g) \mapsto E(f, g) := gcd(f, g)$.
- $f(x) = x^4 - 1, g(x) = x^2 + x - 2, gcd(f, g) = x - 1$
- $\tilde{f}(x) = x^4 - 1.0001, \tilde{g}(x) = x^2 + x - 2.0001, \delta = 10^{-4}, gcd(\tilde{f}, \tilde{g}) = 1.$
- $\mathbb{C}^2[x]$ has the stratification structure: $\overline{\mathcal{P}_2^{4,2}} \subset \overline{\mathcal{P}_1^{4,2}} \subset \overline{\mathcal{P}_0^{4,2}}$, where

  $$\mathcal{P}_i^{4,2} = \{(f, g) : \mathsf{deg}(f) = 4, \mathsf{deg}(g) = 2, \mathsf{deg}(gcd(f, g)) = i\}, i = 0, 1, 2.$$
- $gcd(f, g) = x - 1 \in \mathcal{P}_1^{4,2}$, $gcd(\tilde{f}, \tilde{g}) = 1 \in \mathcal{P}_0^{4,2}$.
- Problem: Given $\tilde{f}, \tilde{g}, \delta$ and not knowing f, g, identify $\mathcal{P}_1^{4,2}$!

Z. Zeng's approach

Theoretical part. Example.

- $E : \mathbb{C}^2[x] \to \mathbb{C}[x]$ exact algorithm s.t. $(f, g) \mapsto E(f, g) := gcd(f, g)$.
- $f(x) = x^4 - 1, g(x) = x^2 + x - 2, gcd(f, g) = x - 1$
- $\tilde{f}(x) = x^4 - 1.0001, \tilde{g}(x) = x^2 + x - 2.0001, \delta = 10^{-4}, gcd(\tilde{f}, \tilde{g}) = 1.$
- $\mathbb{C}^2[x]$ has the stratification structure: $\overline{\mathcal{P}_2^{4,2}} \subset \overline{\mathcal{P}_1^{4,2}} \subset \overline{\mathcal{P}_0^{4,2}}$, where

  $\mathcal{P}_i^{4,2} = \{(f, g) : \mathtt{deg}(f) = 4, \mathtt{deg}(g) = 2, \mathtt{deg}(gcd(f, g)) = i\}, i = 0, 1, 2.$
- $gcd(f, g) = x - 1 \in \mathcal{P}_1^{4,2}$, $gcd(\tilde{f}, \tilde{g}) = 1 \in \mathcal{P}_0^{4,2}$.
- Problem: Given $\tilde{f}, \tilde{g}, \delta$ and not knowing f, g, identify $\mathcal{P}_1^{4,2}$!
- Answer (Z. Zeng): $\mathcal{P}_1^{4,2}$ is the highest codimension manifold among all manifolds that intersect the δ-neighborhood of $\tilde{f}, \tilde{g}$!

Z. Zeng's approach

Theoretical part. Principles.

Consider $\tilde{P}$ a perturbation from an exact problem P with sufficiently small error δ. Formulate an approximate solution to $\tilde{P}$ using the 3-strikes principle:

- The approximate solution of $\tilde{P}$ is the exact solution of a problem $\hat{P}$ within δ.
- $\hat{P}$ is on the highest codimension manifold Π intersecting the $N_{\tilde{P},\delta}$.
- $\hat{P}$ is the nearest problem to $\tilde{P}$ on Π.

Remark: This approximate solution satisfy the property: as the error δ approaches 0 the approximate solution converges to the exact solution.

Z. Zeng's approach

Practical part. A 2-stage approach may help to solve an ill-posed problem:

- Stage 1: maximizing the codimension of the manifolds.
  (i.e. determine the structure of the solution).
  Computation tools: matrix building.
- Stage 2: minimizing the distance to the manifold.
  Computation tools: nonlinear least-squares, Gauss-Newton iteration.

Z. Zeng's approach

Practical part. Example.

Given $(\tilde{f}, \tilde{g}, \delta) \in \mathbb{C}^2[x] \times \mathbb{R}_+$, $m = deg(f) \geq n = deg(g)$ compute the $gcd(\tilde{f}, \tilde{g})$!

- Stage 1: Compute the degree of $gcd(\tilde{f}, \tilde{g})$!
- Stage 2: Compute the coefficients of $gcd(\tilde{f}, \tilde{g})$!

Z. Zeng's approach

Practical part. Example.

Stage 1. Compute the degree of $gcd(\tilde{f}, \tilde{g})$! by using a low rank approximation of the Sylvester matrix of $(\tilde{f}, \tilde{g})$, i.e. $S := S(\tilde{f}, \tilde{g})$!

- **Theorem** 1: $deg(gcd(\tilde{f}, \tilde{g})) = m + n - rank(S(\tilde{f}, \tilde{g}))$.

Z. Zeng's approach

Practical part. Example.

Stage 1. Compute the degree of $gcd(\tilde{f}, \tilde{g})$! by using a low rank approximation of the Sylvester matrix of $(\tilde{f}, \tilde{g})$, i.e. $S := S(\tilde{f}, \tilde{g})$!

- **Theorem** 1**:** $deg(gcd(\tilde{f}, \tilde{g})) = m + n - rank(S(\tilde{f}, \tilde{g}))$.
- How to compute $rank(S(\tilde{f}, \tilde{g}))$ in the presence of data perturbations?

Z. Zeng's approach

Practical part. Example.

Stage 1. Compute the degree of $gcd(\tilde{f}, \tilde{g})$! by using a low rank approximation of the Sylvester matrix of $(\tilde{f}, \tilde{g})$, i.e. $S := S(\tilde{f}, \tilde{g})$!

- **Theorem** 1**:** $deg(gcd(\tilde{f}, \tilde{g})) = m + n - rank(S(\tilde{f}, \tilde{g}))$.
- How to compute $rank(S(\tilde{f}, \tilde{g}))$ in the presence of data perturbations?
- Intuition: If S is rank deficient, then small perturbations of the matrix values can yield a matrix of full rank! We approximate S by a low rank matrix $\tilde{S}$, by Singular Value Decomposition (SVD) of S.

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

Z. Zeng's approach

Practical part. Example.

Stage 1. Compute the degree of $gcd(\tilde{f}, \tilde{g})$! by using a low rank approximation of the Sylvester matrix of $(\tilde{f}, \tilde{g})$, i.e. $S := S(\tilde{f}, \tilde{g})$!

- **Theorem** 1: $deg(gcd(\tilde{f}, \tilde{g})) = m + n - rank(S(\tilde{f}, \tilde{g}))$.
- How to compute $rank(S(\tilde{f}, \tilde{g}))$ in the presence of data perturbations?
- Intuition: If S is rank deficient, then small perturbations of the matrix values can yield a matrix of full rank! We approximate S by a low rank matrix $\tilde{S}$, by Singular Value Decomposition (SVD) of S.
- **Theorem** 2: Let $A \in \mathcal{M}_{m\times n}(\mathbb{C})$ with $m \geq n$. Then there exists $U \in \mathcal{M}_{m\times m}, V \in \mathcal{M}_{n\times n}$ orthogonal, and a unique $\Sigma(A) \in \mathcal{M}_{m\times n}$ with $\Sigma(A) = diag(\sigma_1 \geq ... \geq \sigma_r)$ s.t. $A = U\Sigma V^t$.
- **Remark:** $rank(A) = r$ and $r \leq min(m, n) = n$.

Z. Zeng's approach

Practical part. Example.

Stage 1. Compute the degree of $gcd(\tilde{f}, \tilde{g})$!

- **Theorem** 3**:** Let $S \in \mathcal{M}_{m \times n}$ with $\Sigma(S) = diag(\sigma_1, \sigma_2, ..., \sigma_n)$ as in Theorem 2. Assume $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_k > \theta \geq \sigma_{k+1} \geq ... \geq \sigma_n$ for $\theta \in \mathbb{R}_+$. Then there exists $\tilde{S}$, with $\Sigma(\tilde{S}) = diag(\sigma_1, \sigma_2, ..., \sigma_k)$, i.e. $rank(\tilde{S}) = k$ and

$$\min_{rank(B)=k} ||S - B|| = ||S - \tilde{S}|| \leq \theta.$$

- **Remark:** By dropping insignificant singular values of S (i.e. all $\sigma_i \leq \theta$) we obtain $\tilde{S}$ with $rank(\tilde{S}) < rank(S)$ and $||S - \tilde{S}|| \leq \theta$!

Z. Zeng's approach

Practical part. Example.

Stage 2. Compute the coefficients of $gcd(\tilde{f}, \tilde{g})$ with $deg = k$!

- Find $(\hat{u}, \hat{v}, \hat{w})$ with $gcd(\tilde{f}, \tilde{g}) \cong \hat{u}, gcd(\hat{v}, \hat{w}) = 1, deg(\hat{u}) = k$ and

$$\left\{ \begin{array}{l} \hat{u}\hat{v} \cong \tilde{f} \\ \hat{u}\hat{w} \cong \tilde{g} \end{array} \right. . \tag{2}$$

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

Z. Zeng's approach

Practical part. Example.

Stage 2. Compute the coefficients of $gcd(\tilde{f}, \tilde{g})$ with $deg = k$!

- Find $(\hat{u}, \hat{v}, \hat{w})$ with $gcd(\tilde{f}, \tilde{g}) \cong \hat{u}, gcd(\hat{v}, \hat{w}) = 1, deg(\hat{u}) = k$ and

$$\begin{cases} \hat{u}\hat{v} \cong \tilde{f} \\ \hat{u}\hat{w} \cong \tilde{g} \end{cases} . \tag{2}$$

- Rewrite (2) as $F(\mathbf{u}, \mathbf{v}, \mathbf{w}) \cong \mathbf{b}$, where **u,v,w**, respectively **b** represents the coefficients vectors of the polynomials $\hat{u}, \hat{v}, \hat{w}$, respectively $\tilde{f}, \tilde{g}$.

Z. Zeng's approach

Practical part. Example.

Stage 2. Compute the coefficients of $gcd(\tilde{f}, \tilde{g})$ with $deg = k$!

- Find $(\hat{u}, \hat{v}, \hat{w})$ with $gcd(\tilde{f}, \tilde{g}) \cong \hat{u}, gcd(\hat{v}, \hat{w}) = 1, deg(\hat{u}) = k$ and

$$\begin{cases} \hat{u}\hat{v} \cong \tilde{f} \\ \hat{u}\hat{w} \cong \tilde{g} \end{cases} . \tag{2}$$

- Rewrite (2) as $F(\mathbf{u}, \mathbf{v}, \mathbf{w}) \cong \mathbf{b}$, where **u,v,w**, respectively **b** represents the coefficients vectors of the polynomials $\hat{u}, \hat{v}, \hat{w}$, respectively $\tilde{f}, \tilde{g}$.
- Solve overdeterminate system $F(\mathbf{u}, \mathbf{v}, \mathbf{w}) \cong \mathbf{b}$. Solve $\underset{(u,v,w) \in \mathcal{P}_k^{m,n}}{min} ||F(u, v, w) - b|| = ||F(\hat{u}, \hat{v}, \hat{w}) - b||$ by Gauss-Newton.
- **Necessary:** the coefficients of the gcd must be real numbers! When the coefficients are integers, Stage 2 cannot be used (our case)!

1. Why hybrid symbolic-numeric methods?

2. What are hybrid symbolic-numeric methods?

3. Different approaches for hybrid symbolic-numeric methods
   - Z. Zeng's (H. J. Stetter's) approach
   - E. Kaltofen's approach
   - C. Yap's approach
   - Shortly about our approach

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

E. Kaltofen's approach

PROBLEM

- Given: $f, g \in \mathbb{C}[x], \texttt{deg}(f) = m, \texttt{deg}(g) = n, k \leq \texttt{min}(m, n) \in \mathbb{Z}_+$

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

E. Kaltofen's approach

PROBLEM

- Given: $f, g \in \mathbb{C}[x], \texttt{deg}(f) = m, \texttt{deg}(g) = n, k \leq \texttt{min}(m, n) \in \mathbb{Z}_+$
- find: $\tilde{f}, \tilde{g} \in \mathbb{C}[x]$
- s.t. $\underset{deg(gcd(\tilde{f},\tilde{g}))\geq k}{min} ||\tilde{f} - f||^2 + ||\tilde{g} - g||^2 := \mathcal{N}$, and $\texttt{deg}(||\tilde{f} - f||) \leq m, \texttt{deg}(||\tilde{g} - g|| \leq n)$.

E. Kaltofen's approach

METHOD

- Prove the existence of $\mathcal{N}$.
- Compute $\tilde{f}, \tilde{g}, \mathcal{N}$ by an iterative algorithm denoted `IterativeAlgo` given $f, g, k, tol \in \mathbb{R}_+$, based on:

Theorem 4

Given $f(x), g(x) \in \mathbb{C}[x]$ with $\deg(f) = m, \deg(g) = n$. Let $S(f,g)$ the Sylvester matrix of f, g and S_k the k-th Sylvester matrix, $1 \leq k \leq min(m,n)$. Then:

$$\deg(gcd(f,g)) \geq k \Leftrightarrow \text{rank}(S) \leq m+n-k \Leftrightarrow \dim Ker(S_k) \geq 1$$

Let $S_k = [b_k A_k]$, b_k is the first column of S_k, A_k the remaining columns. Then $\dim Ker(S_k) \geq 1 \Leftrightarrow A_k x = b_k$ has a solution.

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

E. Kaltofen's approach

APPLICATION

Algorithm 1 Approximate gcd of univariate polynomials

Input: $f, g \in \mathbb{C}[x], \mathtt{deg}(f) = m \geq n = \mathtt{deg}(g), \epsilon \in \mathbb{R}_+$
Output: $\epsilon\text{-}gcd(f, g)$

- Initialize $k = n$
- Repeat
  - Compute $\tilde{f}, \tilde{g}, \mathcal{N}$ with $\mathtt{IterativeAlgo}(f, g, k, tol)$.
  - $k - -$
- until $\mathcal{N} < \epsilon$ or $k < 0$
- If $k \geq 0$ then compute $\epsilon\text{-}gcd$ from the matrix $S_k(\tilde{f}, \tilde{g})$, for instance with an algorithm like Zeng's algorithm based on SVD.

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

C. Yap's approach (exact geometric computation)

Constructive zero bounds

- an expression E is a syntactic object constructed from a set of operators Ω over $\mathbb{R}$.
- evaluating predicates amounts to determining the sign of E.

Definition

$b > 0$ is a zero bound for E if the following holds: if E is well-defined ($E \neq \uparrow$) and $E \neq 0$ then $|E| \geq b$. $-log_2(b)$ is a zero bit-bound for E.

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

C. Yap's approach (exact geometric computation)

Constructive zero bounds

- an expression E is a syntactic object constructed from a set of operators Ω over $\mathbb{R}$.
- evaluating predicates amounts to determining the sign of E.

Definition

$b > 0$ is a zero bound for E if the following holds: if E is well-defined ($E \neq \uparrow$) and $E \neq 0$ then $|E| \geq b$. $-log_2(b)$ is a zero bit-bound for E.

- Given E determine $sign(E) = \{\uparrow, +1, -1, 0\}$!
  - If $E \neq \uparrow$ then compute $\tilde{E}$ s.t. $|\tilde{E} - E| < \frac{b}{2}$.
  - If $E = \uparrow$ then $\tilde{E} = \uparrow$.
  - If $|\tilde{E}| \geq \frac{b}{2}$ then $sign(E) = sign(\tilde{E})$ else $E = 0$.

Doctoral Program Computational Mathematics Numerical Analysis and Symbolic Computation

C. Yap's approach (exact geometric computation)

Approximate expression evaluation

- Given E and a precision $p \in \mathbb{R}_+$ compute an approximation of E within precision p!
- all E are "programs", rooted, labeled directed acyclic graphs (DAG).
- use precision-driven approach.
  - propagate precision values down to the leaves.
  - approximate the value at the leaf to any desired precision.
  - propate the approximations up to the root.

Numerical filters

- Numerical filters are an effective technique for speeding up predicate evaluation.

1. Why hybrid symbolic-numeric methods?

2. What are hybrid symbolic-numeric methods?

3. Different approaches for hybrid symbolic-numeric methods
   - Z. Zeng's (H. J. Stetter's) approach
   - E. Kaltofen's approach
   - C. Yap's approach
   - Shortly about our approach

Shortly about our approach

- Let I the set of coefficient vectors of polynomials of fixed degree, O a discrete set.
- Let $E : I \to O$ the symbolic algorithm s.t.

Given $f \in I$ assigns $E(f)$ the invariants of the curve defined by f.
This problem is ill-posed!

Shortly about our approach

- Let I the set of coefficient vectors of polynomials of fixed degree, O a discrete set.
- Let $E : I \to O$ the symbolic algorithm s.t.

Given $f \in I$ assigns $E(f)$ the invariants of the curve defined by f.
This problem is ill-posed!

- Let $A : I \times \mathbb{R}_+ \to O$ the symbolic-numeric algorithm we designed s.t.

Given $(f, \epsilon) \in I \times \mathbb{R}_+$ assigns the δ-invariants of the curve defined by f.

- For $f \in I$ a perturbation of f is a function
$f_- : \mathbb{R}_+ \to I, \delta \mapsto f_\delta$ such that $|f - f_\delta| \leq \delta$ for all $\delta \in \mathbb{R}_+$.
We call f the exact data, f_δ the perturbed data, δ the noise level (error, tolerance).

In our problem, we are given f_δ and δ but not f!

Shortly about our approach

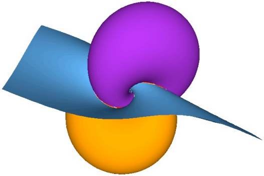
We can prove the following properties of A:

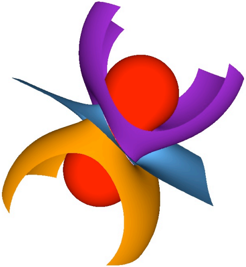
- $A_{\_}$ depends continuously on f_δ **continuity** (1).
- $\exists \alpha : \mathbb{R}_+ \to \mathbb{R}_+$ continuous, monotonic, $\lim_{\delta\to 0} \alpha(\delta) = 0$ s.t. for any f_δ

$$\lim_{\delta\to 0} A(f_\delta, \alpha(\delta)) = E(f), \text{i.e. } \textbf{convergence for perturbed data } (2).$$

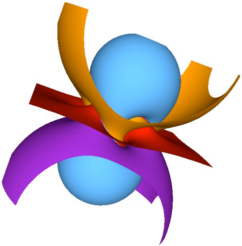
- In this case α is called the "parameter choice rule"!
  The algorithm A_ϵ is called a regularization.

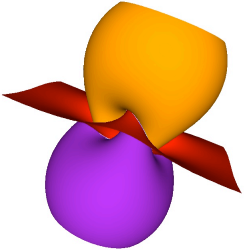
Instead of looking for the exact solution, we look for approximations with (1), (2).





Thank you for your attention.





Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation