A New Algorithm for Computing Certified Numerical Approximations of the Roots of a Zero-dimensional System

Stef Graillat, Philippe Trébuchet

LIP6 - Université Pierre et Marie Curie (Paris 6)

ISSAC 2009
International Symposium on Symbolic and Algebraic Computation
Seoul, Korea, July 28-31, 2009

Motivations

Finding the common solutions to a polynomial system

$$\begin{cases} f_1(x_1,\ldots,x_n) = 0, \\ f_2(x_1,\ldots,x_n) = 0, \\ \vdots \\ f_s(x_1,\ldots,x_n) = 0, \end{cases}$$

with $f_i \in \mathbb{C}[x_1,\ldots,x_n]$ or in algebraic terms, finding the variety V of the ideal $I = \langle f_1,\ldots,f_s\rangle$

We assume that V is finite (I is 0-dimensional)

Outline of the talk

1. The univariate case
2. Brief review of the different approaches
3. The multivariate case
4. Implementation and numerical experiments

The univariate case

Let $f(x) = f_d x^d + \cdots + f_1 x + f_0$ and define $A = \mathbb{C}[x]/\langle f\rangle$
The matrix of the multiplication operator

$$\begin{aligned} M_x : A &\rightarrow A \\ a &\mapsto ax \end{aligned}$$

in the basis $(1, x\ldots, x^{d-1})$ is

$$M_x = \begin{pmatrix} 0 & \cdots & 0 & -f_0/f_d \\ 1 & \ddots & \vdots & \vdots \\ & \ddots & 0 & \vdots \\ 0 & & 1 & -f_{d-1}/f_d \end{pmatrix}$$

The eigenvalues of M_x^T are the roots $\zeta_1, \ldots, \zeta_d$ of f

The univariate case (cont'd)

The eigenvectors of

$$M_x^T = \begin{pmatrix} 0 & \cdots & 0 & -f_0/f_d \\ 1 & \ddots & \vdots & \vdots \\ & \ddots & 0 & \vdots \\ 0 & & 1 & -f_{d-1}/f_d \end{pmatrix}^T$$

associated to the eigenvalue ζ_i is $(1, \zeta_i, \ldots, \zeta_i^{d-1})$

The geometric multiplicity of eigenvalue ζ_i is always one for all i

If one works in the dual of A, the vector $(1, \zeta_i, \ldots, \zeta_i^{d-1})$ can be considered as the evaluation operation at the zero ζ_i since

$$(1, \zeta_i, \ldots, \zeta_i^{d-1}) \cdot (a_0, a_1, \ldots, a_{d-1})^T = \sum_{i=0}^{d-1} a_i \zeta^i$$

Algebraic Methods

Equations → Representation of the quotient algebra → Computations inside the quotient algebra

Our Contribution

-Cyclic
-Robot
-...

-Gröbner bases
-Gen. Norm. Forms
-Resultants

Different methods

1. Rational Univariate Representation (RUR) [Rouillier] : a symbolic representation of the roots
2. Homotopic continuation method [Verschelde, Sommese]
3. Eigenvalue computation [Corless, Gianni, Trager,...] : simultaneous Schur decomposition of the multiplication matrices
4. Eigenvalue/Eigenvector computation [Moller, Stetter] : use either eigenvalue or eigenvector to recover informations on the roots

   $\rightsquigarrow$ Contributions :

   - use of the structure of the eigenvector to speed up the algorithm
   - use of certified numerical algorithms instead of symbolic ones

Multivariate case

Let $f_1, \ldots, f_s \in \mathbb{C}[x_1, \ldots, x_n]$ and define $A = \mathbb{C}[x]/\langle f_1, \ldots, f_s \rangle$

Let B a monomial basis of A

The matrix of the multiplication operator M_i is defined by

$$\begin{array}{rcl} M_i : A & \to & A \\ a & \mapsto & x_i a \end{array}$$

Theorem 1 (Corless/Gianni/Trager 96)

The matrices $M_1, \ldots, M_n$ commute. So there exists a unitary matrix V such that $V^* M_i V = U_i$ is upper triangular for all i (Schur decomposition). The zeros are

$$\zeta_j = [u^1_{jj}, u^2_{jj}, \ldots, u^s_{jj}].$$

Multivariate case (cont'd)

Theorem 2 (Stieckelberger)

The common eigenvectors to all the transposed multiplication operators, are the evaluation at the root :

$$\begin{array}{rcl} 1_{\zeta_j} : A & \rightarrow & \mathbb{C} \\ p & \mapsto & p(\zeta_j) \end{array}$$

One can restrict to the multiplication by one variable.

Algorithm 1 (`Undernum`, Moller & Tenberg)

INPUT : *$lm = (M_1, \ldots, M_n)$ the dual multiplication operators*
i an integer index
lv a list a vectors expressed on the canonical basis
OUTPUT : *A numerical approximation of a common eigenvector to all the M_i.*

- *$Sol = []$*
- *Compute M, the matrix of the restriction of M_i on the vector space spanned by the vectors of lv.*
- *For each eigenvalue v of M do*
  - *if the eigenspace associated to v has dimension 1 then*
    - *Let e be the eigenvector of the eigenspace.*
    - *Let M_{lv} be the matrix whose columns are the vectors of lv.*
    - *$e' = M_{lv}e$*
    - *$Sol = Sol \cup [e']$*
  - *else*
    - *Let le denote the list of eigenvectors associated to v.*
    - *$Sol = Sol \cup \texttt{Undernum}(lm, i+1, le)$*
- *Return Sol*

Algorithm 2 (Symbonum, Moller & Tenberg)

INPUT : $M_1, \ldots, M_n$ *the n transposed multiplication operators*
OUTPUT : *A numerical approximation of the roots of the system* $f_1, \ldots, f_s$

- $Res = []$
- *Let C be the list of the vectors of the canonical basis.*
- $Tmp\_sol = \texttt{Undernum}([M_1, \ldots, M_n], 1, C)$
- *For each v in Tmp_sol do*
  - *tmpres=[]*
  - *for i from 1 to n do*
    - $tmpres = tmpres \cup DotProd(Row(1, M_i), v/v[1])$
  - $Res = Res \cup tmpres$
- *return Res*

A numerical algorithm

Aim : having a numerical version of this algorithm

Problems & drawbacks :

- no certification on what is computed
- numerically speaking, the eigenvectors are not well defined
- the algorithm requires the computation of the multiplication operators by all variables

Example

System : $X^2 = 0, Y^2 = 0$

Monomial basis : $B = \{1, x, y, xy\}$

$$M_x = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \rightsquigarrow M_x^T = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\text{eigenvectors : } \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

Need to examine the action of M_y^T on this 2-dimensional space !

Numerical observations

→ certified error bounds for eigenvalues and eigenvectors [Rump]
Use of interval arithmetic and self-validating methods

→ works only with nonderogatory eigenvalue but

Theorem 3 (Rump)

For $A \in M_n(\mathbb{C})$, let an eigenvalue $\lambda \in \mathrm{Spec}(A)$ (Spec denotes the spectrum) be given with algebraic multiplicity m and let $y \neq 0$, be a vector of $\mathbb{C}^n$ such that $Ay = \lambda y$, i.e. y is an eigenvector associated to λ. Then for all $\epsilon > 0$ there exists $\tilde{A}$ such that $\|A - \tilde{A}\|_\infty \leq \epsilon$ and the following properties hold :

- $\lambda \in \mathrm{Spec}(\tilde{A})$.
- *λ is of algebraic multiplicity m.*
- *λ is of geometric multiplicity one.*
- $\tilde{A}y = \lambda y$.

Numerical observations (cont'd)

- certified error bounds for eigenvalues and eigenvectors [Rump]

  → compute in general an enclosure of a basis of a full invariant subspace and not an enclosure for only eigenvectors

  → but eigenvectors belongs to the full invariant subspace

  → the use of the structure of the eigenvectors makes it possible to recover them from the full invariant subspace

  → use of fast QR-multishift routine (LAPACK) to compute eigenelements

  → certification can be done once at the end of the algorithm

Clusterization

Need to group together eigenvalues that are very closed

Consider an eigenvalue α, and v and u be its associated left eigenvector and right eigenvector : $Au = \alpha u$ and $v^T A = \alpha v^T$.

The reciprocal condition number of α is

$$\mathrm{rcond}_\alpha = \frac{|v^* u|}{\|v\| \cdot \|u\|},$$

α_i and α_j are grouped together if

$$|\alpha_i - \alpha_j| \leq \max\left(\frac{\mathrm{prec}}{\mathrm{rcond}_{\alpha_i}}, \frac{\mathrm{prec}}{\mathrm{rcond}_{\alpha_j}}\right)$$

Structure of the eigenvectors

Theorem 4

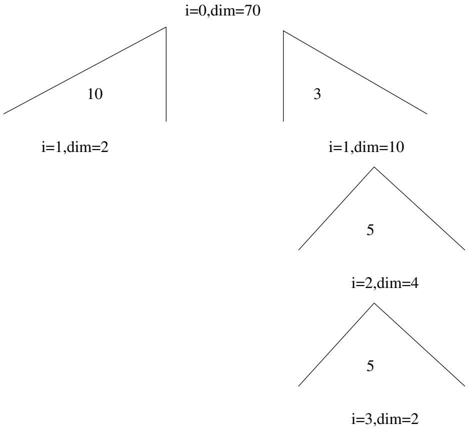
Let $\alpha_1, \dots, \alpha_k$ *be eigenvalues of respectively* $M_{x_1}^T, \dots, M_{x_k}^T$*. Consider the vector space*

$$E = \mathrm{Eig}(M_{x_1}^T, \alpha_1) \cap \mathrm{Eig}(M_{x_2}^T, \alpha_2) \cap \cdots \cap \mathrm{Eig}(M_{x_k}^T, \alpha_k)$$

where $\mathrm{Eig}(M_{x_i}^T, \alpha_i)$ *denotes the full invariant subspace of the matrix* $M_{x_i}^T$ *associated to the eigenvalue* α_i*. Let* m *be a monomial of the monomial basis* B *such that* $m = x_1^{d_1} \cdots x_k^{d_k}$ *Then the common eigenvectors to all the transposed multiplication operators that belong to* E *are such that : the coordinate of* m *in these vectors is* $\alpha_1^{d_1} \cdots \alpha_k^{d_k}$*.*

→ this gives contraints on the eigenvectors

Example on cyclic5



Implementation

Implementation in C++ (2500 lines)

The implementation is divided into three main components :

- the routine to compute the normal form of the quotient algebra
- the routine for performing the numerical root computing
- the routine to certify the clusters of the first matrix chosen

Use of a generic BLAS and LAPACK library with GMP and Boost (for interval)

Timings

Laptop Intel Core 2 Duo 8400 with 4 Go running on Linux 2.6.26

Name	arith	Nb. sol.	Time (s)	Prec
cyclic5	double	70	2	1e-15
cyclic5	long double	70	3	1e-16
cyclic5	mpf_class	70	103.84	<1e-50
katsura6	double	64	0.3	1e-10
katsura6	long double	64	0.91	1e-16
katsura6	mpf_class	64	33.91	1e-40
katsura7	double	128	3.8	1e-10
katsura7	long double	128	7.3	1e-16
katsura7	mpf_class	128	515	1e-43
katsura8	double	256	96	1e-4
katsura8	long double	256	127	1e-10
katsura8	mpf_class	256	$> 1h$	1e-10
fabrice24	double	40	0.07	1e-8
fabrice24	long double	40	0.14	1e-11
fabrice24	mpf_class	40	9	1e-41

double = 64 bits, long double = 80 bits, mpf = 200 bits

Conclusion and future work

Algorithm with two steps :

- a first numerical computation that is currently not certified
- a second step that is a verification of the numerical computations

The main improvements of this algorithm are

- the use of certified numerical computation
- the use of duality
- the use of the structure of the evaluation operators to avoid some recursive calls

Future work :

- parallelization of recursive calls (OpenMP)
- algebraic multiplicity of the clusters
- better use of the structure of the representation of the quotient algebra

Thank you for your attention