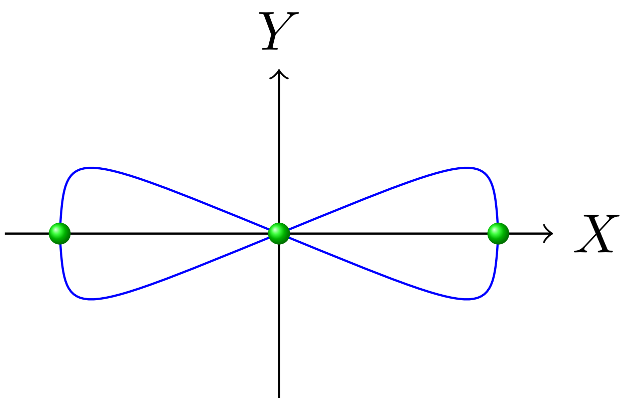
On the complexity of solving bivariate systems

Esmaeil Mehrabi   Éric Schost
CSD, Western University
joint work in part with Romain Lebreton
LIRMM, Montpellier 2

Motivation

- topology of plane or space curves [Diochnos *et al.*, 2009], [Rouillier, 2010], [Emeliyaneko-Sagraloff, 2012], [Bouzidi *et al.*, 2013-4], [Kobel-Sagraloff 2014], [Diatta *et al.* 2014] ...



$$f = \frac{\partial f}{\partial Y} = 0$$

- point counting algorithm [Gaudry-S., 2012]
- useful to solve general polynomial systems [Giusti-Lecerf-Salvy, 2001]

We are talking about symbolic techniques:

- tools: elimination, resultant
- nothing about real or complex root isolation

Warm-up: computing resultants

f, g are in ...	$\mathbb{K}[Y]$
terms in f, g	$\Theta(d)$
terms in $r = \mathsf{Res}\,(f, g, y)$	1
computing r (best known bound)	$O\tilde{}(d)$
optimal?	yes, up to log factors

[Collins, '67], [Schönhage, '71], [Knuth, '71]

Warm-up: computing resultants

f, g are in ...	$\mathbb{K}[Y]$	$\mathbb{K}[X, Y]$
terms in f, g	$\Theta(d)$	$\Theta(d^2)$
terms in $r = \mathsf{Res}\,(f, g, y)$	1	$\Theta(d^2)$
computing r (best known bound)	$\tilde{O}(d)$	$\tilde{O}(d^3)$
optimal?	yes, up to log factors	no

[Collins, '67], [Schönhage, '71], [Knuth, '71], [Reischert, '97] or CRT

Warm-up: computing resultants

f, g are in ...	$\mathbb{K}[Y]$	$\mathbb{K}[X, Y]$	$\mathbb{K}[t, X, Y]$
terms in f, g	$\Theta(d)$	$\Theta(d^2)$	$\Theta(d^3)$
terms in $r = \mathsf{Res}\,(f, g, y)$	1	$\Theta(d^2)$	$\Theta(d^4)$
computing r (best known bound)	$O\tilde{}(d)$	$O\tilde{}(d^3)$	$O\tilde{}(d^5)$
optimal?	yes, up to log factors	no	no

[Collins, '67], [Schönhage, '71], [Knuth, '71], [Reischert, '97] or CRT

Warm-up: computing resultants

f, g are in ...	$\mathbb{K}[Y]$	$\mathbb{K}[X,Y]$	$\mathbb{K}[t,X,Y]$
terms in f, g	$\Theta(d)$	$\Theta(d^2)$	$\Theta(d^3)$
terms in $r = \mathsf{Res}(f, g, y)$	1	$\Theta(d^2)$	$\Theta(d^4)$
computing r (best known bound)	$O\tilde{}(d)$	$O\tilde{}(d^3)$	$O\tilde{}(d^5)$
optimal?	yes, up to log factors	no	no

[Collins, '67], [Schönhage, '71], [Knuth, '71], [Reischert, '97] or CRT

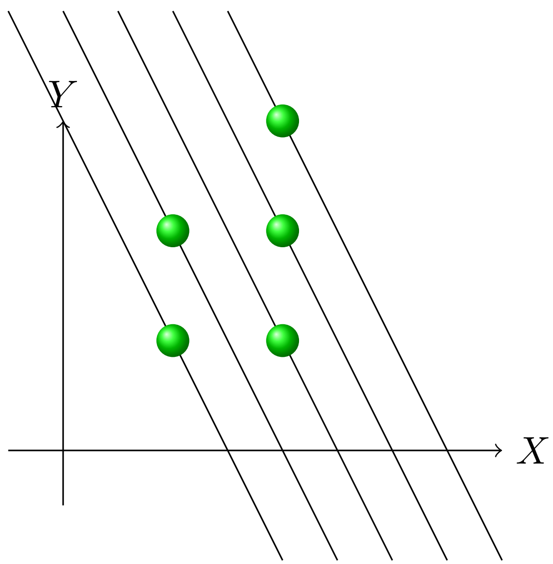
Boolean complexity

- bit-size in $\mathbb{Z}[X,Y] \simeq t$-degree in $\mathbb{K}[t,X,Y]$
- so resultant in $\mathbb{Z}[X,Y]$, with degree and bit-size d: $\Theta(d^4)$ bit, in $O\tilde{}(d^5)$ bit operations

Our problem

Given $f, g \in \mathbb{L}[X, Y]$, where $\mathbb{L}$ is a domain, return $\tau \in \mathbb{Z}$ and $P(X), S(X)$ such that

- τ is not too large ($\tau \leq d^4$)
- P monic, squarefree
- $\sqrt{f(X + \tau Y, Y), g(X + \tau Y, Y)} = \langle P(X), Y - S(X) \rangle$ over $\mathbb{K}[X, Y]$, with $\mathbb{K} = \text{frac}(\mathbb{L})$.



From that, one may compute with P only (count roots, isolate them, ...)

Our problem – refined

Suppose that $\mathbb{L} = \mathbb{Z}$ or $\mathbb{L} = \mathbb{K}[t]$. Length function λ:

- Over $\mathbb{Z}$: $\lambda(a) := \log(|a|)$
- Over $\mathbb{K}[t]$: $\lambda(h) := \deg(h)$

Suppose for simplicity that $\deg(f), \deg(g) \leq d$ and $\lambda(f), \lambda(g) \leq d$.

- output size: $\lambda(P) = O(d^2)$ but $\lambda(S) = O(d^4)$
- modified output: (Kronecker, Macaulay, Canny, Alonso *et al.*, Rouillier, Giusti *et al.*, ...)

$$R = P'S \bmod P$$

  then $\lambda(R) = O(d^2)$

- total: $O(d^4)$ bits / coefficients in $\mathbb{K}$

Main result

1. Over $\mathbb{Z}$. One can solve $f = g = 0$ by a Monte Carlo algorithm, with probability of success $> 1/2$, using $O(d^{4+\varepsilon})$ bit operations, for any $\varepsilon > 0$

1. symbolic Newton iteration [Giusti-Heintz-Pardo-. . . , 1990's]
2. deflation for multiple roots [Lecerf]
3. an extension of Kedlaya and Umans' algorithm [Poteaux-S.]

Optimality

- take $F^{(d)} = \prod_{i=1}^{d}(X - i), \quad G^{(d)} = \prod_{j=1}^{d}(Y - j)$
- output has total bit size $\tilde{\Theta}(d^4)$

2. Over $\mathbb{K}[t]$. The algorithm still works, but we're missing the equivalent of 3.

We still get something for simple roots, though.

Previous work

Easily seen to be polynomial time:

- [Gonzalez Vega - El Kahoui, 94], resultant and subresultants

Deterministic

- Bouzidi et al., 14, $O\tilde{}(d^6)$

Las Vegas

- Bouzidi et al., 14, $O\tilde{}(d^5)$

Monte Carlo

- $O\tilde{}(d^5)$ is easy (as for the resultant)

Lifting techniques

Let $\mathfrak{m}$ be a maximal ideal in $\mathbb{L}$: generated by either a prime or $t - t_0$.

$$
\begin{array}{rc}
 & (P(X), R(X)) \text{ over } \mathbb{K} = \mathrm{frac}(\mathbb{L}) \\
\text{rational reconstruction} & \Uparrow \\
 & (P(X), S(X)) \bmod \mathfrak{m}^{2^\ell} \\
\text{lifting step} & \Uparrow \\
 & \vdots \\
\text{lifting step} & \Uparrow \\
 & (P(X), S(X)) \bmod \mathfrak{m}^{2^1} \\
\text{lifting step} & \Uparrow \\
\text{initialization} & (P(X), S(X)) \bmod \mathfrak{m}^{2^0}
\end{array}
$$

Dominant part: lifting

Newton iteration – simple roots

Usual form:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} x_n \\ y_n \end{bmatrix} - \begin{bmatrix} \frac{\partial f}{\partial X}(x_n, y_n) & \frac{\partial f}{\partial Y}(x_n, y_n) \\ \frac{\partial g}{\partial X}(x_n, y_n) & \frac{\partial g}{\partial Y}(x_n, y_n) \end{bmatrix}^{-1} \begin{bmatrix} f(x_n, y_n) \\ g(x_n, y_n) \end{bmatrix}$$

For this to work nicely, the Jacobian matrix should be invertible at the root we are looking for.

Here: similar formulas to go from $(P(X), S(X)) \bmod \mathfrak{m}^k$ to $(P(X), S(X)) \bmod \mathfrak{m}^{2k}$

Bottleneck: function evaluation, that is computing

$$f(X, Y) \quad \mathrm{mod} \quad (P(X), Y - S(X), \mathfrak{m}^{2k}), \quad \ldots$$

Remark: multivariate versions of Newton iteration often assume that the input is given by a Straight Line Program. We don't.

Over $\mathbb{L} = \mathbb{Z}$

Kedlaya and Umans' algorithm computes $g(S) \bmod P$ in $\mathbb{Z}/N\mathbb{Z}[X]$ using

$$O\tilde{}(d^{1+\varepsilon}\log(N))$$

bit operations, for any $\varepsilon > 0$

- quasi-optimal
- specific to the boolean model

Over $\mathbb{L} = \mathbb{Z}$

Kedlaya and Umans' algorithm computes $g(S)$ mod P in $\mathbb{Z}/N\mathbb{Z}[X]$ using

$$O\tilde{}(d^{1+\varepsilon}\log(N))$$

bit operations, for any $\varepsilon > 0$

- quasi-optimal
- specific to the boolean model

Here (following [Poteaux-S., 2013])

- f is in $\mathbb{Z}[X, Y]$
- reduction modulo $\langle P(X), Y - S(X)\rangle$ = computing $f(X, S)$ mod P
- $N = p^{2k}$, with $\log(p^{2k}) = O(d^2)$
- takes $O(d^{4+\varepsilon})$ bit operations

Over $\mathbb{L} = \mathbb{Z}$

To compute $g(S) \bmod P$, in a nutshell.

1. Forget about $\bmod P$

   1. make g multivariate: find e.g. $G(X, Y)$ such that $g = G(X, X^2)$
   2. so we compute $G(S, S^2 \bmod P) \bmod P$
   3. if we have enough variables, $G(S, S^2 \bmod P, \ldots, S^k \bmod P)$ has a small degree, so we can compute it first, then reduce

Over $\mathbb{L} = \mathbb{Z}$

To compute $g(S)$ mod P, in a nutshell.

1. Forget about $\bmod P$

1. make g multivariate: find e.g. $G(X, Y)$ such that $g = G(X, X^2)$
2. so we compute $G(S, S^2 \bmod P) \bmod P$
3. if we have enough variables, $G(S, S^2 \bmod P, \ldots, S^k \bmod P)$ has a small degree, so we can compute it first, then reduce

2. Reduction to evaluation and interpolation

1. evaluate $S_1 = S, \ldots, S_k = S^k \bmod P$ at $(x_1, x_2, \ldots)$
   we get points $(p_1, p_2, \ldots)$ in k-space
2. evaluate G at $(p_1, p_2, \ldots)$ — hard!
3. interpolate

Over $\mathbb{L} = \mathbb{Z}$

3. Multivariate multipoint evaluation mod p

- if p is very small, the points almost fill up a cube easy
- otherwise, forget the mod p
  - work over $\mathbb{Z}$
  - by working modulo smaller p's

Over $\mathbb{Z}$ or $\mathbb{K}[t]$

Brent and Kung's algorithm computes $g(S) \bmod P$ in $\mathbb{A}[X]$ using

$$\tilde{O}(d^{\frac{\omega+1}{2}})$$

operations in $\mathbb{A}$, using baby steps / giant steps techniques

- $\sqrt{d}$ polynomial multiplications in degree d
- $\sqrt{d}$ matrix multiplications in size $\sqrt{d}$

Over $\mathbb{Z}$ or $\mathbb{K}[t]$

Brent and Kung's algorithm computes $g(S) \bmod P$ in $\mathbb{A}[X]$ using

$$O\tilde{}(d^{\frac{\omega+1}{2}})$$

operations in $\mathbb{A}$, using baby steps / giant steps techniques

- $\sqrt{d}$ polynomial multiplications in degree d
- $\sqrt{d}$ matrix multiplications in size $\sqrt{d}$

Here

- f is in $\mathbb{K}[t, X, Y]$
- we reduce modulo $\langle P(X), Y - S(X)\rangle$
- takes $O\tilde{}(d^{\frac{\omega+7}{2}})$ operations in $\mathbb{K}$
  $\sqrt{d}$ product of polynomial matrices of size $\sqrt{d}$ with entries of degree d and coefficients of size d^2

Remark: improvement using [Huang-Pan 1998]: $4.685 \to 4.667$

First experiments

- Implemented in C++ using NTL [Shoup, 1995]
- $f, g \in \mathbb{Z}[X, Y]$, random, of degree less than d, $\lambda(f) < 64, \lambda(g) < 64$
- Using naive matrix multiplication ($\omega = 3$), so our cost is $\tilde{O}(d^5)$

d	precision	Lifting	$\mathrm{CRT}_{\mathrm{ZZ}}$	$\mathrm{CRT}_{\mathrm{zz}}$
120	32	421	2711	1990
120	64	774	5422	3980
120	128	1728	10845	7961
140	32	818	4902	2671
140	64	1486	9804	5343
140	128	**3045**	19608	**10687**
160	32	1072	7610	5293
160	64	1896	15221	10587
160	128	3958	30442	21174
180	32	1394	11121	6541
180	64	2399	22242	13097
180	128	**4951**	44485	**26195**

Handling multiplicities

First remarks

- Newton iteration not defined at multiple roots
- easy fix: compute only the non-singular roots

In general: deflation techniques

- several approaches [Ojika et al 83], [Lecerf 02], [Leykin et al. 06], ...
- Lecerf's approach
  - involves 2 new variables
  - "symbolic" deflation, no SVD.

Challenge: make it work in our context with an acceptable complexity.

Lecerf's deflation lemma

Suppose $(x, y) \subset \mathbb{K}^2$ is an isolated root of $F = G = 0$, of multiplicity M

- find the smallest m such that either

$$\frac{\partial^m F}{\partial Y^m}(x, y) \neq 0 \quad \text{or} \quad \frac{\partial^m G}{\partial Y^m}(x, y) \neq 0.$$

Lecerf's deflation lemma

Suppose $(x, y) \subset \mathbb{K}^2$ is an isolated root of $F = G = 0$, of multiplicity M

- find the smallest m such that either

$$\frac{\partial^m F}{\partial Y^m}(x, y) \neq 0 \quad \text{or} \quad \frac{\partial^m G}{\partial Y^m}(x, y) \neq 0.$$

- implicit function theorem: there exists J_x in $\mathbb{K}[[\zeta]]$ such that (say)

$$\frac{\partial^{m-1} F}{\partial Y^{m-1}}(\zeta + x, J_x) = 0, \quad J_x(0) = y.$$

Lecerf's deflation lemma

Suppose $(x, y) \subset \mathbb{K}^2$ is an isolated root of $F = G = 0$, of multiplicity M

- find the smallest m such that either

$$\frac{\partial^m F}{\partial Y^m}(x, y) \neq 0 \quad \text{or} \quad \frac{\partial^m G}{\partial Y^m}(x, y) \neq 0.$$

- implicit function theorem: there exists J_x in $\mathbb{K}[[\zeta]]$ such that (say)

$$\frac{\partial^{m-1} F}{\partial Y^{m-1}}(\zeta + x, J_x) = 0, \quad J_x(0) = y.$$

- Claim: one of the power series, say S_x, among

$$\frac{\partial^\ell F}{\partial Y^\ell}(\zeta + x, J_x), \quad \frac{\partial^\ell G}{\partial Y^\ell}(\zeta + x, J_x) \quad (\ell \leq m)$$

has valuation $n \leq M/m$.

Deflation and lifting

What we need

- compute J_x such that $\frac{\partial^{m-1} F}{\partial Y^{m-1}}(\zeta + x, J_x) = 0$
- compute a power series of the form $S_x = \frac{\partial^\ell G}{\partial Y^\ell}(\zeta + x, J_x) \bmod \zeta^{n+1}$, for some $\ell \leq m$.

Main loop (if all roots of P have the same "signature")

- start from $(P^\star, S^\star) = (P \bmod p^k, S \bmod p^k)$
- compute series as before, working in $\mathbb{Z}/p^{2k}\mathbb{Z}[X]/\langle P^\star \rangle$
  - done by computing $G(X + \zeta, J_x + \xi) \bmod \langle P^\star, \zeta^{n+1}, \xi^{m+1} \rangle$
    same kind of reduction as before, in 3 variables
  - $\deg(P) nm \leq d^2$
- deduce $(P \bmod p^{2k}, S \bmod p^{2k})$ easy

In general

We have several factors, with several values of (m, n).

What we really need: given

- integers (m_i, n_i)
- polynomials P_i in $\mathbb{A}[X]$ of degrees e_i $\qquad \mathbb{A} = \mathbb{Z}/p^k\mathbb{Z}$
- J_i in $\mathbb{A}[X, \zeta]_{e_i, n_i+1}$,

compute

- all $\frac{\partial^{m_i} G}{\partial Y^{m_i}}(x + \zeta, J_i) \bmod \langle P_i, \zeta^{n_i+1} \rangle$

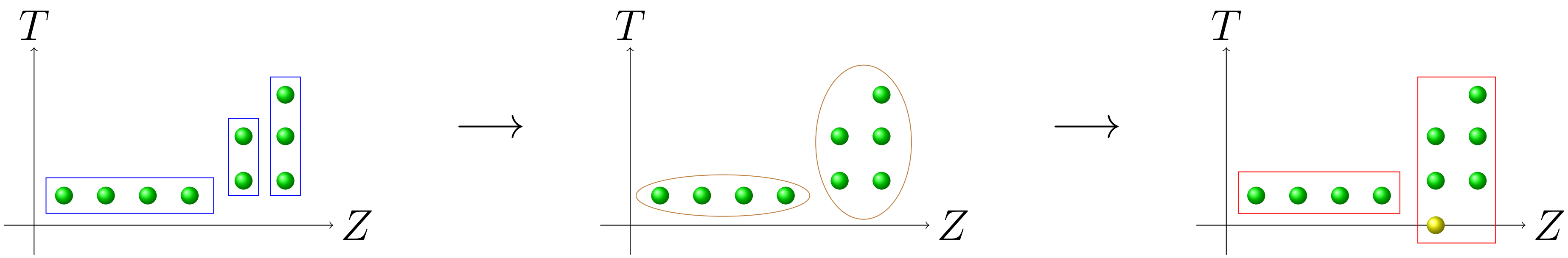
We compute all $G(X + \zeta, J_i + \xi) \bmod \langle P_i, \zeta^{n_i+1}, \xi^{m_i+1} \rangle$.

Remark: $\sum_i e_i n_i m_i \leq d^2$.

Multiple reduction

[Poteaux - S.] we can reduce modulo one triangular set efficiently (quasi-optimal)

- we shouldn't process of all them independently
  (like multipoint evaluation of polynomials)
- we shouldn't merge them all together
  (degrees too large in T)
- merge those with similar T-degree (say between 2^i and $2^{i+1}-1$)



Altogether, $O(\log(d))$ reduction, each of them in time $d^{4+\varepsilon}$.