Worst-case algorithm for solving bivariate systems

Yacine Bouzidi

Sylvain Lazard Marc Pouget Fabrice Rouillier

Inria Nancy - Grand Est

Yacine.bouzidi@inria.fr

15 mai 2013

Plan

Plan

Problem and motivation

Let $P, Q \in \mathbb{Z}[x, y]$ of degree at most d and maximum bitsize τ

- **Problem :**
  - Solving systems of the form $I = \{P, Q\}$ i.e. Isolating the real solutions of I
  - Perform operations with the solutions (eg. IsZero, SignAt, etc)
- **Motivation**
  - Topology and arrangement of planar algebraic curves
  - Eg. Plot a curve C_f defined by $f = 0$ with $f \in \mathbb{Q}[x, y]$
    - Computing isolating box around critical points of $C_f \rightsquigarrow$ solve the system $\{f, \frac{\partial f}{\partial y}\}$
    - Determination of the topology and the geometry inside these boxes
    - Connection between critical points by means of polylines

Plan

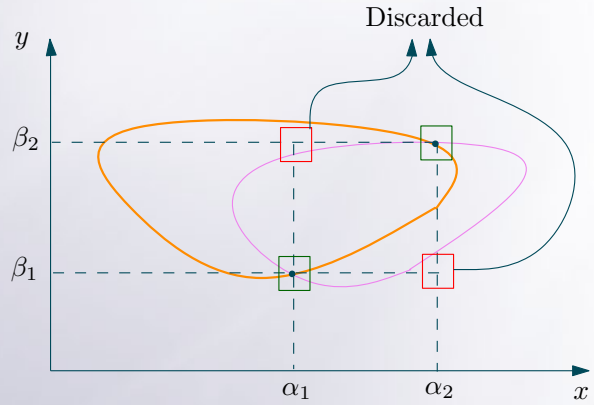
Solving algebraic systems

- **Numerical methods : + Fast / - Certification**
  - Subdivision [Neumaier. 90] . . .
  - Homotopy continuation [Sommese and Wampler. 05] . . .
- **Formal solutions : + Certification / - Symbolic computation**
  - Rational parametrization
    - Gröbner basis + linear algebra [Rouillier, 96][Bostan, Salvy and Schost. 01]
    - Geometric resolution [Lecerf and Durvye. 06]
    - Chow form [Roy and al. 94]
    - . . .
  - Triangular decomposition [Aubry, Lazard and Moreno m. 99]

Solving bivariate systems state-of-the-art algorithm

[Sagraloff and al. 2012]

- **The idea :**
  - Project the solutions on x and y by means of resultant computations
  - Isolate the roots of these resultants
  - Keep/Discard the candidate solutions (pair of projected roots)



- **Advantage :** reduces the amount of symbolic computations
- **Drawback :**
  - some information is lost (multiplicity)
  - less-suited output for some further computations

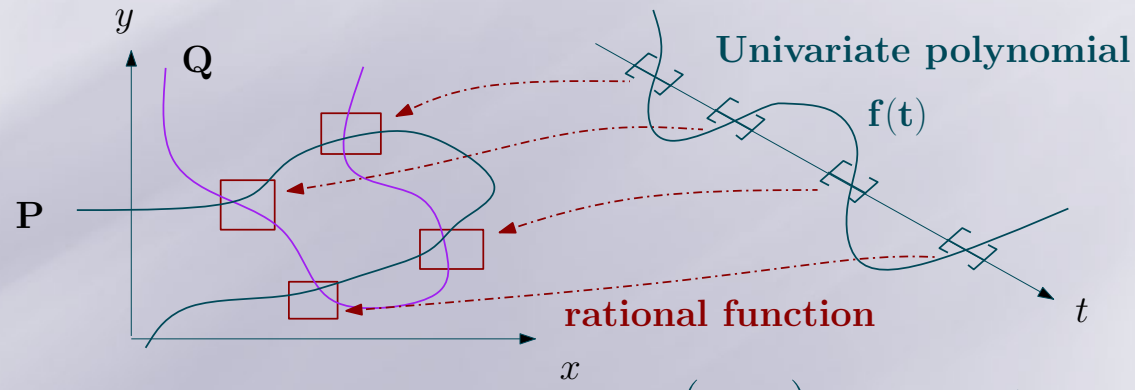
Bit-complexity $\rightsquigarrow \tilde{O}_B(d^8 + d^7\tau)$

Rational Univariate Representation

Definition (RUR)

Let $I =< P, Q >$ be a zero-dimensional ideal and V its variety.
A RUR of I is the datum of :

- A linear form $x + ay$ that **separates** the points of V
- A **bijective** mapping between the solutions of V and the roots of an univariate polynomial f



$\mathbf{Mapping\ (x, y) \mapsto x + ay}$

$$\begin{array}{rcl} (\mathbf{x}, \mathbf{y}) & \mapsto & \mathbf{x + ay} \\ (\frac{\mathbf{f_x(t)}}{\mathbf{f_1(t)}}, \frac{\mathbf{f_y(t)}}{\mathbf{f_1(t)}}) & \hookleftarrow & \mathbf{t} \end{array}$$

Rational Univariate Representation

- Knowing a RUR **reduces** most operations on V to univariate operations
  - Isolating boxes around the real points of $V \rightsquigarrow$ isolating the real roots of f and plugging them into the rational fractions
  - The sign of $F(x, y)$ at $(\alpha, \beta) \in V$ is the sign of $F(\frac{f_x(t)}{f_1(t)}, \frac{f_y(t)}{f_1(t)})$ at $\alpha + a\beta$

Complexity bounds (bivariate case)

- **Existing algorithms**
  - RUR via Triangular decomposition [Gonzalez-V. and al. 96] [Tsigaridas and al. 09]
    - Separating form : $\tilde{O}_B(d^{10} + d^9\tau)$
    - Triangular decomposition using subresultants : $\tilde{O}_B(d^7 + d^6\tau)$
    - Size of the output : $O(d)$ RURs, each of size $\tilde{O}((d^4 + d^3\tau))$
  - RUR via Gröbner basis + linear algebra in $\mathbb{Q}[x, y]/I$ [Rouillier. 96][Bostan, Salvy and Schost. 01]
    - Separating form and RUR : $\tilde{O}_B(d^{10} + d^9\tau)$
    - RUR of size : $\tilde{O}((d^5 + d^4\tau))$ [Dahan and Schost. 04] (only for specific instances)
- **New algorithm** for computing a separating linear form and a RUR in $\tilde{O}_B(d^8 + d^7\tau)$
  - Separating form $\rightsquigarrow \tilde{O}_B(d^8 + d^7\tau)$
  - RUR $\rightsquigarrow \tilde{O}_B(d^7 + d^6\tau)$ of size $\tilde{O}(\quad d^2(d^2 + d\tau))$

Complexity bounds (bivariate case)

- **Existing algorithms**
  - RUR via Triangular decomposition [Gonzalez-V. and al. 96] [Tsigaridas and al. 09]
    - Separating form : $\tilde{O}_B(d^{10} + d^9\tau)$
    - Triangular decomposition using subresultants : $\tilde{O}_B(d^7 + d^6\tau)$
    - Size of the output : $O(d)$ RURs, each of size $\tilde{O}((d^4 + d^3\tau))$
  - RUR via Gröbner basis + linear algebra in $\mathbb{Q}[x, y]/I$ [Rouillier. 96][Bostan, Salvy and Schost. 01]
    - Separating form and RUR : $\tilde{O}_B(d^{10} + d^9\tau)$
    - RUR of size : $\tilde{O}((d^5 + d^4\tau))$ [Dahan and Schost. 04] (only for specific instances)
- **New algorithm** for computing a separating linear form and a RUR in $\tilde{O}_B(d^8 + d^7\tau)$
  - Separating form $\rightsquigarrow \tilde{O}_B(d^8 + d^7\tau)$
  - RUR $\rightsquigarrow \tilde{O}_B(d^7 + d^6\tau)$ of size $\tilde{O}_B(d^2(d^2 + d\tau))$

Plan

RUR via Chow form

u, v two indeterminates.

$f_I(t,u,v) = \prod_{(\alpha,\beta)\in V(I)}(t-(u\alpha+v\beta))^{\mu(\alpha,\beta)}$ the characteristic pol. of the matrix of the multiplication by $ux+vy$ in $\frac{\mathbb{Q}[x,y,u,v]}{I}$

$g_I(t,u,v) = \prod_{(\alpha,\beta)\in V(I)}(t-(u\alpha+v\beta))^{\mu(\alpha,\beta)-1}$

RUR via Chow form [Roy and al. 94]

If u_0x+v_0y separates $V(I)$ then,

$$f(t) = f_I(t,u_0,v_0) \qquad f_1(t) = \frac{\frac{\partial f_I(t,u,v)}{\partial t}(t,u_0,v_0)}{g_I(t,u_0,v_0)}$$

$$f_y(t) = \frac{\frac{\partial f_I(t,u,v)}{\partial v}(t,u_0,v_0)}{g_I(t,u_0,v_0)} \qquad f_x(t) = \frac{\frac{\partial f_I(t,u,v)}{\partial u}(t,u_0,v_0)}{g_I(t,u_0,v_0)}$$

Resultant

Let $P = \sum_{i=0}^{p} a_i y^i$ and $Q = \sum_{i=0}^{q} b_i y^i$ be two non-zero polynomials of degree p and q in $\mathbb{D}[X]$.

$$Syl(P,Q) = \overbrace{\begin{pmatrix} a_p & a_{p-1} & \cdots & \cdots & a_0 & & \\ & a_p & a_{p-1} & \cdots & \cdots & a_0 & \\ & & \ddots & & & & \ddots \\ & & & a_p & a_{p-1} & \cdots & \cdots & a_0 \\ b_q & b_{q-1} & \cdots & & b_0 & & & \\ & b_q & b_{q-1} & \cdots & & b_0 & & \\ & & \ddots & & & & \ddots & \\ & & & \ddots & & & & \ddots \\ & & & & b_q & b_{q-1} & \cdots & b_0 \end{pmatrix}}^{\text{p+q columns}} \left.\begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix}\right\} \begin{matrix} \text{q rows} \\ \\ \\ \\ \text{p rows} \\ \\ \\ \\ \end{matrix}$$

$Res_y(P,Q)$ is the determinant of $Syl(P,Q)$

Resultant

$$P = \sum_{i=0}^{p} a_i(x)y^i \quad \text{and} \quad Q = \sum_{i=0}^{q} b_i(x)y^i \quad \text{in} \quad \mathbb{Z}[x, y]$$

Theorem

If $P, Q \in \mathbb{Z}[x, y]$ coprime and $a_p(x)$ and $b_q(x)$ have no common root, then $Res_y(P, Q) = c \prod_{(\alpha,\beta)\in V(P,Q)} (x - \alpha)^{\mu(\alpha,\beta)}$, $c \in \mathbb{Z}^*$

Geometrically, $Res_y(P, Q) \in \mathbb{Z}[x]$ is a polynomial whose roots are the projections on the x-axis of the intersection points of P and Q (possibly at infinity)

Corollary

Let $t = x + sy$ a "generic" linear form and consider the polynomials $\tilde{P}(t - sy, y), \tilde{Q}(t - sy, y)$ then,

$$Res_y(\tilde{P}, \tilde{Q}) = L_R(s) \prod_{(\alpha,\beta)\in V(P,Q)} (t - (\alpha + s\beta))^{\mu(\alpha,\beta)}$$

Computing the RUR associated to a

- The polynomial $R(t,s) = Res_y(\tilde{P}, \tilde{Q})$ is up to a factor, the Chow form of $< P, Q >$ with $(u,v) = (1,s)$

Theorem

If $t = x + ay$ *separates* $V(< P, Q >)$, *then,*

$$f(t) = \frac{R(t,a)}{L_R(a)} \qquad f_1(t) = \frac{f'(t)}{\gcd(f(t), f'(t))}$$

$$f_y(t) = \frac{\frac{\partial R}{\partial s}(t,a) - f(t)\frac{\partial L_R}{\partial s}(a)}{L_R(a)\gcd(f(t), f'(t))} \qquad f_x(t) = tf_1(t) - d_t(f)\overline{f(t)} - af_y(t).$$

Bit-complexity $\rightsquigarrow \tilde{O}_B(d^7 + d^6\tau)$ (Computation of $R(t,s)$)

Bitsize $\rightsquigarrow \tilde{O}(d^2\tau_a + d\tau)$ (Bounds on the bitsize of resultants + Mignotte)

Plan

Computing a separating form $x + ay$

- **Classical algorithm :**

  There are at most $\binom{d^2}{2}$ bad choices of a which is the maximum number of alignements defined by at most d^2 solutions (Bézout's bound)

Algorithm : naive separating form

- Compute $R(t,s) = Res_y(P(t-sy,y), Q(t-sy,y))$
- For $d^4 > \binom{d^2}{2}$ choices of a
  - compute the polynomial $R(t,a)$, the specialization of $R(t,s)$ at a
  - compute $\overline{R(t,a)}$ the squarefree part of $R(t,a)$
- Select the a for which the degree of $\overline{R(t,a)}$ is maximal

Bit-complexity $\rightsquigarrow \tilde{O}_B(d^{10} + d^9\tau)$ (d^4 squarefree part computations)

Computing a separating form $x + ay$

- Work over $\frac{\mathbb{Z}}{p\mathbb{Z}}$ to avoid coefficient swell
- **Problem :** $x + ay$ is separating over $\frac{\mathbb{Z}}{p\mathbb{Z}}$ do not implies that it is also separating over $\mathbb{Z}$ . . . Except under some conditions !

Theorem

Let p *be a prime such that* $Lc_y(P(t - sy, y))Lc_y(Q(t - sy, y))$ *do not vanish modulo* p *and* $\#V(< P_p, Q_p >) = \#V(< P, Q >)$ *then,*
$x + ay$ *separates* $V(< P_p, Q_p >) \Rightarrow x + ay$ *separates* $V(< P, Q >)$

- Previous algorithm then runs in $\tilde{O}_B(d^8)$

Goal : compute a prime p satisfying the above conditions

Computing a separating form $x + ay$

Denote by $\mathcal{N}$ (resp. $\mathcal{N}_p$) the number of distinct solutions of $< P, Q >$ (resp. $< P_p, Q_p >$)

Lemma

Let p be a prime number such that $Lc_y(P(t - sy, y))Lc_y(Q(t - sy, y))$ do not vanish modulo p then, $\mathcal{N}_p \leq \mathcal{N}$

Lemma

There exist at most $B = \tilde{\Theta}(d^4 + d^3\tau)$ prime numbers such that $\mathcal{N}_p < \mathcal{N}$

Algorithm : Computing good prime

- For $B = \tilde{\Theta}(d^4 + d^3\tau)$ prime number p that do not cancel $Lc_y(P(t - sy, y))Lc_y(Q(t - sy, y))$
  - Compute $\mathcal{N}_p \rightsquigarrow \tilde{O}_B(d^4)$?
- Choose p that maximizes $\mathcal{N}_p$

Computing $\mathcal{N}_p$

- **Idea :** Compute $\sum_{(\alpha,\beta)\in V} \mu(\alpha,\beta)$ and $\sum_{(\alpha,\beta)\in V} (\mu(\alpha,\beta)-1)$

Algorithm : Computing $\mathcal{N}_p$

- Triangular decomposition of $\{P_p, Q_p\} \rightsquigarrow \{F_i(x), G_i(x,y)\}_{i\in\mathcal{I}}$

  $\rightsquigarrow \sum_{(\alpha,\beta)\in V} \mu(\alpha,\beta) = \sum_{i\in\mathcal{I}} i\mathsf{deg}(F_i)$
- Triangular decomposition of $\{G_i, \frac{\partial G_i}{\partial y}\} \rightsquigarrow \{F_{i,j}(x), G_{i,j}(x,y)\}_{j\in\mathcal{J}_i}$

  $\rightsquigarrow \sum_{(\alpha,\beta)\in V} (\mu(\alpha,\beta)-1) = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}_i} j\mathsf{deg}(F_{i,j})$
- Return $\sum_{i\in\mathcal{I}} [i\mathsf{deg}(F_i) - \sum_{j\in\mathcal{J}_i} j\mathsf{deg}(F_{i,j})]$

Conclusion

- New algorithm for computing a separating linear form that improves by a factor d^2 the best known complexity
- New bound on the bitsize of the polynomials of the RUR that matches that of Gonzalez-vega and al. but with only one RUR instead of $O(d)$
- The same complexity holds for sign_at operation and is_in_radical

Thank you
for your attention

Conclusion

- New algorithm for computing a separating linear form that improves by a factor d^2 the best known complexity
- New bound on the bitsize of the polynomials of the RUR that matches that of Gonzalez-vega and al. but with only one RUR instead of $O(d)$
- The same complexity holds for sign_at operation and is_in_radical

Thank you
for your attention