Computer Algebra
Lecture 6

James Davenport

University of Bath

10 September 2018

An example from Lecture 5

$$
\begin{aligned}
x^2 - 1 &= 0 \qquad (1)\\
y^2 - 1 &= 0 \qquad (2)\\
(x-1)(y-1) &= 0 \qquad (3)
\end{aligned}
$$

Solutions:
$(-1, 1), (1, -1), (1, 1)$
These equations *are* the simplest description of this variety.
Solve equation (1).

$x = 1$ (3) disappears, and we are left with (2): $y = \pm 1$.

$x = -1$ (3) becomes $-2(y-1) = 0$, so $y = 1$, which also satisfies (2).

A bigger example from Lecture 5

$$
\begin{aligned}
x^{2000} - 1 &= 0 \qquad (4) \\
y^{2000} - 1 &= 0 \qquad (5) \\
(x^{1000} - 1)(y^{1000} - 1) &= 0 \qquad (6)
\end{aligned}
$$

We could consider each solution of (4) separately, but this is boring.
Instead consider
$\gcd(x^{2000} - 1, \mathrm{lc}_y((x^{1000} - 1)(y^{1000} - 1))) = x^{1000} - 1$.
Then we have the same case analysis as before.

Can we generalise?

For zero-dimensional ideals, with a `lex` ordering a Gröbner base has to look like.

$$\begin{gathered} p_n(x_n) \\ p_{n-1,1}(x_{n-1},x_n),\ldots,p_{n-1,k_{n-1}}(x_{n-1},x_n), \\ p_{n-2,1}(x_{n-2},x_{n-1},x_n),\ldots,p_{n-2,k_{n-2}}(x_{n-2},x_{n-1},x_n), \\ \cdots \\ p_{1,1}(x_1,\cdots,x_{n-1},x_n),\ldots,p_{1,k_1}(x_1,\cdots,x_{n-1},x_n), \end{gathered}$$

where k_i is the number of polynomials involving x_i but not any x_j for $j < i$ and

$$\deg_{x_i}(p_{i,j}) \le \deg_{x_i}(p_{i,j+1}) \tag{7}$$

and p_{i,k_i} is monic in x_i.

Another way of looking at this

$$
\begin{array}{ccccc}
x_1 & \cdots & x_{n-2} & x_{n-1} & x_n \\
 & & p_{1,k_1} & & \\
\hline
 & & & \vdots & \\
 & & p_{1,1} & & \\
\hline
 & \ddots & & \vdots & \\
 & & p_{n-2,k_{n-2}} & & \\
\cline{3-5}
 & & & \vdots & \\
 & & p_{n-2,1} & & \\
\cline{3-5}
 & & & p_{n-1,k_{n-1}} & \\
\cline{4-5}
 & & & \vdots & \\
 & & & p_{n-1,1} & \\
\cline{4-5}
 & & & & p_n
\end{array}
$$

Essentially upper triangular. except that, for any i there may be several $p_{i,j}$. What do they do?

Those extra equations

As with $(x-1)(y-1)$ in the example, they serve to rule out some of the solutions. If we had just the p_{i,k_i}, and these had degree d_i, the number of solutions would be $\prod_{i=1}^n d_i$. As it is, we have $\prod_{i=1}^n d_i - \#\{\text{those } \alpha \text{ barred by } k_{i,j}(\alpha) \neq 0\}$.
Let $G_k = G \cap k[x_k, \ldots, x_n]$, i.e. those polynomials in $x_k, \ldots, x_n$ only.

Theorem (Gianni–Kalkbrener [Gia89, Kal89])

Let α be a solution of G_{k+1}. Then if $\mathrm{lc}_{x_k}(p_{k,i})$ vanishes at α, then $(p_{k,i})$ vanishes at α. Furthermore, the lowest degree (in x_k) polynomial of the $p_{k,i}$ not to vanish at α, say p_{k,m_α}, divides all of the other $p_{k,j}$ at α. Hence we can extend α to solutions of G_k by adding $x_k = \mathrm{RootOf}(p_{k,m_\alpha})$.

Put another way, if p_{k,m_α} allows a solution extending α, so do $p_{k,m_\alpha+1}, p_{k,m_\alpha+2}, \ldots$.

The Algorithm

This gives us an algorithm to describe the solutions of a zero-dimensional ideal from such a Gröbner base G. This is essentially a generalisation of back-substitution into triangularised linear equations, except that there may be more than one solution, since the equations are non-linear, and possibly more than one equation to substitute into.

Algorithm (Gianni–Kalkbrener)

1: **procedure** $\mathrm{GK}(G, n)$ ▷ *G 0-dim, lex GB*
2: $S := \{x_n = \mathrm{RootOf}(p_n)\}$
3: **for** $k = n-1, \ldots, 1$ **do**
4: $S := \mathrm{GKstep}(G, k, S)$
5: **end for**
6: **return** S
7: **end procedure**

The Single Step

procedure GKSTEP(G, k, A) ▷ A a list of solutions of G_{k+1}.
▷ Output A list of solutions of G_k.
    $B := \emptyset$
    for $\alpha \in A$ **do**
        $i := 1$
        while $(L := (\mathrm{lc}_{x_k}(p_{k,i}))(\alpha)) = 0$ **do**
            $i := i + 1$
        end while
        if L is invertible with respect to α **then**
            $B := B \cup \{(\alpha \cup \{x_k = \mathrm{RootOf}(p_{k,i}(\alpha))\})\}$
        else ▷ α is split as $\alpha_1 \cup \alpha_2$
            $B := B \cup \mathrm{GKstep}(G, k, \{\alpha_1\}) \cup \mathrm{GKstep}(G, k, \{\alpha_2\})$
        end if
    end for
    return B
end procedure

What does step 9 mean?

"**if** L is invertible with respect to α"

$$p_2 = x^2 - 1 \tag{8}$$

$$p_{1,1} = (x-1)(y-1) \tag{9}$$

$$p_{1,2} = y^2 - 1 \tag{10}$$

(8) says $x = \mathrm{RootOf}(x^2 - 1)$. Next polynomial is (9)
Is $x - 1$ invertible when $x = \mathrm{RootOf}(x^2 - 1)$?

Common sense $x = \pm 1$ and $x - 1 = 0$ when $x = 1$: not invertible, but $x - 1 = -2$ when $x = -1$: invertible.

Algorithm $x - 1 = \gcd(x-1, x^2-1)$, so write $x^2 - 1 = (x-1)(x+1)$ and consider the factors separately α_1 and α_2

But if α is $(x_n = \mathrm{RootOf}(p_n), x_{n-1} = \mathit{RootOf}(p_{n-1,j_{n-1}}), \ldots, x_{k+1} = \mathrm{RootOf}(f))$ this $\gcd(L, f)$ has to be computed allowing for $x_n = \mathrm{RootOf}(p_n), x_{n-1} = \mathit{RootOf}(p_{n-1,j_{n-1}}), \ldots$.

Theorem only true in dimension 0

Example ([FGT01, Example 3.6])

Let $I = \langle ax^2 + x + y, bx + y \rangle$ with the order $a \prec b \prec y \prec x$. The Gröbner base is $B_1 \cup B_2$ and there are no polynomials in (a, b) only, in (a, b, y) we have $B_2 := \{ay^2 + b^2y - by\}$, and in all variables $B_1 := \{ax^2 + x + y, axy - by + y, bx + y\}$.

normally: B_2 gives y (generally as the solution of a quadratic), then $x = -y/b$ except when $b = 0$,when $ay^2 = 0$ so $y = 0$, and $ax^2 + x = 0$, so $x = 0$ or $x = -1/a$.

$a = b = 0$: B_2 vanishes, so we would be tempted, by analogy with Theorem 1, to deduce that y is undetermined. But in fact $B_1|_{a=b=0} = \{x + y, y, y\}$, so $y = 0$ (and then $x = 0$).

$a = 0, b = 1$: Again B_2 vanishes. This time, $B_1|_{a=0,b=1} = \{x + y, 0, x + y\}$, and y is undetermined, with $x = -y$.

Lexicographic orderings are great

They are essentially the equivalent of “upper triangular” for matrices.

But they are much more expensive to compute.

Algorithm (Faugère–Gianni–Lazard–Mora [FGLM93])

Input *A Gröbner base G for a zero-dimensional ideal I with respect to $>'$; an ordering $>''$.*

Output *A Gröbner base H for I with respect to $>''$.*

Details in the paper, but we see it as a “black box”.

There is also an algorithm for non-zero-dimensional ideals: “The Gröbner Walk” [CKM97]

Proposition (Lecture 5 Slide 19)

If it's finite, the number (counted with multiplicity) of solutions of a system with Gröbner basis G is equal to the number of monomials which are not reducible by G.

Algorithm (Solve Polynomial Equations)

procedure SOLVE(*S*) ▷ *A set S of polynomials*
  $G :=$ *Buchberger*$(S, \prec_{\mathrm{tdeg}})$
  if *G is not zero-dimensional* **then**
    return *"not zero-dimensional"*
  else *Proposition says how many solutions*
    $H :=$ *FGLM*$(G, \prec_{\mathrm{lex}})$
    Use Gianni–Kalkbrener to solve H
    We can check the solution count against Proposition
  end if
end procedure

In the author's experience, describing the solutions of a set of polynomial equations when the dimension is not zero is still rather an art form (Slide 10), but much aided by the computation of Gröbner bases

Bibliography I

S. Collart, M. Kalkbrener, and D. Mall.
Converting Bases with the Gröbner Walk.
J. Symbolic Comp., 24:465–469, 1997.

J.-C. Faugère, P. Gianni, D. Lazard, and T. Mora.
Efficient Computation of Zero-Dimensional Gröbner Bases by Change of Ordering.
J. Symbolic Comp., 16:329–344, 1993.

E. Fortuna, P. Gianni, and B. Trager.
Degree reduction under specialization.
J. Pure Appl. Algebra, 164:153–163, 2001.

P. Gianni.
Properties of Gröbner bases under specializations.
In *Proceedings EUROCAL 87*, pages 293–297, 1989.

Bibliography II

M. Kalkbrener.
Solving systems of algebraic equations by using Gröbner bases.
In *Proceedings EUROCAL 87*, pages 282–292, 1989.