MATH 392 – Seminar in Computational Commutative Algebra
Second Computer Laboratory Day
February 18, 2019

Background and Goals

Maple contains a package called `Groebner` that contains facilities for defining monomial orders, finding leading terms and monomials of polynomials carrying out the division algorithm in $k[x_1, \ldots, x_n]$, computing Gröbner bases, and related computations. Our lab session today will introduce you to these facilities. Some problems at the end will be part of Problem Set 5, due Friday, February 22.

The `Groebner` *Package*

Launch Maple (preferably in the worksheet interface), then enter the command

```
with(Groebner);
```

The output should be the list of commands in this new package: `[Basis, FGLM, Hilbert-Dimension, etc.]`.

Monomial Orders in Maple

Maple allows you to define all of the monomial orders we have discussed in class (and many, many others too!).

- Lex orders are obtained with the specification `plex(varlist)`, where `varlist` is the list of variables in *decreasing order*. For instance `plex(x,y,z)` gives a lex order with $x > y > z$.
- Graded lex orders are obtained with the specification `grlex(varlist)`
- Graded reverse lex orders are obtained wiht the specification `tdeg(varlist)`

These will be included in the commands for doing polynomial division, computing Gröbner bases, etc. *as options*. If you ever want to check what a leading term of a polynomial is with respect to a given monomial order, the `LeadingTerm` command returns the leading coefficient and the leading monomial. For instance, try

```
LeadingTerm(6*x*y^5 + x^4*y,plex(x,y));
LeadingTerm(6*x*y^5 + x^4*y,tdeg(x,y));
```

Why are the results different?

Polynomial Division in Maple

The multivariable polynomial division algorithm we have discussed is implemented in the Maple command `NormalForm` from the `Groebner` package. The format is

```
NormalForm(poly,polylist,monomialorder);
```

where `poly` is the dividend polynomial, `polylist` is the list of divisors (in square brackets, separated by commas), and `monomial order` is one of the monomial orders discussed before (or any of the others Maple "knows").

Here's a worked example. Suppose we want to divide $f = x^5y + xy^4$ by $f_1 = x^3y + 4y, f_2 = xy^2 - 3$ using the lex order with $x > y$. Enter the following commands one by one (and press ENTER to execute):

```
f := x^5*y + x*y^4;
f1 := x^3*y + 4*y;
f2 := x*y^2 - 3;
NormalForm(f,[f1,f2],plex(x,y));
```

(You could also do this in one line by putting the expressions for the polynomials directly into the `NormalForm` command.) The output is the *remainder on division*. If you want to see the quotients too (that is, the a_i in the expression

$$f = a_1f_1 + a_2f_2 + r)$$

you can include the option `'Q'` in the `NormalForm` command, then print out the value of `Q` afterwards:

```
NormalForm(f,[f1,f2],plex(x,y),'Q');
Q;
```

The second command will show the quotients as a list of polynomials. Try it!

Gröbner bases

Recall from last Friday's class that a *Gröbner basis* for an ideal I with respect to a monomial order $>$ is a finite collection of polynomials $G = \{g_1, \ldots, g_t\}$ with the property that

$$\langle LT_>(I)\rangle = \langle LT_>(g_1), \ldots, LT_>(g_2)\rangle.$$

Equivalently,

For every element $f \in I$, $LT_>(f)$ is divisible by $LT(g_i)$ for some $1 \leq i \leq t$.

Gröbner bases in Maple

The "workhorse" of the `Groebner` package is the `Basis` command, which computes a Gröbner basis for the ideal generated by a list of polynomials with respect to a monomial order. The basic technique for doing this is known as Buchberger's Algorithm. (Technical Notes: We'll see what that means soon. Also, the output will actually be the *unique reduced Gröbner basis*; we'll learn what that means later this week.) The format is

```
Basis(polylist,monomialorder);
```

For example, continuing the computation from before, enter

```
B := Basis([f1,f2],plex(x,y));
```

The output should show

$$B := [4y^6 + 27, 9x + 4y^4]$$

(or something equivalent; Maple sometimes outputs the terms in polynomials in different orders because the way it stores them internally involves some randomness).

As we will see, Gröbner bases for ideals have especially nice properties. In particular when we divide by a Gröbner basis G for I, we have an *ideal membership test*:

$$f \in I \text{ if and only if the remainder on division by } G \text{ is } 0.$$

This is *not true for arbitrary bases of an ideal*(!). For example, using the same f_1, f_2 as above,

$$f = y \cdot f_1 - x^2 \cdot f_2 = 3x^2 + 4y^2 \in I = \langle f_1, f_2 \rangle.$$

However, what happens when we compute the remainder on division of f by f_1, f_2? What happens when we compute the remainder on division by the Gröbner basis `B` computed above?

Problems

(A) Lex Gröbner bases are especially nice for solving systems of polynomial equations, or finding the points in a variety. In this problem, you will consider $\mathbf{V}(x^2y - z^3, 2xy - 4z - 1, z - y^2)$ in $\mathbf{R}^3$.

(1) Compute a *lex* Gröbner basis (say with $x > y > z$) for the ideal $I = \langle x^2y - z^3, 2xy - 4z - 1, z - y^2 \rangle$, and call the Gröbner basis H.

(2) (You'll want to wait until after Wednesday's class (Feb. 20) to answer this one.) Why is it true that setting the Gröbner basis polynomials equal to zero gives a system with the same set of solutions as the original equations

$$\begin{aligned} x^2y - z^3 &= 0 \\ 2xy - 4z + 1 &= 0 \\ z - y^2 &= 0? \end{aligned}$$

(3) Identify a polynomial containing only z in your H; it should be the first polynomial in the list, or `H[1]`. Solve the equation obtained by setting that polynomial equal to zero using the command `rts:=fsolve(H[1],z,complex);` to find approximations to all the complex roots. (In the Maple output here I is the imaginary unit $I = \sqrt{-1}$.) The `fsolve` command uses an approximate *numerical method* to solve equations. You can pick out the individual roots of the z equation with `rts[1]`, `rts[2]`, ... , `rts[7]`.

(4) Then substitute each real root back into the system of equations (use the Maple `subs` command – see the online help for how this works) and solve them for the other variables, y and x. How many real solutions (x, y, z) are there?

(B) Lex Gröbner bases are also especially nice for "implicitizing" parametric equations. For example, consider the surface with this parametrization:

$$\begin{aligned} x &= t + u \\ y &= t^2 + 2tu \\ z &= t^3 + 3t^2u \end{aligned}$$

(This is the union of all the tangent lines to the twisted cubic curve.)

(1) Plot the surface using the `plot3d` command.

(2) Compute a Gröbner basis for the ideal

$$I = \langle x - t - u, y - t^2 - 2tu, z - t^3 - 3t^2u \rangle$$

with respect to a lex order with $t > u > x > y > z$. There should be a polynomial in your basis that depends on x, y, z. This gives an implicit equation of a surface containing the image of the parametrization mapping above. What is this polynomial?

(3) Use the `implicitplot3d` command to plot the surface defined by the polynomial you found in (2). How is it related to the plot from (1)?