Gröbner Basis Reduction

Lecture 14b

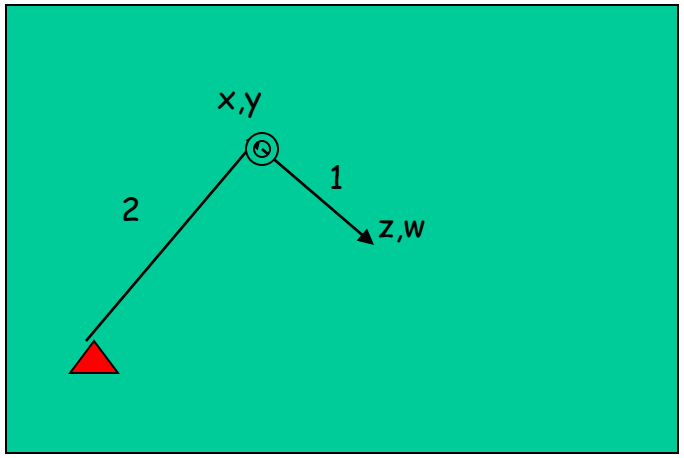
Solving systems of polynomial equations

- Two (or more) polynomials p, q in one variable X, find the common solutions: Compute g=GCD(p,q). Find the roots of g. They are the common roots of p and q.
- Two (or more) LINEAR equations in some set of variables: Compute the solution via Gaussian elimination or similar method.

- Both are special cases of Gröbner/Grobner/Groebner basis calcs.
- [named by Bruno Buchberger for his thesis advisor, (1965); who first "discovered" it is in dispute, but Buchberger clearly popularized it.]

Typical problem. (From von zur Gathen, p 565)

- Set up a robot arm and joint: governed by simple equations. Where can it reach?



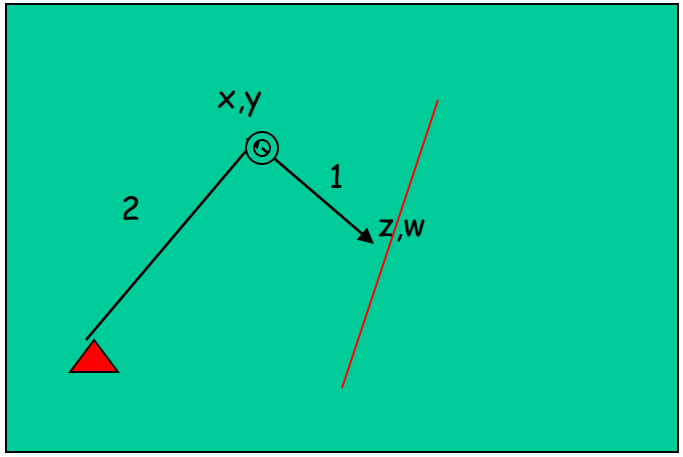
$x^2 + y^2 - 4 = 0$ and

$(x-z)^2+(y-w)^2-1 = 0$

determines a POLYNOMIAL IDEAL

Typical problem. (From von zur Gathen, p 565)

- Set up a robot arm and joint: governed by simple equations. Where can it reach?



Question: can the robot reach a point on the line $u=\lambda v + \mu$

The ideal (polynomial)

- The ideal generated by a family of generators consists of the set of linear combinations of these generators, with constant coefficients. P,Q are equivalent wrt an ideal if their difference belongs to the ideal.

Trivial observations

- There are many systems of generators for each ideal. We can add any linear combination of them to the description.
- We want to find the "simplest."

- To find the simplest, for each polynomial, find its most complex monomial, and try to get rid of it, by simplification using other identities in the idea.

Simplest may not be smallest

- $(s^{20}-1, s^2+c^3-1\} \rightarrow$
- $\{ -10c^3 + 45c^6 - 120c^9 + 210c^{12} - 252c^{15} + 210c^{18} - 120c^{21} + 45c^{24} - 10c^{27} + c^{30}, -1 + c^3 + s^2\}$

- Simpler if *any* polynomial in c is simpler than *any* polynomial in s. If you are counting number of characters, it can grow exponentially.
- (Grobner basis program in Mathematica did this)

Ordering Monomials

- Lexicographic: $a<b<c\ldots<d$
- $a^2 < ab < ac..$
- Same lex, then go by degree $a^2<a^3<a^4$
- In general $x< y$ then $xc < yc$
- Alternative: total degree then lex. E.g. $b^3<a^4$
- Alternative: total degree then reverse lex.

Every non-zero poly P can be written this way

- Principal monomial plus lower order terms:
- $\sum_{i=0} a_i X_i$ with X_i being a product of powers. If we let X_0 be the "largest" then $a_0 X_0$ is the principal monomial
- Then $P = a_0 X_0 + \sum_{i=1} a_i X_i$ and $M_P = a_0 X_0$.
- A polynomial P is reduced with respect to G, a finite set of polys and > a fixed order if no principal monomial of an element of G divides the principal monomial of P.

Can a polynomial P be reduced?

- $P = a_0X_0 + \sum_{i=1} a_iX_i$
- A polynomial P is COMPLETELY reduced with respect to G, a finite set of polys and > a fixed order if no principal monomial of an element of G divides ~~the~~ any ~~principal~~ monomial of P.

- If P is NOT completely reduced, we can subtract from it a multiple of an element of G and eliminate this monomial, get a new polynomial less than the previous one. That is, for some h, $P-hg_i$ is in the same ideal but simpler. (maybe $c*P-h*g_i$)

Does that tell us what to do?

- If P is NOT completely reduced, we can subtract from it a multiple of an element of G and eliminate this monomial, get a new polynomial less than the previous one. That is, for some h, $P - hg_i$ is in the same ideal but simpler. Which element matters for efficiency. Also, there are variations by analogy with the GCD / PRS discussions.

Def. of Standard or Grobner Basis

- A system of generators (or basis) G of an ideal I is called a *standard* basis or Grobner basis (with respect to a given order <) if every reduction of a polynomial P of I to a reduced polynomial (wrt G) always gives 0.

- Dividing P wrt a system of polynomials.. Keep dividing as long as you can (see vzg sec. 21)

The idea behind Buchberger's method

- An arbitrary ideal basis does not, in general constitute a Grobner basis. Buchberger proposed to "complete" the basis by adding a finite number of new polynomials to it. This requires only consideration of a finite number of "S polynomials" of pairs of polynomials from the initial ideal.
- Spoly(p,q)=lcm(M_p,M_q)(p/M_p-q/M_q) where M_q is the principal monomial for polynomial q.

Buchberger's algorithm

- Start with a collection of polynomials *G* representing an ideal.
- Pick 2 different polynomials p,q from *G*.
- Suppose Spoly(p,q) reduces (in *G*) to r. If r=0, then pick another pair until there aren't any more distinct pairs. If r is not 0, add it to *G*, and repeat.
- If you want a unique / reduced GB, reduce each P with respect to *G*-{P}.

Buchberger's algorithm, improvements

- If you want a unique / reduced GB, reduce each P with respect to G-{P}.
- If you want to improve efficiency, you can check various cases ("criterion 1,2") which vastly reduce the time, though it is still exponential.
- Total degree ordering is usually much faster.
- Parallel computation has been explored several times.

Applications of "ideal theory"

- Computing modulo side-relations (e.g. our running example of $\sin^2(x)+\cos^2(x)=1$).
- Geometry theorem proving (Wu, Chou, Kapur, Kutzler and Stifter).
- Large stream of publications on variations.

--additional material--