1 Monomial Orders.

In the polynomial algebra over F a field in one variable $x, F[x]$, we can do long division (sometimes incorrectly called the Euclidean algorithm). If $f(x) = a_0 + a_1x + ... + a_nx^n \in F[x]$ with $a_n \neq 0$ then we write $\deg f(x) = n$ and $LC(f(x)) = a_m$. If $g(x)$ is another element of $F[x]$ then we have

$$f(x) = h(x)g(x) + r(x)$$

with $h(x), r(x) \in F[x]$ and $\deg r[x] < m$. This expression is unique. This result can be varified by long division. As with all divisions we assume $g(x) \neq 0$. Which we will recall as a pseudo code (such code terminates on a return) the input being f, g and the output h and r:

$f_0 = f; h_0 = 0; k = 0; m = \deg g;$

Repeat:

If $\deg(f_k(x) < \deg(g(x)$ return $h_k(x), f_k(x); n = \deg f_k;$

$f_{k+1}(x) = f_k(x) - \frac{LC(f_k(x))}{LC(g(x))}x^{n-m}g(x)$

$h_{k+1}(x) = h_k(x) + \frac{LC(f_k(x))}{LC(g(x))}x^{n-m};$

$k = k + 1;$

Continue;

We note that this code terminates since at each step when there is no return then the new value of the degree of $f_x(x)$ has strictly decreased.

The theory of Gröbner bases is based on a generalization of this algorithm to more than one variable. Unfortunately there is an immediate difficulty. The degree of a polynomial does not determine the highest degree part of the polynomial up to scalar multiple. But this is precisely what is needed. To fix this problem one introduces what is known as a monomial order.

We say that a total order (i.e satisfies transitivity and tricotomy), $>$, on $\mathbb{Z}^n_{\geq 0}$ is a *monomial order* if it satisfies the following conditions (Greek letters will designate elements of $\mathbb{Z}^n_{\geq 0}$):

1. $\lambda \neq (0, ..., 0)$ implies that $\lambda > 0$.
2. If $\lambda > \mu$ then $\lambda + \xi > \mu + \xi$.
3. Any non-empty subset of $\mathbb{Z}^n_{\geq 0}$ has a minimal element.

Exercise 1. Show that the only possible order on $\mathbb{Z}_{\geq 0}$ satsifying 1. and 2. is the usual order.

Exercise 2. Show that condition 3 is equivalent with the condition: If $\alpha_1 \geq \alpha_2 \geq \alpha_3 \geq \cdots \geq \alpha_m \geq \cdots$ then there exists M such that $\alpha_k = \alpha_M$ for $k \geq M$.

We will now explain two of the main ways of getting monomial orders on $\mathbb{Z}^{p+q}_{\geq 0}$ if we have monomial orders $>_1$ on $\mathbb{Z}^p_{\geq 0}$ and $>_2$ on $\mathbb{Z}^q_{\geq 0}$.

I. Write $\lambda = (\xi, \nu), \mu = (\delta, \sigma)$ with $\xi, \delta \in \mathbb{Z}^p_{\geq 0}$ and $\nu, \sigma \in \mathbb{Z}^q_{\geq 0}$ then the new order is given as follows: $\lambda > \mu$ if $\xi >_1 \delta$ or if $\xi = \delta$ and $\nu >_2 \sigma$.

If $\lambda \neq 0$ then if $\lambda = (\xi, \nu)$ if $\xi \neq 0$ then $\xi >_1 0$ so $\lambda > 0$ if $\xi = 0$ then if $\lambda \neq 0$ then $\nu \neq 0$ hence $\nu >_2 0$ hence $\lambda > 0$. If $\lambda > \mu = (\delta, \sigma)$ then writing $\alpha = (\rho, \tau)$ we see that if $\xi >_1 \delta$ then $\xi + \rho >_1 \delta + \rho$ if $\xi = \delta$ then $\xi + \rho = \delta + \rho$ and $\nu >_2 \tau$ which implies $\nu + \tau >_2 \sigma + \tau$. Thus condition 2 is satsified. Condition 3 is

satsified since in such a sequence the first components are in decreasing order thus must stabilize. Once they stabilize the second omponents are in decreasing order.

II. Here there are three levels of tests. Let $\lambda = (\xi, \nu)$ and $\mu = (\delta, \sigma)$. If $|\lambda| > |\mu|$ then $\lambda > \mu$ if $|\lambda| = |\mu|$ and $\nu <_2 \sigma$ then $\lambda > \mu$ (notice the reversal here) If $|\lambda| = |\mu|$ and $\nu = \sigma$ and if $\xi >_2 \delta$ then $\lambda > \mu$.

We leave it to the reader to check that this defines a monomial order.

If we start with the only possible order on $\mathbb{Z}_{\geq 0}$ then use method I on $\mathbb{Z}^2_{\geq 0}$ then method I. on $\mathbb{Z}^3_{\geq 0} = \mathbb{Z}_{\geq 0} \times \mathbb{Z}^2_{\geq 0}$ and continue in this way to $\mathbb{Z}^n_{\geq 0}$ then we get Lexicographic order or Lex order. That is, if $\lambda - \mu = (0, ..., 0, u, ...)$ and $u > 0$ then $\lambda > \mu$. We write $>_{Lex}$for this order.

The other standard order is Graded Reverse Lex or GrRevLex. Which is gotten by an iteration of method II. Here the first test is on the norm. Next we look at the last entry. If the last entry of λ is strictly less than that of μ then $\lambda > \mu$ otherwise if one has equality and $n > 2$ we use the same test on the second to last entry (if $n = 2$ and then at this stage we know $\lambda = \mu$). If one has equality in the next to last index and $n > 3$ then one goes to the next index up and uses the same test, etc. We write $>_{GrRLex}$ for this order.

Exercise 3. If $>$ is a total order on $\mathbb{Z}^n_{\geq 0}$ that satisfies 2. in the definition of monomial order. Then we introduce a new version of the order by saying that if $|\lambda| > |\mu|$ then $\lambda >_{Gr} \mu$ and if $|\lambda| = |\mu|$ and if $\lambda > \mu$ then $\lambda >_{Gr} \mu$. Show that $>_{Gr}$ defines a monomial order.

The order in exercise 3 is called the graded version of $>$. We denote by GrLex the graded version of Lex and denote the order by $>_{GrLex}$.

Exercise 4. Show that if $n = 2$ then GrLex is the same as GrRevLex.

If x^λ and x^μ are monomials and if $>$ is a monomial order we will write $x^\lambda > x^\mu$ if $\lambda > \mu$.

Fix a monomial order, $>$. (We will use it throught the rest of this section). If $f(x_1, ..., x_n) \neq 0$ is a polynomial we can write

$$f(x_1, ..., x_n) = a_\lambda x^\lambda + \sum_{\mu < \lambda} a_\mu x^\mu.$$

We will call x^λ the leading monomial of $f(x_1, ..., x_n)$ and denote it by $LM(f(x_1, ..., x_n))$ (we will also write $\lambda = LM(f(x_1, ..., x_n))$. We will call a_λ the leading coefficient of $f(x_1, ..., x_n)$ and denote it $LC(f(x_1, ..., x_n)$.

If $\lambda, \mu \in \mathbb{Z}^n_{\geq 0}$ then we say that λ dominates μ if $\lambda - \mu \in \mathbb{Z}^n_{\geq 0}$. With all of this in place we can set up the multivariate *division algorithm*. Here $f(x_1, ..., x_n), g(x_1, ..., x_n) \in F[x_1, ..., x_n]$.

$h_0(x_1, ..., x_n) = 0; f_0(x_1, ..., x_n) = f(x_1, ..., x_n); \mu = LM(g(x)); k = 0;$

Repeat:

If $LM(f_k(x))$ does not dominate μ return $h_k(x_1, ..., x_n), f_k(x_1, ..., x_n)$;

$f_{k+1}(x_1, ..., x_n) = f_k(x_1, ..., x_n) - \frac{LC(f_k(x_1,...,x_n))}{LC(g(x_1,...,x_n))} x^{LM(f_k(x_1,...,x_n)-\mu} g(x)$

$h_{k+1}(x) = h_k(x) + \frac{LC(f_k(x_1,...,x_n))}{LC(g(x_1,...,x_n))} x^{LM(f_k(x_1,...,x_n)-\mu}$;

$k = k + 1$;Continue;

The output of this algorithm is a pair of polynomials $h(x_1, ..., x_n)$ and $r(x_1, ..., x_n)$ with

$$f(x_1, ..., x_n) = h(x_1, ..., x_n)g(x_1, ..., x_n) + r(x_1, ..., x_n)$$

and the leading monomial of $r(x_1, ..., x_n)$ does not dominate that of $g(x_1, ..., x_n)$. We will call $r(x_1, ..., x_n)$ the reduction of $f(x_1, ..., x_n)$ relative to $g(x_1, ..., x_n)$ relative to the given monomial order. We will denote r be remainder(f, g).

Exercises 5. Calculate h and r for $f = x^4y^4 + x^2y^6$ and $g = x^3y + x^2y^3$ using Lex and GrLex.

6. Show that if $f = ug$ then the division algorthm outputs $h = u$ and $r = 0$.

If we have a set of nonzero polynomials $f_1, ..., f_m$ than will now describe a procedure analogous to Gaussian elimination to put them in a convenient "normal form". In which we relace the polynomials by polynomials $h_1, ..., h_l$ generating the same ideal with the following properties:

1. If $i \neq j$ then $LM(h_i)$ does not domiante $LM(h_j)$.
2. $LM(h_1) < LM(h_2) < ... < LM(h_l)$.

We now give the *normalize algorithm*. Here we partially order polynomials by their leading monomials using the monomial order that we have fixed. The statement Break means that if one is doing operations $1, 2, ...$ and the Break is at i then one stops doing that operation immediately (it takes the ith step as the last in the loop).

For $i = 1, ..., m$ $f_{i,0} = f_i$; $k = m; l = 0$;

Repeat:

Remove all of the $f_{i,l}$ that are 0. Relabel the rest so that they are in the same order as before. Subtract the number of zeros deleted from k.

Order the $f_{1,l}, ..., f_{k,l}$ in the first subindex (remember the partial order is according to the order on the lead monomials).

$TST = (-1, -1)$;

For $i = 1, i \leq k - 1$,If $TST \neq (-1, -1)$ Break; For $j = i + 1, j \leq k$ if $f_{j,l}$ dominates $f_{i,l}$ then $TST = (i, j) = (TST_1, TST_2)$;Break;

If $TST = (-1, -1)$ then Return $f_{1,l}, ..., f_{k,l}$;

For $s = 1, s < TST_2, f_{s,l+1} = f_{s,l}; f_{TST_2,l+1} =$ remainder$(f_{TST_2,l}, f_{TST_1,l})$; For $s = TST_2 + 1, s \leq k, f_{s,l+1} = f_{s,l}$;

$l = l + 1$;

Continue.

We must check that this algorithm terminates. Suppose not then at each stage there must be a pair of indices such that a nontrivial division has taken place. That is at each stage l there is some i such that $LM(f_{i,l}) > LM(f_{i,l+1})$. Since there are only m possible values of i there must be an infinite sequence $l_1 < l_2, ...$ such that $LM(f_{i,l_j}) > LM(f_{i.l_{j+1}})$. This contradicts 3. in the definition of monomial order.

We will now give the *reduction algorithm* that will be important to our development of Gröbner bases. Let $G = \{g_1, ..., g_m\}$ be a set of non-zero poly-

nomialsin a given order and let f be a polynomial. Here is the algorithm (see the previous algorithm for the meaning of Break).

$f_0 = f$; $a_1 = a_2 = ... = a_m = 0; k = 0$;

Repeat:

$TST = -1$;

For $i = 1$ to m if $LM(f_k)$ dominates $LM(g_i)$ then $TST = i$ and Break;

If $TST == -1$ return $(a_1, ..., a_m)$, f_k;

Do division algorithm on f_k and g_{TST} with output b, f_{k+1};

$a_i = a_i + b$;

$k = k + 1$;

Continue;

The output of this algorithm is $(a_1, ..., a_m), r$. With $r, a_i \in F[x_1, ..., x_n]$, $f = a_1g_1 + ... + a_mg_m + r$ and $LM(r)$ does not dominate any of the $LM(g_i)$. The polynomial r will be called the reduction of f with respect to the ordered set $g_1, ..., g_m$.

If $LM(g_i) \neq LM(g_j)$ for $i \neq j$ and $LM(g_i) < LM(g_{i+1})$ for $i = 1, ..., m-1$ then we will use the notation $red_G(f)$ for the output r of the above algorithm.

Exercise 7. Show that this algorithm terminates.

Exercise 8. Suppose that in the above algorithm we used a different ordering of the elements of G would we get the same r?

2 Gröbner bases.

If I is an ideal in $F[x_1, ..., x_n]$ and if $>$ is a monomial order then we define $LM(I)$ to be the ideal generated by $\{LM(f) | f \neq 0, f \in I\}$. We call $LM(I)$ the monomial ideal associated with I and the order $>$.

If I is an ideal and if $>$ is a monomial order then we set $S(I)$ equal to the set of monomials that do not dominate any element of $\{LM(f) | f \neq 0, f \in I\}$.

Lemma 1 *$F[x_1, ..., x_n] = I \bigoplus Span_F S(I)$. Furthermore,$S(I)$ is a linearly independent set.*

Proof. The second assertion is obvious. That the sum in the asertion is direct is also obvious (we leave this as an exercise with the hint that it is obvious). Let A be the set of all $f(x_1, ..., x_n) \notin I \bigoplus Span_F S(I)$. We assume that A is nonempty. Then the set $\{LM(f) | f \in A\}$ has a minimal element. Let $f \in A$ be such that $LM(f)$ is that element. We now assume that $LC(f) = 1$. If $LM(f) \notin S(I)$ then $LM(f)$ dominates $LM(g)$ for some $g \in I$. We may assume that $LC(g) = 1$. Then $LM(f - \frac{LM(f)}{LM(g)}g) < LM(f)$ and $f - \frac{LM(f)}{LM(g)}g \in A$. This is a contradiction. If $LM(f) \in S(I)$ then $LM(f - LM(f)) < LM(f)$ and we run into the same contradiction. ■

Corollary 2 *The dimension of the variety defined by $LM(I)$ is the same as that defined by I.*

Proof. The value of the Hilbert function, $h(k)$, of $F[x_1, ..., x_n]/I$ equal to the number of elements, λ, of $S(I)$ with $|\lambda| \leq k$. This is the same value as for $F[x_1, ..., x_n]/LM(I)$. ■

One can see that the monomial ideal of I relative to the monomial order $>$ is an important computational tool. Our problem is to find a way to calculate it. We will first introduce another concept that is also apparently non-algorithmic and will (following Buchberger) find an algorithm to calculate it.

Lemma 3 *Let $>$ be a momomial order and let I be an ideal in $F[x_1, ..., x_n]$. Let $\{g_1, ..., g_m\}$ be a set elements of I. Then the following are equivalent:*

1. The set $\{LM(g_1), ..., LM(g_m)\}$ generates $LM(I)$.

2.Order the g_i arbitrarily. If f is a polynomial then $f \in I$ if and only if the reduction of f with respect to $g_1, ..., g_m$ is 0*.*

We will call a set of generators for I that satisfies the conditions of the lemma for $>$ a Gröbner basis for I with respect to $>$. We note that 1. implies that Gröbner bases always exist since we can apply the Hilbert basis theorem to $LM(I)$ to get a finite generating set $x^{\mu_1}, ..., x^{\mu_m}$ for LM and choose $g_1, ..., g_m$ in I so that $LM(g_i) = x^{\mu_i}$. We also note that 2. implies that a Gröbner basis is a generating set of I.

We will now prove the result.

Assume that $\{g_1, ..., g_m\}$ satisfies 1. Then if $f \in I$ we have $LM(f)$ must dominate some element of $\{LM(g_1), ..., LM(g_m)$. Let $LM(g_i)$ be the one with smallest index. If we apply the division algorithm to f and g_i the remainder has a lead monomial strictly less than that of f. So let A be the set of all f such that 2. is not satisfied. Let f be an element of A with $LM(f)$ minimal. Now use the procedure just discribed derive a contradiction. Clearly if f reduces to 0 by repeated applications of the division algorithm then $f \in I$.

We will now show that 2. implies 1. We must only show that if $f \in I$, $f \neq 0$ then $LM(f)$ dominates $LM(g_i)$ for some i. But this is obvious since it is necessary in order to begin the reduction of f to 0 via the division algorithm.

As an immediate corollary we have:

Corollary 4 *Let $G = \{g_1, ..., g_m\}$ be such that $LM(g_i \neq LM(g_j)$ for $i \neq j$. Then G is a Gröbner basis for $\langle G \rangle$ if and only if $f \in F[x_1, ..., x_n]$ is in $\langle G \rangle$ if and only if $red_G(f) = 0$.*

Notice that if we have a Gröbner basis of I with respect to $>$ then we can test whether $f \in I$ using the reduction algorthm. In fact, the algorithm shows us how to write f in terms of the g_i. However much more is true.

We say that a Gröbner basis, $\{g_1, ..., g_m\}$ for I relative to $>$ is reduced if no monomial of g_i dominates $LM(g_j)$ for $j \neq i$.

Lemma 5 *Given a monomial order $>$ each ideal has a unique reduced Gröbner basis with leading all coeffiecients* 1*..*

Proof. Let $\{g_1, ..., g_m\}$ be a Gröbner basis for I with respect to the monomial order $>$. We may assume that all of the leading coefficients of the g_i are 1. We note that if $LM(g_i)$ dominates $LM(g_j)$ for $j \neq i$ then $\{g_1, ..., g_m\} - \{g_i\}$ is still a Gröbner basis. We can thus assume that no $LM(g_i)$ dominates $LM(g_j)$ for $i \neq j$. We can now assume that this property is satisfied and that $LM(g_1) < LM(g_2) < ... < LM(g_m)$. We note that if x^λ is a monomial with non-zero coefficient in g_i then $\lambda \leq LM(g_i)$thus λ cannot dominate $LM(g_j)$ for $j > i$. Thus no monomial of g_1 dominates $LM(g_i)$ for $i > 1$. We now consider the non-leading monomials of g_2. Let

$$g_i = \sum_\lambda a_{\lambda,i} x^\lambda$$

Let $\Sigma_2 = \{\lambda | a_{\lambda,2} \neq 0, \lambda \text{ dominates } LM(g_1)\}$ replace g_2 by

$$g_2 - \sum_{\lambda \in \Sigma_2} a_{\lambda,2} x^{\lambda - LM(g_1)} g.$$

Now no monomial of g_2 dominates the lead monomial of g_i for $i \neq 2$. We now consider g_3 we first do exactly the above procedure to g_3 relative to g_2 so that the new g_3 has no monomial that dominates $LM(g_2)$. Now do the same with g_3 and g_1. We next do the same thing with g_4 relative to $g_i, i = 1, 2, 3$. When we get to m we have a reduced Gröbner basis. This shows that from a Gröbner basis we can always algorithmically construct a reduced Gröbner basis.

Let $\{g_1, ..., g_m\}$ and $\{h_1, ..., h_l\}$ be reduced Gröbner bases of I with respect to $>$ with leading coeffiecients 1.. We first assert that $\{LM(g_1), ..., LM(g_m)\} = \{LM(h_1), ..., LM(h_l)\}$. Indeed $LM(g_i)$ must dominate $LM(h_j)$ for some j and $LM(h_j)$ must dominate $LM(g_k)$ for some k. Thus $LM(g_i)$ dominates $LM(g_k)$ so we must have $k = i$. This implies $LM(g_i) = LM(h_j)$. This proves the assertion. Thus $m = l$ and we can reorder so that $LM(g_i) = LM(h_i)$. We note that if $g_i - h_i \neq 0$ then $LM(g_i - h_i) < LM(g_i)$. But $LM(g_i - h_i)$ must dominate some $LM(g_j)$ this implies that either g_i or h_i as a non-leading monomial that domiates one of the $LM(g_j)$. This contradicts the definition of reduced. ■

This result says that if we can effectively construct Gröbner bases then we have an effective test for whether two ideals are equal.

Exercises 1. Show that the procedure described in the first part of the proof does produce a reduced Gröbner basis.

2. Write pseudocode to convert a Gröbner basis to a reduced Gröbner basis.

3 Syzygies.

In this section we will give a test (due to Buchberger) as to when a generating set for an ideal is a Gröbner basis relative to a monomial order. Through this section we keep a fixed monomial order $>$. If $f, g \neq 0$ are polynomials then let x^λ be the least common multiple of $LM(f)$ and $LM(g)$. We note that if

$LM(f) = x^\mu$ and $LM(g) = x^\nu$ then $\lambda = (\lambda_1, ..., \lambda_n)$ with $\lambda_i = \max(\mu_i, \nu_i)$. If $\xi = \lambda - \mu$ and $\delta = \lambda - \nu$ then we set

$$S(f, g) = LC(g)x^\xi f - LC(f)x^\delta g.$$

Buchberger's criterion is:

Theorem 6 *Let I be an ideal. Let $G = \{g_1, ..., g_m\}$ be a generating set Then G is a Gröbner basis of I if and only if for all i, j we have $S(g_i, g_j) = \sum_k a_{ijk}(x_1, ..., x_n)g_k$ and $LM(a_{ijk}g_k) \leq LM(S(g_i, g_j))$.*

This condition is necessary since very element of I reduces to 0 relative to a Gröbner basis and the reduction process yields an expression with the above property. In actual practice one mainly uses the following "weaker form" of the criterion.

Corollary 7 *Let I be an ideal. Let $G = \{g_1, ..., g_m\}$ be a generating set such that $LM(g_i) \neq LM(g_j)$ for $i \neq j$.Then G is a Gröbner basis of I if and only if for all i, j we have $red_G(S(g_i, g_j)) = 0$.*

Exercises:

1. Let I be the ideal in $F[x_1, ..., x_{2n}]$ generated by $\{x_i x_{n+j} - x_j x_{n+i} | i < j\}$. Show that this is a Gröbner basis relative to Lex.

2. In this exercise we will also use Lex for our order. Let I be the ideal generated by $e_1(x_1, ..., x_n), e_2(x_1, ..., x_n), ..., e_n(x_1, ..., x_n)$ with $e_m(x_1, ..., x_n) = \sum_{1 \leq i_1 < \cdots < i_m \leq n} x_{i_1} x_{i_2} \cdots x_{i_m}$, ...Use the identity

$$0 = \prod_{j=1}^{m} (x_m - x_j) = \sum_{j=0}^{m} (-1)^{m-j} e_{m-j}(x_1, ..., x_m) x_m^j$$

to show that there is a generating set $h_1 = e_1, h_2, ..., h_n = x_n^n$ for I with $LM(h_j) = x_j^j$. (Hint: The displayed formula says that

$$x_m^m = -\sum_{j=0}^{m-1} (-1)^{m-j} e_{m-j}(x_1, ..., x_m) x_m^j.$$

Use $e_j(x_1, ..., x_{m+1}) = e_j(x_1, ..., x_m) + e_{j-1}(x_1, ..., x_m)x_{m+1}$ to show that the leading term of

$$-\sum_{j=0}^{m-1} (-1)^{m-j} e_{m-j}(x_1, ..., x_n) x_m^j$$

is x_m^m.) Prove that $\{h_1, ..., h_n\}$ is a Gröbner basis for I with respect to Lex. For this you may want to use the next exercise.

3. Let $0 \neq f, g \in F[x_1, ..., x_n]$ be such that $LM(f)$ and $LM(g)$ are relatively prime. Then $S(f, g) = af + bg$ with $LM(f, g) \leq LM(af)$ and $LM(f, g) \leq LM(bg)$. (Hint: Assume $LC(f) = LC(g) = 1$. Then $f = LM(f) + u$ and

$g = LM(g) + v$. Thus $S(f,g) = LM(g)f - LM(f)g = vf - ug$. Show that the leading terms of vf and ug are distinct.)

We will now prove the Buchberger criterion.

We assume that each $S(g_i, g_j) = \sum a_{ijk} g_k$ with $LM(a_{ijk} g_k) \leq S(g_i, g_j)$. Let $f \in I$ be nonzero. Then we must show that $LM(f)$ dominates some $LM(g_i)$. For this it would be enough to prove that $f = \sum b_l g_l$ with $LM(b_l g_l) \leq LM(f)$ fpr all l. Since then we must have some $LM(b_i g_i) = LM(f)$ so $LM(f)$ dominates $LM(g_i)$.

We procede to prove this by contradiction. Let among all expressions $f = \sum b_l g_l$, $f = \sum c_i g_i$ be the one with $\max\{LM(c_i g_i) | i = 1, ..., m\} = \lambda$, minimal. We note that $\lambda \geq LM(f)$. We assume that $\lambda > LM(f)$ and derive a contradiction. If we relabel we may assume that $LM(c_i g_i) = \lambda$ for $i \leq k$ and $LM(c_i g_i) < \lambda$ for $i > k$. We write $LM(g_i) = x^{\delta_i}$ and we assume that $LC(g_i) = 1$. We write $LM(c_i) = x^{\xi_i}$ and $LC(c_i) = u_i$. Then $\xi_i + \delta_i = \lambda$ for $i \leq k$. We have $f = \sum_{i \leq k} u_i x^{\xi_i} g_i + \sum_{i \leq k} (c_i - u_i x^{\xi_i}) g_i + \sum_{i > k} c_i g_i$. We have $LM(c_i - u_i x^{\xi_i} g_i) < \lambda$ for $i \leq k$ and $LM(c_i g_i) < \lambda$ for $i > k$. We look at the first term. We note that the coefficient of x^λ in that term is $\sum_{i \leq k} u_i$. This must be the coefficient of x^λ in f which is 0. Thus $\sum_{i \leq k} u_i = 0$. We note that if $(z_1, ..., z_k) \in F^k$ and $\sum_{i \leq k} z_i = 0$ then $(z_1, ..., z_k) = \sum_{i=1}^{k-1} w_i (e_i - e_{i+1})$ with e_i the vector with a 1 in the i-th position and 0's everywhere else. This says that there are elements of F, d_i, such that

$$\sum_{i \leq k} u_i x^{\xi_i} g_i = \sum_{i \leq k-1} d_i z (x^{\xi_i} g_i - x^{\xi_{i+1}} g_{i+1}). \qquad (*)$$

Since $\xi_i + \delta_i = \lambda$ we see that $x^{\xi_i} g_i - x^{\xi_{i+1}} g_{i+1} = x^{\mu_i} S(g_i, g_{i+1})$. for an appropriate μ_i We also note that $LM(S(g_i, g_{i+1})) < LM(LCM(g_i, g_{i+1}))$. Our hypothesis implies that $S(g_i, g_j) = \sum a_{ijk} g_k$ with $LM(a_{ijk} g_k) \leq LM(S(g_i, g_j))$. Thus we have

$$\sum_{i \leq k} u_i x^{\xi_i} g_i = \sum d_i a_{ii+1k} x^{\mu_i} g_k$$

with $LM(a_{ii+1k} g_k) \leq LM(S(g_i, g_{i+1})) < \lambda - \mu_i$. Now putting all of these terms together we have an expression

$$f = \sum h_i g_i$$

with $LM(h_i g_i) < \lambda$ for all i. This is the desired contradiction.

Exercise. Carry out the details of the proof of (*) above.

4 Buchberger's algorithm.

We will now give an algorithm that allows the effective computation of Gröbner bases on a computer. We start with a fixed monomial order and a generating set for an ideal $\{f_1, ..., f_m\}$ in $F[x_1, ..., x_n]$. As an output we will get a

Gröbner basis. Polynomials are partially ordered via their leading terms. After normalization a finite set of polynomials is totally ordered.

Here is the pseudo code:

PresentBasis = Normalze($f_1, ..., f_m$); Queue = $\{S(f,g)|f,g \in$ Presentbasis, $S(f,g) \neq 0\}$.

Repeat:

If Queue = $\emptyset$ return PresentBasis;

u a minimal element of the Queue;

$v = red_{\text{PresentBasis}}(u)$; Queue = Queue $-\{u\}$;

If $v = 0$; Continue;

Queue = Queue $\cup\{S(f,v)|f \in$ PresentBasis$\}$;

PresentBasis=PresentBasis$\cup\{v\}$;

Continue;

We must show that this pseudocode terminates. So assume that it doesn't terminate. Then the algorithm yields an infinite set $\{v_1, v_2, ...\}$ such that $LM(v_i)$ doesn't dominate $LM(v_j)$ for $i \neq j$. Consider $I_j = \langle LM(v_1), ..., LM(v_j)\rangle$ assume that $I_{j+1} = I_j$ then $LM(v_{j+1}) = \sum_{i \leq j} h_i LM(v_i)$. This implies that $LM(v_{j+1})$ must be a monomial in one of the $h_i LM(v_i), i \leq j$. But then we would have $LM(v_{j+1})$ dominates $LM(v_i)$. This is contrary to the outcome of the (assumed to be) failed pseudocode. Thus the ideals I_j must never stabilize and this contradicts the Hilbert basis theorem.

In practice there are several tricks due to Buchberger to speed up the algorithm (one is related to Exercise 3 in section 3). There is another that is related to a generaliziation of the algorithm to submodules of free modules. These improvements are only about 20%. There are no known exponential speadups. These algorithms properly souped up are the basis of computational algebra packages such as Mathematica, Maple, Macaulay, Singular,...

We will give algorithms based on the (assumed) calculation of a Gröbner basis in Mathematica code.

5 Hilbert Series

In this section we will fix a monomial order, $>$. We consider a homogeneous ideal $I \subset F[x_1, ..., x_n]$. Then $A = F[x_1, ..., x_n]/I$ is graded by degree. Here the m-th part of the grade is, A^m, the image in A of the polynomials homogeneous of degree m. We define the formal power series

$$H(A,q) = \sum_{m=0}^{\infty} \dim A^m q^m.$$

$H(A)$ is called the Hilbert series of the graded algebra A. If $I = \sqrt{I}$ and $X = \mathbb{P}^{n-1}(I)$ then we will give and interpretation of the numbers $\dim A^m$ later

in terms of sheal cohomology. In this section we will give algorithmic methods to calculate it. We will also use the algorithm to prove some properties of this series.

Let $LM(I)$ be the monomial ideal associated to I corresponding to the chosen order. Let S be a finite set of monomial generators for $LM(I)$. We can find S by calculating a Gröbner basis of I. We will now assume that that has been done and we have S. We look upon S as a subset of $\mathbb{Z}^n_{\geq 0}$. If $\lambda, \mu \in \mathbb{Z}^n_{\geq 0}$ we write $\lambda \succ \mu$ if $\lambda - \mu \in \mathbb{Z}^n_{\geq 0}$ (that is if λ dominates μ). We write $\lambda \not\succ \mu$ if λ doesn't dominate μ. If $T \subset \mathbb{Z}^n_{\geq 0}$ we write $\lambda \not\succ T$ if $\lambda \not\succ \mu$ for all $\mu \in T$. With this notation in place we can begin our approach to calculation Hilbert series

Set for $T \subset \mathbb{Z}^n_{\geq 0}$, $B(T) = \{\lambda \in \mathbb{Z}^n_{\geq 0} | \lambda \not\succ T\}$. We set

$$Hilb(T, n) = \sum_{\lambda \in B(T)} q^{|\lambda|}.$$

From the material in section 2 we have

Lemma 8 $H(A, q) = Hilb(S, n)$.

This says that we need only give an algorithm for the calculation of $Hilb(T, n)$ for T a finite subset of $\mathbb{Z}^n_{\geq 0}$. First if $T = \emptyset$ then $B(T) = \mathbb{Z}^n_{\geq 0}$ so

$$Hilb(\emptyset, n) = \sum_{\lambda \in \mathbb{Z}^n_{\geq 0}} q^{|\lambda|} = \frac{1}{(1-q)^n},$$

Next we not that if $0 \in T$ then $B(T) = \emptyset$ so $Hilb(T, n) = 0$.

If $n = 1$ and $T \neq \emptyset$ then $T = \{(a_1), ..., (a_m)\}$ if $r = \min\{a_1, ..., a_m\}$ then $B(T) = B(\{r\})$. Hence

$$Hilb(T, 1) = \sum_{j<r} q^j = 1 + q + ... + q^{r-1}.$$

Now assume that $n > 1$ and $T \neq \emptyset$. Then we set $m = \max\{\lambda_1 | \lambda = (\lambda_1, ..., \lambda_n) \in T\}$. We also set for $\lambda \in \mathbb{Z}^n_{\geq 0}$, $\lambda' = (\lambda_2, ..., \lambda_n)$ and if $U \subset \mathbb{Z}^n_{\geq 0}$ we set $U' = \{\lambda' | \lambda \in U\}$. We note that

I. If $\lambda \in \mathbb{Z}^n_{\geq 0}$ and $\lambda_1 \geq m$ then $\lambda \in B(T)$ if and only if $\lambda' \in B(T')$.

If $0 \leq i \leq m-1$ we set $T(i) = \{\lambda \in T | \lambda_1 \leq i\}$. We note that

II. If $\mu \in \mathbb{Z}^{n-1}_{\geq 0}$ and $0 \leq i \leq m-1$ then $(i, \mu) \in B(T)$ if and only if $\mu \in B(T(i)')$.

We will prove the second and leave the first as an exercise.

Suppose that $(i, \mu) \in B(T)$ and $\mu \notin B(T(i)')$. Then there is $\xi \in T(i)'$ with $\mu \succ \xi$. Now there exists $j \leq i$ such that $(j, \xi) \in T$ and since $(i, \mu) \succ (j, \xi)$ we have a contradiction. Thus if $(i, \mu) \in B(T)$ then $\mu \in B(T(i)')$. If $\mu \in B(T(i)')$ and $(i, \mu) \notin B(T)$ then there is a $\nu \in T$ with $(i, \mu) \succ \nu$. But then $\nu_1 \leq i$ thus $\nu' \in T(i)$ and we have a contradiction. This completes the proof of II.

This leads directly to the following (recursive) algorithm.

Pseudocode to calculate $Hilb(T,n)$ given a finite subset, T, in $\mathbb{Z}^n_{\geq 0}$. (Here we assume $n \geq 1$.)

If $T = \emptyset$ return

$$\frac{1}{(1-q^n)}.$$

If $0 \in T$ then return 0;

If $n = 1$ then

$r = \min\{\lambda_1 | \lambda \in T\}$;

return $1 + q + ... + q^{r-1}$;

$m = \max\{\lambda_1 | \lambda \in T\}$.

return

$$\sum_{i=0}^{m-1} q^i Hilb(T(i)', n-1) + \frac{q^m}{1-q} Hilb(T', n-1).$$

Example. If we start with the example in section 3 where $I = \langle e_1, ..., e_n \rangle$ in $F[x_1, ..., x_n]$. Then we saw that the monomial ideal corresponding to Lex is generated by

$$T = \{(1,0,0,...,0), (0,2,0,...,0), (0,0,3,...,0), ..., (0,0,....,0,n)\}.$$

We will denote this as T_n. It is convenient to use the algorithm above starting with the last variable rather than the first. Thus the prime notation will mean drop the last coordinate and the $T(i)$ notation will mean that the last coordinate is at most i. Assume $n > 1$. Then we have in the algorithm $m = n$. If $i \leq n-1$ then

$$T(i)' = \{(1,0,0,...,0), (0,2,0,...,0), (0,0,3,...,0), ..., (0,0,....,0,n-1)\}$$

whereas $0 \in T'$. So if $n > 1$ the algorithm returns

$$\sum_{i=0}^{n-1} q^i Hilb(T(i)', n-1) = (1 + q + ... + q^{n-1}) Hilb(T_{n-1}, n-1).$$

We follow the recursion and get (since $Hilb(\{(1), 1)) = 1$).

$$Hilb(T_1, 1) \prod_{j=1}^{n-1} (1 + q + ... + q^j) = \prod_{j=1}^{n-1} (1 + q + ... + q^j).$$

We can rewrite this as

$$\prod_{i=1}^{n} \frac{1-q^i}{1-q}.$$

Exercise 1. Show that the Hilbert series corresponding to $F[x_1, ..., x_{2n}]/I$ as in Exercise 1 in section 3 is

$$\frac{1+nq}{(1-q)^{n+1}}.$$

We note that the algorithm implies that if $T \subset \mathbb{Z}^n_{\geq 0}$ then

$$Hilb(T,n) = \frac{p(q)}{(1-q)^n}$$

with $p(q)$ a polynomial in q with integral coefficients. In the obove example $p(q) = \prod_{i=1}^{n}(1-q^n)$.

We also note that if $p(q) = \sum_{i=0}^{d} p_i q^i$ then $p(q)$ is divisible by $(1-q)$ if and only if $p(1) = \sum_{i=0}^{d} p_i = 0$. If that is so and we set $a_j = p_0 + p_1 + ... + p_j$ then $a_j = 0$ for $j > d-1$ and $p(q) = (1-q)\sum a_j q^j = (1-q)h(q)$. If $h(1) = 0$ we can do this procedure again. We therefore see that it is a simple computational procedure to write

$$p(q) = (1-q)^r P(q)$$

with $P(1) \neq 0$.

Proposition 9 *Assume that $H(A,q)$ has been writteh in the form*

$$\frac{P(q)}{(1-q)^d}$$

with $P(1) \neq 0$. Then $d = \dim \mathbb{A}^n(I)$. In other words, the dimension of the projective variety in $\mathbb{P}^{n-1}$ that is the locus of $0's$ of I is $d-1$.

Proof. We note that

$$\frac{1}{(1-q)^d} = \sum_{j=0}^{\infty} \binom{j+d-1}{d-1} q^j.$$

Thus if $P(q) = a_0 + a_1 q + ... + a_m q^m$ then

$$\frac{P(q)}{(1-q)^d} = \sum_{i=0}^{m}\sum_{j=0}^{\infty} a_i \binom{j+d-1}{d-1} q^{j+i}.$$

On the other hand

$$H(A,q) = \sum_{k=0}^{\infty} \dim A^k q^k.$$

This implies that

$$\dim A^k = \sum_{i=0}^{m} a_i \binom{k-i+d-1}{d-1}.$$

We we set

$$h_A(t) = \sum_{i=0}^{m} a_i \frac{(t-i+d-1)(t-i+d-2)\cdots(t-i+1)}{(d-1)!}$$

This implies that if $k \geq m$ then

$$\dim A^k = h(k) = (\sum_{i=0}^{m} a_i)\frac{k^{d-1}}{(d-1)!} + \text{lower degree in } k.$$

This implies that the Hilbert polynomal of the filtered algebra $A_j = A_0 + ... + A_j$ is

$$(\sum_{i=0}^{m} a_i)\frac{t^d}{d!} + \text{lower degree in } t.$$

We note that since $\sum_{i=0}^{m} a_i \in \mathbb{Z} - \{0\}$ that d is indeed the degree of the Hilbert polynomial. ■

We will call $h_A(t)$ the graded (projective) Hilbert polynomial. It has the property that

$$\dim A^k = h_A(k) \text{ for } k >> 0.$$

Exercise 2. Show that if I is as in exercise 1 then $\dim \mathbb{A}^{2n}(I) = n + 1$. Calculate the graded Hilbert polynomial of $A = F[x_1, ..., x_{2n}]/I$.