Gröbner Bases

with *DERIVE*

Alessandro Perotti
Università degli Studi di Trento

Trento, March 2004

Groebner6.dfw - a DfW Utility File for computing Gröbner Bases in DERIVE 6

(Version 1.2 for DERIVE 6.00, February 2004)
Alessandro Perotti, Dept. of Mathematics, University of Trento, Italy
perotti@science.unitn.it
www.science.unitn.it/~perotti/groebner.htm

This DfW utility file contains a function that computes Gröbner bases of a set of polynomials with respect to a monomial ordering and other related functions. DERIVE 6 contains the new internal function GROEBNER_BASIS to construct the Gröbner basis for a collection of polynomials based on lexicographic ordering of the variables. We introduce new functionalities giving the possibility of choosing the monomial ordering. Besides allowing the explicit computation of Gröbner bases and normal forms, the new functions can be used, for example, to eliminate some variables between the equations and to perform the gaussian reduction of a system depending on parameters.
We wish to thank V. Anisiu for his useful suggestions.

• 1. Functions

Groebner(f,v,ord)	reduced Gröbner basis for a set of polynomials
Eliminate(f,v1,v2)	elimination of a set of variables in equations f=0
TotalDegree(f, v)	total degree of a polynomial
OrdMatrix(ord, n)	matrices inducing relevant monomial orderings
ElOrd(k, n)	matrices inducing elimination orderings
LeadingTerm(f, v, ord)	leading term of a polynomial
NF(f, ff, v, ord)	normal form of a polynomial with respect to a set of polynomials
Buchberger(f, v, ord)	(non reduced) Gröbner basis for a set of polynomials
System(e, v, k)	gaussian reduction of a system depending on parameters
Eigensystem(a, ord, v, w)	reduction of the eigensystem Ax=λx
RREF(a, k)	row-reduction of a matrix depending on parameters
RREF(a, b, k)	row-reduction of augmented matrix depending on parameters
LagrangeMultipliers(f,g,v)	computation of min-max points using Lagrange multipliers

• 2. Description of the functions

• 2.1 Computation of Gröbner Bases

Groebner([f1,...,fm], [x1,...,xn], ord)
computes the reduced Gröbner basis for the set of polynomials generated by **f1,...,fm** with respect to the ordered variables **x1,...,xn** and to the monomial ordering described by **ord.**

The argument **ord** that defines the monomial ordering can be:
- an integer between 1 and 4 (or the variables **lex, grlex, grevlex, invlex**), corresponding to **lexicographic, graded lexicographic, graded reverse lexicographic, inverse lexicographic**
- a non-singular matrix with non-negative integral entries that induces the ordering (see [1] and 2.2 for more information).

- The default ordering is **lexicographic** with respect to the variables **x1>x2>...>xn.**
- If also the second argument is omitted, the ordering is **lexicographic** with respect to the variables appearing in **f1,...,fm** ordered by internal Derive ordering.

• Examples

```
#1:  Groebner([x^2 + y^2 + z^2 + w^2, x^2 + 2·y^2 - y·z - w^2, x^3 + z^3 - w^3],

       [x, y, z, w], grevlex)

#2:    [y^2 - y·z - z^2 - 2·w^2, x^2 + y·z + 2·z^2 + 3·w^2, x^3 + z^3 - w^3]

#3:  Groebner([x^2 + y^2 + z^2 + w^2, x^2 + 2·y^2 - y·z - w^2, x^3 + z^3 - w^3, w

       - 1])

#4:  [w - 1, z^12 - 4·z^9 + 5·z^8 + 12·z^6 - 10·z^5 + 5·z^4 - 16·z^3 + 18·z^2 +

       16, 4·y - z^11 + 4·z^8 - 5·z^7 - 8·z^5 + 10·z^4 - 5·z^3 + 8·z^2 - 10·z,

       x + z^3 - 1]

#5:  GROEBNER_BASIS([x^2 + y^2 + z^2 + w^2, x^2 + 2·y^2 - y·z - w^2, x^3 + z^3 -

       w^3, w - 1], [x, y, z, w])

#6:  [w - 1, z^12 - 4·z^9 + 5·z^8 + 12·z^6 - 10·z^5 + 5·z^4 - 16·z^3 + 18·z^2 +

       16, 4·y - z^11 + 4·z^8 - 5·z^7 - 8·z^5 + 10·z^4 - 5·z^3 + 8·z^2 - 10·z,

       x + z^3 - 1]

#7:  Groebner([x^2·y^3 - x·y^2, x^4·y - x], [x, y], [1 3; 3 0])
```

```
#8:                                  [ 3                ]
                                     [x  - x, x·(y - x)  ]
```

Eliminate([f1,...,fm], [x1,...,xh], [y1,...,yk])

constructs equations in which some variables have been eliminated. It eliminates the first set of variables **x1,...,xh** between the polynomial equations **f1=0,...,fm=0** in the variables **x1,...,xh,y1,...,yk.**

- If the third argument is omitted, the equations are considered with respect to all the variables appearing in **f1,...,fm**.
- As it is shown in the examples of section 3, the function can be used in some cases also when the equations are not polynomial.

• Examples

```
                                [ 2    2    2    2   2      2          2       3    3
#9:   Eliminate([x  + y  + z  + w , x  + 2·y  - y·z - w , x + z  - w , w

       - 1], [x, w])

                          [ 2          2             6      3      2    ]
#10:                      [y  - y·z - z  - 2, y·z + z  - 2·z  + 2·z  + 4]

                          [ 2  3      2   4       ]
#11:  Eliminate([x ·y  - x·y , x ·y - x], [x])

#12:                                  []
```

• 2.2 Other functions related to Gröbner Bases

TotalDegree(f, [x1,...,xn])

computes the total degree of a multivariate polynomial **f** in the variables **x1,...,xn**.

```
                        5  7          5
#13:  TotalDegree(2·x ·y  - 3·x·y·z )

#14:                                  12
```

OrdMatrix(ord, n)

gives an integer square matrix of order **n** that induces the rational monomial ordering **ord**, which can be **lex** or 1, **grlex** or 2, **grevlex** or 3, **invlex** or 4. See [1] for more details.

```
#15:  MAP_LIST(OrdMatrix(ord, 3), ord, [1, ..., 4])

                 [[ 1  0  0 ]  [ 1  1  0 ]  [ 1  1  1 ]  [ 0  0  1 ]]
                 [[         ]  [         ]  [         ]  [         ]]
#16:             [[ 0  1  0 ]  [ 1  0  1 ]  [ 1  1  0 ]  [ 0  1  0 ]]
                 [[         ]  [         ]  [         ]  [         ]]
                 [[ 0  0  1 ]  [ 1  0  0 ]  [ 1  0  0 ]  [ 1  0  0 ]]
```

ElOrd(k, n)

gives an integer square matrix of order **n** that induces an elimination ordering with respect to which the first **k** variables always precede the others.

```
#17:  ElOrd(2, 4)
```

```
                                  [ 1  0  0  1 ]
                                  |            |
                                  | 1  0  0  0 |
#18:                              |            |
                                  | 0  1  1  0 |
                                  |            |
                                  [ 0  1  0  0 ]
```

LeadingTerm(f, [x1,...,xn], ord)

computes the leading term of the polynomial **f** with respect to the ordering **ord** in the variables **x1>x2>...>xn.**

- The default ordering is lexicographic with respect to all the variables appearing in **f.**

```
                                 2  3      2  2  2      5
#19:  MAP_LIST(LeadingTerm(2·x ·y  + 3·x ·y ·z  – x·y , [x, y, z], ord),


        ord, [lex, grlex, grevlex, invlex])
```

```
                    [   2  3     2  2  2       5     2  2  2]
#20:                [2·x ·y , 3·x ·y ·z , – x·y , 3·x ·y ·z ]
```

NF(f, [f1,...,fk], [x1,...xn], ord)

returns the remainder of the polynomial **f** on division by **f1,...,fk** with respect to the ordering **ord**.
If **[f1,...,fk]** is a Gröbner basis, the remainder does not depend on the ordering of the polynomials and is called **Normal Form** of the polynomial **f** with respect to **[f1,...,fk]**. It is zero if and only if **f** is a polynomial combination of **f1,...,fk.**

- The default ordering is lexicographic with respect to all the variables appearing in **f.**

```
           2    [     3    3   2            2      2   2          2
#21:  NF(3·x ·y, [x + z  – w , x  + y·z + 2·z  + 3·w , y  – y·z – z  –

          2]
        2·w ], [x, y, z, w], grlex)
```

```
                                  2    2       2        3
#22:                 3·x – 9·y·(z  + w ) – 6·w ·z – 3·w
```

```
           2    [     3    3   2            2      2   2          2
#23:  NF(3·x ·y, [x + z  – w , x  + y·z + 2·z  + 3·w , y  – y·z – z  –

          2]
        2·w ])
```

```
                                6      3  3      6
#24:                      y·(3·z  – 6·w ·z  + 3·w )
```

Buchberger([f1,...,fm], [x1,...,xn], ord)

implements the Buchberger algorithm for the computation of a (non-reduced) Gröbner basis for the set of polynomials generated by **f1,...,fm** with respect to the ordered variables **x1,...,xn** and to the monomial ordering described by **ord.** The basis obtained always containes the generators **f1,...,fm**.

- The function **Groebner(f,v,ord)** uses an improved version of Buchberger algorithm for the computation of the reduced basis.
- The default ordering is lexicographic with respect to all the variables appearing in **f.**

```
                  [ 2    2    2    2   2      2          2       3    3]
#25:  Buchberger([x  + y  + z  + w , x  + 2·y  - y·z - w , x + z  - w ],


        [x, y, z, w], grevlex)

      [ 2    2    2    2   2      2          2       3    3     2
#26:  [x  + y  + z  + w , x  + 2·y  - y·z - w , x + z  - w , - y  + y·z

          2      2]
        + z  + 2·w ]

                  [ 2    2    2    2   2      2          2       3    3]
#27:  Buchberger([x  + y  + z  + w , x  + 2·y  - y·z - w , x + z  - w ])

      [ 2    2    2    2   2      2          2       3    3     2
#28:  [x  + y  + z  + w , x  + 2·y  - y·z - w , x + z  - w , - y  + y·z

           2      2         6      3  3      2    6      2     2     4
        + z  + 2·w , y·z + z  - 2·w ·z  + 2·z  + w  + 3·w , - w ·y·(w  +

             11      3  8      7    5     6      2        3  4      3
        3) + z   - 4·w ·z  + 5·z  + z ·(5·w  + 3·w ) - 10·w ·z  + 5·z  -

           5  2   4              6      2    17      3  14      13
        2·w ·z ·(w  + 3) + z·(3·w  + 7·w ), z   - 6·w ·z   + 5·z   +

         11      6      2        3  10      9      5  8     4
        z  ·(14·w  + 6·w ) - 20·w ·z   + 5·z  - 8·w ·z ·(2·w  + 3) +

         7      6       2        3  6    5     12       8      4
        z ·(25·w  + 13·w ) - 10·w ·z  + z ·(9·w   + 30·w  + 9·w ) -

           5  4     4            7  2   8      4          14      3  11
        2·w ·z ·(5·w  + 13) - 2·w ·z ·(w  + 6·w  + 9), - z   + 4·w ·z

             10      2  8   4            3  7      6    5     9       5
        - 5·z   - 6·w ·z ·(w  + 1) + 10·w ·z  - 5·z  + z ·(4·w  + 12·w )

           2  4     4          4  2   8      4          3  13      12
        - w ·z ·(5·w  + 13) - w ·z ·(w  + 6·w  + 9), 2·w ·z   - 2·z   -

           6  10       3  9       8    7      9       5       2  6     4
        8·w ·z   + 18·w ·z  - 10·z  + z ·(12·w  + 12·w ) - 4·w ·z ·(8·w

                    3  5      4     12       8         3      9       5
        + 3) + 30·w ·z  - 2·z ·(4·w   + 12·w  + 5) + z ·(18·w  + 50·w )
```

$$- 2\cdot w^2\cdot z^2\cdot(5\cdot w^4 + 13) + z\cdot(2\cdot w^{15} + 12\cdot w^{11} + 18\cdot w^7) - 2\cdot w^{12} - 12\cdot w^8 - 18\cdot w^4,\ z^{12} - 4\cdot w^3\cdot z^9 + 5\cdot z^8 + z^6\cdot(6\cdot w^6 + 6\cdot w^2) - 10\cdot w^3\cdot z^5 + 5\cdot z^4 - 4\cdot w^5\cdot z^3\cdot(w^4 + 3) + z^2\cdot(5\cdot w^6 + 13\cdot w^2) + w^{12} + 6\cdot w^8 + 9\cdot w^4\Big]$$

• 2.3 Elementary applications to linear systems and matrices

`System([e1,...,em], [x1,...,xn], [k1,...,ks])`

performs the gaussian reduction of the system of linear equations **`e1=0,...,em=0`** in the variables **`x1,...,xn`** depending polynomially on parameters **`k1,...,ks`**.
If the vector of parameters is omitted, then in the resulting equations may appear rational functions of the parameters and they may be not equivalent to the given equations for some values of the parameters.

#29: $\text{System}\left(\left[x + y + z - 1,\ k^2\cdot x + 4\cdot y + 9\cdot z + 11,\ k\cdot x + 2\cdot y + 3\cdot z + 1\right],\ [x, y, z],\ [k]\right)$

#30: $[(k - 3)\cdot(z + 3),\ (k - 2)\cdot(y - 4),\ y\cdot(z + 3) - 4\cdot z - 12,\ x + y + z - 1]$

From the reduced system we get that if k=3 the system has infinite solutions x = -3-t, y = 4, z = t (t any real), if k=2 the system has infinite solutions x = 4-t, y = t, z = -3 (t any real), else it has a unique solution x = 0, y = 4, z = -3.

#31: $\text{System}\left(\left[x + y + z - 1,\ k^2\cdot x + 4\cdot y + 9\cdot z + 11,\ k\cdot x + 2\cdot y + 3\cdot z + 1\right],\ [x, y, z]\right)$

#32: $\left[3\cdot k\cdot(3 - k)\cdot(z + 3),\ (k^2 - 4)\cdot(y - 4),\ k\cdot x\right]$

`Eigensystem(a, ord, [x1,...,xn], w)`

performs the gaussian reduction of the eigenvalues system $Ax=\lambda x$ associated to a square matrix **a**. The default ordering is **`lex`**, with default linear variables **`x1,...,xn`** and eigenvalue variable **`w`**.

$$\#33:\quad \text{Eigensystem}\begin{bmatrix} 2 & 1 & 0 \\ 1 & 3 & 4 \\ 0 & 0 & 2 \end{bmatrix}$$

$$\#34:\quad \left[x3\cdot(w-2),\ x2\cdot(w^2-5\cdot w+5),\ x2\cdot x3,\ -w\cdot x2+x1+3\cdot x2+4\cdot x3\right]$$

$$\#35:\quad \text{Eigensystem}\left(\begin{bmatrix} 2 & 1 & 0 \\ 1 & 3 & 4 \\ 0 & 0 & 2 \end{bmatrix}, \text{grlex}, [x, y, z], t\right)$$

$$\#36:\quad \left[z\cdot(t-2),\ -x+y\cdot(t-3)-4\cdot z,\ y\cdot z,\ x\cdot(t-2)-y,\ x\cdot z+4\cdot z^2,\ x^2+x\cdot y-y^2-16\cdot z^2\right]$$

RREF(a, [k1,...,ks])

tries to compute a Row Reduced Echelon Form of a matrix **a** depending on parameters **k1,...,ks**.

RREF(a, b, [k1,...,ks])

computes the Row Reduced Echelon Form of the augmented matrix **[a,b]** depending on parameters **k1,...,ks**. The second argument **b** can be a vector or a matrix. Any null row is deleted.

If the parameters are omitted, then the internal function **ROW_REDUCE** is called for a more efficient computation. In this case, the reduced matrix may contain rational functions of the parameters and the reduction may be not valid for some values of the parameters.

$$\#37:\quad \text{RREF}\left(\begin{bmatrix} 2 & 3 & 4\cdot h \\ 3 & 2\cdot h & 1 \\ 0 & 2 & 1 \end{bmatrix}, [h]\right)$$

$$\#38:\quad \begin{bmatrix} 28 & 0 & 5 \\ 0 & 2 & 1 \\ 0 & 0 & 28\cdot h-13 \end{bmatrix}$$

The **RREF** shows that the matrix has rank 2 for h=13/28, rank 3 otherwise.

$$\#39:\quad \text{RREF}\left(\begin{bmatrix} 2 & 3 & 4\cdot h \\ 3 & 2\cdot h & 1 \\ 0 & 2 & 1 \end{bmatrix}, [2, 1, 1], [h]\right)$$

$$\#40: \quad \begin{bmatrix} 28 & 0 & 5 & 9 - 8\cdot h \\ 0 & 2 & 1 & 1 \\ 0 & 0 & 28\cdot h - 13 & 4\cdot h - 1 \end{bmatrix}$$

$$\#41: \quad \mathrm{RREF}\begin{bmatrix} 2 & 3 & 4\cdot h \\ 3 & 2\cdot h & 1 \\ 0 & 2 & 1 \end{bmatrix}$$

$$\#42: \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- **2.4 Another application: the Lagrange multipliers**

`LagrangeMultipliers(f,[g1,...,gk],[x1,...,xn])`
applies the method of Lagrange multipliers for the computation of critical points of a polynomial function **`f(x1,...xn)`** subject to k polynomial constraints **`g1=0,...,gk=0`**. It returns a system of equations reduced with respect to lexicographic ordering, whose solutions are the coordinates of the critical points.

$$\#43: \quad \mathrm{LagrangeMultipliers}(z^2 - x\cdot y\cdot z + x, \left[x^2 + y^2 - 1,\ y\cdot z - 2\right])$$

$$\#44: \quad \left[z^8 - 4\cdot z^6 - 4,\ 2\cdot y - z^5\cdot(z^2 - 4),\ 2\cdot x - z^2\cdot(z^2 - 4)\right]$$

$$\#45: \quad \mathrm{LagrangeMultipliers}(z^2 - x\cdot y\cdot z + x, [y\cdot z - 2])$$

$$\#46: \quad [1]$$

If the constant polynomial 1 is returned, the function has no critical points subject to the constraints.

- **3 Other examples**

In the univariate case, the Gröbner basis contains only the GCD of the polynomials.

$$\#47: \quad \mathrm{Groebner}(\left[x^9 - 3\cdot x^8 + x^7 - 3\cdot x^6 - 3\cdot x^5 + 6\cdot x^4 + 17\cdot x^3 - 22\cdot x^2 - 11\cdot x + 15,\ 3\cdot x^7 - 9\cdot x^6 + 5\cdot x^5 - 15\cdot x^4 - 4\cdot x^3 + 3\cdot x^2 + 48\cdot x - 63\right])$$

$$\#48: \quad [x - 3]$$

The function **`Eliminate`** can be applied to find the fourth degree equation of a torus

starting from its parametric equations. The trigonometric functions can be considered as new variables to be eliminated, subject to conditions sin(t)^2+cos(t)^2=1 and sin(u)^2+cos(u)^2=1.

```
#49:  [(2 + COS(t))·COS(u), (2 + COS(t))·SIN(u), SIN(t)]
```

```
                  ⎡                                                  2     2
#50:  Eliminate(⎣x - (2 + ct)·cu, y - (2 + ct)·su, z - st, st  + ct  -

          2     2    ⎤
       1, su  + cu  - 1⎦, [ct, st, cu, su], [x, y, z])
```

```
       ⎡ 4      2   2    2         4      2   2         4      2    ⎤
#51:   ⎣x  + 2·x ·(y  + z  - 5) + y  + 2·y ·(z  - 5) + z  + 6·z  + 9⎦
```

In the following example the **`Eliminate`** function is called two times to obtain the equations of the projections of a parametric curve on the coordinate planes yz, xz, xy.

```
                                  ⎡            3      4⎤
#52:  gb_xyz := Eliminate(⎣x - t, y - t , z - t ⎦, [t])
```

```
                          ⎡           2      2   3      2   3    ⎤
#53:          gb_xyz := ⎣x·y - z, x ·z - y , y  - x·z , x  - y⎦
```

```
#54:  MAP_LIST(Eliminate(gb_xyz, [v]), v, [x, y, z])
```

```
                                  ⎡  4    3 ⎤
                                  ⎢ y  - z  ⎥
                                  ⎢         ⎥
                                  ⎢  4      ⎥
#55:                              ⎢ x  - z  ⎥
                                  ⎢         ⎥
                                  ⎢  3      ⎥
                                  ⎣ x  - y  ⎦
```

An example taken from Cox,D., Little,J., O'Shea,D., *Ideals, Varieties, and Algorithms*, Springer-Verlag, New York-Berlin-Heidelberg (1992):

```
                ⎡ 5    4    3       3    2    2    ⎤
#56:  polys := ⎣x  + y  + z  - 1, x  + y  + z  - 1⎦
```

```
#57:  Groebner(polys, [x, y, z], grevlex)
```

```
       ⎡ 3    2    2       2   2    2         4    3          4    3
#58:   ⎣x  + y  + z  - 1, x ·(y  + z  - 1) - y  - z  + 1, x·(y  + z  - 1)

          4    2     2         4      2       2  2   2
       + y  + y ·(2·z  - 2) + z  - 2·z  + 1, x ·z ·(z  + z - 2) +

             2   2         4    3      2         6    4  2    2   3
       x·(2·y ·(z  - 1) + z  - z  - 2·z  + 2) + y  - y ·z  + y ·(z  -

          2         5    4    3      2    ⎤
       2·z  + 1) - z  - z  + z  + 3·z  - 2⎦
```

```
#59:  Groebner(polys, [x, y, z], grlex)
```

```
#60: [ 3    2    2       2   2    2         4    3          4    3
     [x  + y  + z  - 1, x ·(y  + z  - 1) - y  - z  + 1, x·(y  + z  - 1)

        4    2     2         4      2       2  2   2
      + y  + y ·(2·z  - 2) + z  - 2·z  + 1, x ·z ·(z  + z - 2) +

           2   2         4    3      2         6    4  2    2   3
      x·(2·y ·(z  - 1) + z  - z  - 2·z  + 2) + y  - y ·z  + y ·(z  -

         2         5    4    3      2          2     4    3      2
      2·z  + 1) - z  - z  + z  + 3·z  - 2, x·(y ·(3·z  - z  - 6·z  +

           6      5      4      3      2         8    6    4     3
      4) + z  - 3·z  - 3·z  + 3·z  + 6·z  - 4) + y  - y  + y ·(2·z  -

         2           2   4      2           6      4      3      2    ]
      5·z  + 3) - 7·y ·(z  - 2·z  + 1) - 2·z  + 9·z  - 2·z  - 9·z  + 4]
```

An assignment to a variable is a convenient way for using a lengthy result in other computations:

```
#61:  PROG(gb := Groebner(polys, [x, y, z], lex),

        MAP_LIST(LeadingTerm(f), f, gb))

              [ 12     11        2       4      2  4   2  2   3]
#62:          [y  , x·z  , 24·x·y ·z, x·y , 12·x ·z , x ·y , x ]
```

This means that the lex Gröbner basis contains 7 polynomials...

```
#63:  MAP_LIST(DIM(TERMS(EXPAND(f))), f, gb)

#64:                          [25, 49, 53, 9, 49, 6, 4]
```

...and that these polynomials contain up to 53 terms...

```
#65:  MAP_LIST(TotalDegree(f), f, gb)

#66:                          [12, 13, 12, 5, 12, 4, 3]
```

...and their maximum total degree is 13.

```
          10  3  3
#67:  NF(x  ·y ·z , gb)

                 11  3    7     6      3     3   9      6    3
#68:            y  ·z  + y ·(2·z  - 2·z ) + y ·(z  - 2·z  + z )

           ⎛⎡ 2   3   4·h ⎤  ⎡ 2  1 ⎤     ⎞
           ⎜⎢             ⎥  ⎢      ⎥     ⎟
#69:  RREF⎜⎢ 3  2·h   1   ⎥, ⎢ 1  1 ⎥, [h]⎟
           ⎜⎢             ⎥  ⎢      ⎥     ⎟
           ⎝⎣ 0   2    1  ⎦  ⎣ 1  1 ⎦     ⎠
```

$$\#70: \quad \begin{bmatrix} 28 & 0 & 5 & 9 - 8\cdot h & 7 - 8\cdot h \\ 0 & 2 & 1 & 1 & 1 \\ 0 & 0 & 28\cdot h - 13 & 4\cdot h - 1 & 4\cdot h - 7 \end{bmatrix}$$

$$\#71: \quad \text{LagrangeMultipliers}(z^2 - x\cdot y\cdot z + x, \begin{bmatrix} x^2 + y^2 - 1 \end{bmatrix})$$

$$\#72: \quad \begin{bmatrix} z\cdot(256\cdot z^6 - 32\cdot z^4 + 17\cdot z^2 + 3), y + z\cdot(32\cdot z^4 - 2\cdot z^2 + 1), z\cdot(2\cdot x + 16\cdot z^4 - 9\cdot z^2 + 1), x^2 - 16\cdot z^4 + z^2 - 1 \end{bmatrix}$$

- **References**

[1] A. Perotti, ***Gröbner Bases with DERIVE***, International DERIVE Journal, Vol.3,n.2, 83-98 (1996)

Appendix: Gröbner Bases

1. NOTATION AND DEFINITIONS

Let k be the coefficient field and $k[x_1, \ldots, x_n]$ the polynomial ring in n indeterminates $x_1, \ldots, x_n$. Every polynomial f is a sum $\sum_{\alpha \in A} a_\alpha x^\alpha$ of *terms* $a_\alpha x^\alpha$ ($a_\alpha \neq 0$), where A is a finite subset of n-tuples of nonnegative integers $\alpha_1, \ldots, \alpha_n$. A *term* is a product of a nonzero coefficient a_α and a *monomial* $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$. The sum $|\alpha| = \alpha_1 + \ldots + \alpha_n$ of the exponents in a monomial is called the *total degree* of the monomial.

Once the order of the variables has been fixed, every monomial is uniquely determined by the n-tuple $\alpha_1, \ldots, \alpha_n$ of its exponents in $\mathbf{Z}^n_{\geq 0}$. A *monomial ordering* is a total ordering on $\mathbf{Z}^n_{\geq 0}$ (or equivalently, on the set of monomials) which is a well-ordering, compatible with the sum of the exponents in $\mathbf{Z}^n_{\geq 0}$: if $\alpha > \beta$, then $\alpha + \gamma > \beta + \gamma$ for any $\gamma \in \mathbf{Z}^n_{\geq 0}$.

Examples: the *lexicographic* order *lex*, defined by $x^\alpha >_{lex} x^\beta$ if the first non-zero component of the vector $\alpha - \beta$ is positive, is a monomial ordering. Another example is the *graded* (or *total*) *lexicographic* order *grlex*, defined by $x^\alpha >_{grlex} x^\beta$ if $|\alpha| > |\beta|$, or $|\alpha| = |\beta|$ and $\alpha >_{lex} \beta$.

Let $f = \sum_{\alpha \in A} a_\alpha x^\alpha$ be a non-zero polynomial and let $>$ be a fixed monomial ordering. The *multidegree* of f is the n-tuple $\mathrm{MDEG}(f) = \max\{\alpha \in \mathbf{Z}^n_{\geq 0}$ such that $a_\alpha \neq 0\}$. The *leading coefficient* of f is $\mathrm{LC}(f) = a_{\mathrm{MDEG}(f)}$, the *leading monomial* is $\mathrm{LM}(f) = x^{\mathrm{MDEG}(f)}$ and the *leading term* is $\mathrm{LT}(f) = \mathrm{LC}(f)\,\mathrm{LM}(f)$.

Examples: with respect to the *lex* order with $x > y$, the polynomial $2x^2y^3 + 5xy^5 - 3xy - 2y + 4$ has multidegree $(2, 3)$ and leading term $2x^2y^3$; with respect to the *grlex* order, the multidegree is $(1, 5)$ and the leading term is $5xy^5$.

2. THE DIVISION ALGORITHM

The choice of a monomial ordering makes it possible to extend the well-known division algorithm for univariate polynomials. Let $F = (f_1, \ldots, f_s)$ be an ordered s-tuple of polynomials in $k[x_1, \ldots, x_n]$. Every polynomial f can be written as

$$f = q_1 f_1 + q_2 f_2 + \cdots + q_s f_s + r$$

where $\mathrm{MDEG}(f) > \mathrm{MDEG}(q_i f_i)$ for any i such that $q_i f_i \neq 0$ and the *remainder* r is a sum of terms, none of which is divisible by a $\mathrm{LT}(f_i)$.

Remark: the quotients $q_1, \ldots, q_s$ and the remainder r depend on the choice of the monomial ordering and, in general, also on the ordering of the divisors $f_1, \ldots, f_s$.

The following algorithm gives $q_1, \ldots, q_s$ and the remainder r:

$q_1 := 0; \ldots; q_s := 0; r := 0$
$p := f$
WHILE $p \neq 0$ DO
    IF there exists a first index i such that $\mathrm{LT}(f_i)$ divides $\mathrm{LT}(p)$ THEN
        $q_i := q_i + \frac{\mathrm{LT}(p)}{\mathrm{LT}(f_i)}$
        $p := p - \frac{\mathrm{LT}(p)}{\mathrm{LT}(f_i)} f_i$
    ELSE
        $r := r + \mathrm{LT}(p)$
        $p := p - \mathrm{LT}(p)$

3. GRÖBNER BASES AND BUCHBERGER'S ALGORITHM

Let I be an ideal of polynomials. The subset $\{g_1, \ldots, g_s\}$ of I is a *Gröbner basis* of I with respect to a fixed monomial ordering if the monomials $\mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_s)$ generate the *initial ideal* $\langle \mathrm{LT}(I) \rangle$ generated by all the leading terms of elements in I. The set $G = \{g_1, \ldots, g_s\}$ is a *reduced Gröbner basis* for I if it is a Gröbner basis such that every element has leading coefficient 1 and for every $g_i \in G$, no monomial of g_i is generated by the leading terms of the other elements of G. This condition guarantees the uniqueness of the reduced Gröbner basis of an ideal.

If $G = \{g_1, \ldots, g_s\}$ is a Gröbner basis of an ideal I, then the ambiguity in the definition of a remainder of f on division by $g_1, \ldots, g_s$ disappears. In this case, the remainder r is also called the *normal form* of f with respect to G, and denoted by $\mathrm{NF}(f, G)$. Otherwise, it is called the normal form of f with respect to the *ordered* set $(g_1, \ldots, g_s)$.

The division algorithm gives the following useful characterization of Gröbner bases. In particular, it provides a membership criterion for the ideal.

Proposition 1: the subset $\{g_1, \ldots, g_s\}$ of I is a Gröbner basis of I if and only if for every $f \in I$ the normal form of f with respect to $g_1, \ldots, g_s$ is zero.

Buchberger's algorithm is based on a stronger version of the last result. We recall the definition of the *S-polynomial* of f and g:

$$S(f, g) = \frac{x^\gamma}{\mathrm{LT}(f)} f - \frac{x^\gamma}{\mathrm{LT}(g)} g$$

where the monomial x^γ is the *least common multiple* of $\mathrm{LM}(f)$ and $\mathrm{LM}(g)$.

Proposition 2: the subset $G = \{g_1, \ldots, g_s\}$ of I is a Gröbner basis of I if and only if for every pair (i, j), $i \neq j$, the normal form of $S(g_i, g_j)$ with respect to G is zero.

Buchberger's algorithm

Let $F = (f_1, \ldots, f_s)$ be a set of generators of the ideal I. A Gröbner basis $G = (g_1, \ldots, g_t)$ of I is obtained by the following algorithm:

$B := \{(i, j) | 1 \leq i < j \leq s\}$
$G := F$
$t := s$
WHILE $B \neq \emptyset$ DO
    select $(i, j) \in B$
    $g := \mathrm{NF}(S(g_i, g_j), G)$
    IF $g \neq 0$ THEN
        $t := t + 1$
        $G := G \cup \{g\}$
        $B := B \cup \{(i, t) | 1 \leq i \leq t - 1\}$
    $B := B - \{(i, j)\}$

This algorithm can be made more efficient by giving conditions to know in advance if the normal form of $S(g_i, g_j)$ does not need to be included in the new generating set. We refer the reader to section 2.9 of Cox et al, 1992 for a complete treatment.

4. ORDERINGS ON $\mathbf{Z}^n$

Definition 1: we call an *ordering* on $\mathbf{Z}^n$ any total ordering compatible with addition in $\mathbf{Z}^n$.

Every monomial ordering $>$ extends to a unique ordering on $\mathbf{Z}^n$. To see this, given $\alpha, \beta \in \mathbf{Z}^n$, let $\gamma(\alpha, \beta)$ be the smallest element (with respect to the well-ordering $>$) of the non-empty set

$\{\gamma' \in \mathbf{Z}^n_{\geq 0} | \alpha - \beta + \gamma' \in \mathbf{Z}^n_{\geq 0}\}$. Then we say $\alpha > \beta$ if and only if $\alpha - \beta + \gamma(\alpha, \beta) > \gamma(\alpha, \beta)$ in $\mathbf{Z}^n_{\geq 0}$. Since $\gamma(\alpha + \delta, \beta + \delta) = \gamma(\alpha, \beta)$ for every $\alpha, \beta, \delta \in \mathbf{Z}^n$, this induced ordering is compatible with the addition in $\mathbf{Z}^n$.

Definition 2: a matrix A in $\mathrm{GL}(n, \mathbf{Z})$ induces an ordering $>_A$ on $\mathbf{Z}^n$ by defining

$$\alpha >_A \beta \qquad \text{if and only if} \qquad \alpha A >_{lex} \beta A$$

Examples: the *lex* order is induced by the identity matrix I_n; the *grlex* order, the *graded inverse lex order* (*grevlex*) and the *inverse lex order* (*invlex*) (see Cox et al, 1992 section 2.2) are induced by the matrices

$$A_{grlex} = \begin{pmatrix} 1 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \end{pmatrix} \quad A_{grevlex} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & \cdots & 1 & 0 \\ \vdots & & & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix} \quad A_{invlex} = \begin{pmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 1 & 0 \\ \vdots & & \vdots & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix}$$

In the following, we shall show that a large class of monomial orderings can be obtained in such a way.

Robbiano (1985 and 1986) has proved that *every* total ordering on $\mathbf{Z}^n$ is induced (in the sense given above) by a *real* nonsingular matrix. In fact, every such ordering extends to a (not necessarily unique) continuous ordering on $\mathbf{R}^n$ (with the euclidean topology). Then it is shown possible to find an orthogonal basis of $\mathbf{R}^n$ whose elements are positive with respect to the ordering. The matrix whose columns are the vectors of the basis in decreasing order induces the ordering on $\mathbf{R}^n$.

The following result, whose easy proof we omit, gives the relation existing between matrices which induce the same ordering.

Proposition 3: two matrices $A, B \in \mathrm{GL}(n, \mathbf{R})$ induce the same ordering on $\mathbf{R}^n$ if and only if they define the same right lateral modulo the (non-normal) subgroup T^+ of upper triangular matrices with positive diagonal. In particular, $A, B \in \mathrm{GL}(n, \mathbf{Z})$ induce the same ordering on $\mathbf{Z}^n$ if and only if $B = AU$, where U is an upper triangular, integer matrix with only 1 on the diagonal.

Definition 3: an ordering $>$ on $\mathbf{Z}^n$ will be called *rational* if there exists a nonsingular rational matrix which induces $>$.

Examples: all the orderings shown above are rational; the following is an example of a non-rational ordering on $\mathbf{Z}^2$: $\alpha > \beta$ if and only if $\sqrt{2}\alpha_1 + \alpha_2 > \sqrt{2}\beta_1 + \beta_2$.

Other rational orderings are *product orders* constructed from two (or more) rational orders (see section 2.4 of Cox et al, 1992): in this case an inducing matrix is the direct sum of the matrices of the given orderings. Another class of rational orderings on $\mathbf{Z}^n$ is given by the *elimination orders* $>_k$ (Bayer and Stillman, 1987): $\alpha >_k \beta$ if and only if $\alpha_1 + \cdots + \alpha_k > \beta_1 + \cdots + \beta_k$, or $\alpha_1 + \cdots + \alpha_k = \beta_1 + \cdots + \beta_k$ and $\alpha >_{grevlex} \beta$. A matrix corresponding to $>_k$ is the following

$$A_{>_k} = \begin{pmatrix} 1 & & & & & & & \\ \vdots & & \mathrm{O} & & & & A^{k-1} & \\ 1 & & & & & & & \\ 1 & 0 & \cdots & 0 & 0 & \cdots & & 0 \\ 0 & & & & & & & \\ \vdots & & A^{n-k} & & & & \mathrm{O} & \\ 0 & & & & & & & \end{pmatrix}$$

where A^i denote the square matrix of order i associated to *grevlex*.

Proposition 4: every rational ordering $>$ is induced by a matrix $A \in \mathrm{GL}(n, \mathbf{Z})$.

Proof: let B be a rational matrix inducing $>$. By means of elementary column operations, the transpose matrix B^T can be transformed into its *Hermite form* H, which is a non-negative, lower triangular matrix. Then $H = B^T K$, where $K \in \mathrm{GL}(n, \mathbf{Z})$. Set $U = H^T$ and $A = (K^T)^{-1}$ and get $AU = B$. □

Now we return to monomial orderings. They are characterized by the following condition (see Corollary 2.4.6 in Cox et al, 1992):

Proposition 5: an ordering on $\mathbf{Z}^n$ restricts to a monomial ordering if and only if every non-zero vector in $\mathbf{Z}^n_{\geq 0}$ is positive with respect to the ordering.

Proposition 6: every rational monomial ordering $>$ is induced by a non-negative matrix in $\mathrm{GL}(n, \mathbf{Z})$.

Proof: let $A \in \mathrm{GL}(n, \mathbf{Z})$ a matrix which induces $>$. For every element e_j of the standard basis of $\mathbf{Z}^n$, we get from Proposition 5 that $e_j A >_{lex} 0$. This means that the rows of A are positive with respect to the *lex* order. By adding to every column an integral linear combination of the preceding columns, A can be transformed into a non-negative matrix. □

This result gives an easy method to compute the leading term with respect to any rational monomial ordering. Let $f \in k[x_1, \ldots, x_n]$ be a polynomial and $A = (a_{ij}) \in \mathrm{GL}(n, \mathbf{Z})$ a non-negative matrix corresponding to the ordering $>$. Consider the following change of variables

$$x_i = \prod_{j=1}^{n} y_j^{a_{ij}}$$

Then the leading term of $f(x_1, \ldots, x_n)$ with respect to $>$ is the monomial obtained by applying the inverse transformation (in general, a rational transformation) to the leading term of $f(y_1, \ldots, y_n) \in k[y_1, \ldots, y_n]$ with respect to the *lex* order.

Remark: in order to construct Gröbner bases, it is sufficient to consider rational monomial orderings, since any real matrix can be approximated by a rational one.

5. THE REDUCED GRÖBNER BASIS

Given a Gröbner basis $F = \{f_1, \ldots, f_s\}$, a reduced Gröbner basis G can be obtained by the following algorithm:

$G := F$
FOR all $g \in G$ DO
    IF there exists $h \in G$, $h \neq g$, such that $\mathrm{LT}(h)$ divides $\mathrm{LT}(g)$ THEN
        $G := G - \{g\}$
    ELSE
        $g := \mathrm{NF}(g, G - \{g\})$
FOR all $g \in G$ DO
    $g := \frac{g}{\mathrm{LC}(g)}$

REFERENCES

Bayer, D. and Stillman, M. (1987). *A theorem on refining division orders by the reverse lexicographic order.* Duke J.Math., 55, 321–328

Becker, T. and Weispfenning, V. (1993). *Gröbner bases*, Springer-Verlag, New York-Berlin-Heidelberg

Carrà Ferro, G. and Sit, W. (1994). On Term-Orderings and Rankings. In Fischer, K., Loustaunau, P., Shapiro, K., Green, E., Farkas, D. (eds.). *Computational algebra*, Lecture Notes in Pure and Applied Mathematics Vol.151, 31–77.

Cox, D., Little, J., O'Shea, D. (1992). *Ideals, Varieties, and Algorithms*, Springer-Verlag, New York-Berlin-Heidelberg

Perotti, A., (1996) *Gröbner Bases with DERIVE*, International DERIVE Journal, Vol.3,n.2, 83–98

Robbiano, L. (1985). *Term orderings on the polynomial ring.* Proceedings EUROCAL 1985, LNCS 204, 513–517

Robbiano, L. (1986). *On the theory of graded structures.* J.Symb.Comp., 2, 139–170