Notes Informelles de Calcul Formel IX
N° M810.0987
SYMMETRIC POLYNOMIALS AND ELIMINATION

Marc Giusti [1]
Centre de Mathématiques (URA CNRS D.0169) / Laboratoire d'Informatique
Ecole Polytechnique
91128 PALAISEAU CEDEX
and
GRECO de Calcul Formel No. 60
uucp : giusti@cmep.polytechnique.fr
bitnet : cfmagi@frpoly11

Daniel Lazard [1]
LITP (tour 45-55)
4, Place Jussieu
75252 Paris Cedex 05
Unité associée au CNRS No. 248
and
GRECO de Calcul Formel No. 60
uucp : ... mcvax!inria!litp!dl
bitnet : dl@frunip11

Annick Valibouze
LITP (tour 45-55)
4, Place Jussieu
75252 Paris Cedex 05
Unité associée au CNRS No. 248
and
GRECO de Calcul Formel No. 60
uucp : ... mcvax!inria!litp!avb
bitnet : avb@frunip11

October 1987

[1] Partially supported by a grant of the PRC "Mathématiques et Informatique".

INTRODUCTION

Let us consider the Graeffe method to find the real roots of a polynomial P : a basic step consists in computing the polynomial whose roots are the square of the roots of the given polynomial. This is one of the simplest example of transforming a polynomial by an algebraic morphism.

A concrete problem is now to be able to perform such a transformation using a standard system of computer algebra, providing the usual algebraic tools. In this particular case a first way is to compute the resultant $R(y) = Res_x(P(x), y - x^2)$. A second method is to remark that the coefficients of the required polynomial are symmetric functions of the roots of P, hence can be expressed in terms of the elementary symmetric functions which are the coefficients of P up to the sign.

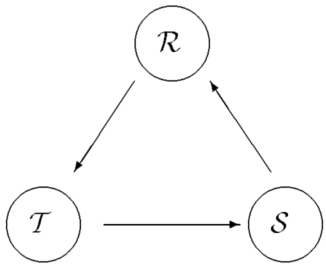
We shall describe in this paper a general transformation of polynomials, and show that the classical deep relationships between the problems :

($\mathcal{T}$) transforming equations by a morphism

($\mathcal{R}$) elementary elimination theory by resultants

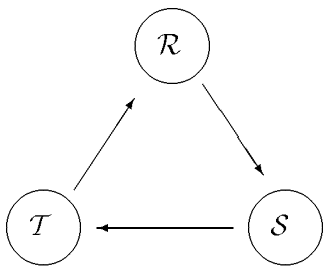
($\mathcal{S}$) change of bases for symmetric polynomials.

can be illustrated in Computer Algebra. Actually they are algorithmically equivalent because every one is a particular case of another one, as shown by the following diagram :



where an arrow linking two problems means that one can describe an algorithm solving the second one if such holds for the first one.

Conversely we will study the possibility of inverting arrows, i.e. finding direct paths as follows :



Eventually we shall give various applications, including comparisons of time. All computations were done on the VAX 780 of the LITP, using MACSYMA and programs written in FRANZ LISP.

1 THE PROBLEMS

1.1 ($\mathcal{T}$) transforming equations by a morphism

1.1.1 Definitions

Let k be a field, and K an algebraic closure of k. We will denote by $\mathbf{N}$ the set of natural (i.e. positive) integers.

Given an integer p, consider an element $I = (i_1, i_2, \ldots, i_p)$ of $\mathbf{N}^p$. Its *weight* $\mid I \mid$ is the sum $i_1 + \cdots + i_p$. Let R_I be the polynomial algebra $k[x_1^{(1)}, \ldots, x_{i_1}^{(1)}, \ldots, x_1^{(p)}, \ldots, x_{i_p}^{(p)}]$. A *transformation of type* I is nothing else than a polynomial f of R_I, to which is associated a mapping :

$$f : K^{i_1} \times K^{i_2} \times \ldots \times K^{i_p} \longrightarrow K$$

which will be also denoted by f.

The set $\mathbf{N}^p$ is naturally partially ordered. If J is larger than I, it gives rise to a natural inclusion $R_I \subseteq R_J$.

The *multidegree* D of a p-uple of univariate polynomials $(P_1, \ldots, P_p)$ is the sequence $(d_1, \ldots, d_p)$ of their degrees.
Now we can define the *direct image* or *transformation* :

$$f_* : k[x]^p \longrightarrow k[x]$$

which associates to any ordered p-uple of monic polynomials $P = (P_1, \ldots, P_p)$, of multidegree D larger than I, the monic polynomial $f_*(P)$ obtained in the following way : informally speaking, it is the univariate polynomial whose roots are the elements of K obtained by substituting in f the variables $x_i^{(j)}$'s by the different roots of P_j. We give now a precise definition of $f_*(P)$:

Since the multidegree D is larger than I, we can associate to every polynomial P_j of degree d_j the set $r(P_j) = (a_1^{(j)}, \ldots, a_{d_j}^{(j)})$ of its d_j roots in K ordered in an arbitrary way $(1 \leq j \leq p)$. Choose an evaluation map $E_a : R_D \longrightarrow K$ which is the algebra homomorphism which sends the variable $x_i^{(j)}$ to $a_i^{(j)}$. The product S_D of symmetric groups $S_{d_1} \times \cdots \times S_{d_p}$ acts naturally on R_D : by the natural inclusion $R_I \subseteq R_D$, f becomes an element of R_D ; let $O_{S_D}(f)$ be its orbit under S_D.

Eventually $f_*(P)$ is the polynomial whose roots are the images by the evaluation map E_a of the elements of the orbit of f, i.e. :

$$f_*(P)(x) = \prod_{g \in O_{S_D}(f)} (x - E_a(g))$$

1.1.2 Properties

By construction, the definition of $f_*(P)$ does not depend on the choice of the evaluation map. The proof is straightforward : if E_b is another choice of an evaluation map, let us call τ the permutation of S_D which sends a on b. Then :

$$
\begin{aligned}
f_*(P)(x) &= \prod_{h \in O_S(f)} (x - E_b(h)) \\
&= \prod_{h \in O_S(f)} (x - E_{\tau(a)}(h)) \\
&= \prod_{h \in O_S(f)} (x - E_a(\tau(h)) \\
&= \prod_{\tau^{-1}(g) \in O_S(f)} (x - E_a(g)) \\
&= \prod_{g \in O_S(f)} (x - E_a(g))
\end{aligned}
$$

Note also that $f_*(P)$ has a priori its coefficients in the closure K, but it is easy to see that they are actually in k. We denote by $e_i^{(j)}$ the i^{th} symmetric elementary polynomial in the variables $x_1^{(j)}, \ldots, x_{d_j}^{(j)}$ $(1 \leq j \leq p,\ 1 \leq i \leq d_j)$. A polynomial of R_D invariant under S_D can be expressed as a polynomial of $k[e_1^{(1)}, \ldots, e_{d_1}^{(1)}, \ldots, e_1^{(p)}, \ldots, e_{d_p}^{(p)}]$. The proof is obvious by induction on p, and we will refer often to these invariant polynomials as "multisymmetric". A symmetric function of the elements of the orbit $O_{S_D}(f)$ being multisymmetric, after specialisation it is now clear that the coefficients of $f_*(P)$ are in k.

1.1.3 Examples

1. In the classical case where the type is (1), $f_*(P)$ is nothing else than the polynomial whose roots are the images of the roots of P by f. The transformation has a nice functorial property, since its respects the product.

   Furthermore if we denote by f^* the algebra homomorphism "inverse image" associated to f (i.e. $f^*(P) = P \circ f$, which is in this particular case the polynomial whose roots are the inverse images by f of the roots of P), we have the two following properties :

   (i) $f_* f^*(P) = P^d$ where d is the degree of P.

   (ii) $f^* f_*(P)$ is divisible by P.

2. Consider the following transformation $f(x_1^{(1)}, x_2^{(1)}) = x_1^{(1)} - x_2^{(1)}$ of type (2). What is the constant term of $f_*(P)$? Obviously the discriminant of P.
   In a similar way, the constant term of the transformation of two polynomials P_1, P_2 by $f(x_1^{(1)}, x_1^{(2)}) = x_1^{(1)} - x_1^{(2)}$ (of type (1,1)) is the resultant of the two polynomials.

1.2 Representation and manipulation of the algebra of symmetric polynomials

A standard computer algebra system must provide the usual algebraic tools allowing to handle with a polynomial algebra. This implies on the first hand a data structure and an internal representation, on the second hand algorithms and the corresponding implementations of the algebra operations.

In applications of computer algebra, we frequently have to manipulate objects which are invariant under some group of permutation of the indeterminates, for example symmetric polynomials. But we are quickly stucked in a space problem if we use the standard previous tools, since the symmetric group S_n contains $n!$ permutations. So we need to represent this algebra of invariants in the most contracted way : we don't want to list the elements of an invariant subset, but only a fundamental domain. For example, a monomial form is the sum of the elements of the orbit of a single monomial, and we need only to store an element and the operation "+". Note immediately that we have to pay something since we assume implicitely that we are able to compute the isotropy subgroup of a monomial ...

To complete this data structure, we need two functions inverse of each other, one of contraction and the other one of explosion, passing from the usual representation of polynomials in the system to the contracted one and conversely.

Furthermore algorithms for the internal operations in the algebra must be available, with the corresponding implementations. Eventually if the algebra admits several bases as algebra or vector spaces, we need the corresponding algorithms allowing to expand on a new basis a symmetric polynomial given on an old one.

All these goals are attained in the extension SYM to MACSYMA, written by A. VALIBOUZE [V1].

2 EQUIVALENCE OF THE THREE PROBLEMS

2.1 $(\mathcal{R}) \Longrightarrow (\mathcal{T})$

2.1.1 Relationship with elimination

Let us consider the affine algebraic subvariety V of $K^{|I|+1}$ defined by the $\mid I \mid +1$ equations :

$$\begin{aligned} P_1(x_1^{(1)}) &= \cdots = P_1(x_{i_1}^{(1)}) = 0 \\ &\ldots \\ P_p(x_1^{(p)}) &= \cdots = P_p(x_{i_p}^{(p)}) = 0 \\ y &= f(x_1^{(1)}, \ldots, x_{i_1}^{(1)}, \ldots, x_1^{(p)}, \ldots, x_{i_p}^{(p)}) \end{aligned}$$

Let y_0 be a root of $f_*(P)$; then there exists common roots of the previous equations in the variables $x_i^{(j)}$ if we set y to y_0.
Hence y_0 belongs to the projection of V on the y-axis. Thus solving the problem of transforming equations is part of an *elimination* problem of the variables $x_i^{(j)}$ among the previous equations.

The variety V is obviously a complete intersection of dimension 0 and degree $d_1^{i_1} \ldots d_p^{i_p}$. The projection of V on the y-axis is again of same dimension, hence is defined set-theoretically by a univariate polynomial with well-defined roots, whose multiplicity varies unfortunately with the various definitions of elimination.

Here it is one of the seldom cases where the elimination can be done by the use of two by two resultants, since every variable $x_i^{(j)}$ occurs only in two equations. We have the following theorem :

2.1.2 Theorem :

Let $R(y)$ be the polynomial obtained by eliminating two by two the variables $x_i^{(j)}$ among the equations defining V. If $G_{S_I}(f)$ is the isotropy subgroup fixing f, the polynomial $f_*(P)^{\#G_{S_I}}$ divides R.

Proof : Rename the variables $x_i^{(j)}$ ($1 \le j \le p$, $1 \le i \le i_j$) as z_u ($1 \le u \le\mid I \mid$). Similarly define the polynomials Q_u ($0 \le u \le\mid I \mid$) like that :

$$Q_0 = y - f(x_1^{(1)}, \ldots, x_{i_1}^{(1)}, \ldots, x_1^{(p)}, \ldots, x_{i_p}^{(p)})$$

and for u larger than 1, $Q_u = P_q$ where q is the smallest integer such that $u < i_1 + \cdots + i_q$. Consider now the following sequence of resultant polynomials R_u depending on the $\mid I \mid -u$ last variables ($1 \le u \le\mid I \mid$) :

$$\begin{aligned} R_0 &= Q_0 \\ R_u &= Res(Q_u(z_u), R_{u-1}, z_u)\ ,\ 1 \le u \le\mid I \mid \end{aligned}$$

By the definition of the resultant, R_u is obtained by multiplying the specialisations of R_{u-1} obtained by setting the values of the variable z_u to every element of $r(Q_u)$ (recall this notation for the set of the roots of the polynomial Q_u in the algebraic closure K). By an easy induction on u we get :

$$\begin{aligned} R_u &= \prod_{(c_1,\ldots,c_u)\in r(Q_1)\times\cdots\times r(Q_u)} Q_0(c_1, \ldots, c_u, z_{u+1}, \ldots, z_{|I|}) \\ &= \prod_{(c_1,\ldots,c_u)\in r(Q_1)\times\cdots\times r(Q_u)} (y - f(c_1, \ldots, c_u, z_{u+1}, \ldots, z_{|I|})) \end{aligned}$$

Hence $R = R_{|I|}$ is nothing else than the product :

$$\begin{aligned} R_{|I|} &= \prod_{(c_1,\ldots,c_{|I|})\in r(Q_1)\times\cdots\times r(Q_{|I|})} Q_0(c_1, \ldots, c_{|I|}) \\ &= \prod_{(c_1,\ldots,c_{|I|})\in r(Q_1)\times\cdots\times r(Q_{|I|})} (y - f(c_1, \ldots, c_{|I|})) \\ &= \prod_{(b^{(1)},\ldots,b^{(p)})\in r(P_1)^{i_1}\times\cdots\times r(P_p)^{i_p}} (y - f(b_1^{(1)}, \ldots, b_{i_1}^{(1)}, \ldots, b_1^{(p)}, \ldots, b_{i_p}^{(p)})) \end{aligned}$$

So there is an obvious factor F obtained by extending the product only to *two by two distinct* i_j-uples of roots of P_j ($1 \le j \le p$). For any set E, let us denote by E_*^n the subset of the n-cartesian product E^n consisting of the n-uples of elements of E two by two distinct. This factor can be written :

$$F(y) = \prod_{(b^{(1)},\ldots,b^{(p)})\in r(P_1)_*^{i_1}\times\cdots\times r(P_p)_*^{i_p}} (y - f(b_1^{(1)}, \ldots, b_{i_1}^{(1)}, \ldots, b_1^{(p)}, \ldots, b_{i_p}^{(p)}))$$

Choose now a particular element $a = (a^{(1)}, \ldots, a^{(p)})$ of $r(P_1)_*^{i_1} \times \cdots \times r(P_p)_*^{i_p}$, and let us associate to it an evaluation map E. The product S_D acts naturally on the product $r(P_1)_*^{i_1} \times \cdots \times r(P_p)_*^{i_p}$, and :

$$
\begin{aligned}
F(y) &= \prod_{(\sigma_1,\ldots,\sigma_p)\in S_D/G_{S_D}(f)} (y - f(a^{(1)}_{\sigma_1(1)},\ldots,a^{(1)}_{\sigma_1(i_1)},\ldots,a^{(p)}_{\sigma_p(1)},\ldots,a^{(p)}_{\sigma_p(i_p)}))^{\#G_{S_I}(f)} \\
&= (f_*(P))^{\#G_{S_I}(f)}
\end{aligned}
$$

Note that the degree of this factor F is the degree of $f_*(P)$ times the cardinal of $G_{S_I}(f)$, i.e. the cardinal of S_D divided by the cardinal of $G_{D\setminus I}$, i.e. eventually $\prod_{r=1}^{p}(d_r - i_r + 1$ to be compared to the degree of $R(y)$, which is $\prod_{r=1}^{p} d_r^{i_r}$).

Now to conclude it is enough to compute a squarefree decomposition of R, and to throw away the other factors, which can be assumed known by induction. Actually if we consider the collapsed transformations obtained by equaling two or more variables corresponding to the same component of I, the parasite factors are some power of such transformations of P.

2.2 $(\mathcal{T}) \Longrightarrow (\mathcal{S})$

It will be sufficient to show that we can compute change to and from a particular basis, i.e. the symmetric elementary polynomials.

Given $X = (x_1,\ldots,x_n)$. we want to compute a monomial form $M_I(X)$ (i.e. the sum of the monomials of the orbit of X^I under the action of S_n) as a function of the elementary symmetric functions $e_1, e_2, \ldots, e_n$ of the x_i's using $(\mathcal{T})$. Let us call P the polynomial $x^n + \sum_{i=1}^{n}(-1)^i e_i x^{n-i}$. If we choose $p = 1$ and f of type $lg(I) = l$ as follows :

$$f(x_1, x_2, \ldots, x_l) = x_1^{i_1} x_2^{i_2} \ldots x_l^{i_l}$$

the opposite of the coefficient of y^{n-1} in $f_*(P)(y)$ is exactly M_I. This is an immediate consequence of the definition of $f_*(P)$.

2.3 $(\mathcal{S}) \Longrightarrow (\mathcal{R})$

Let P and Q be two univariate monic polynomials of degree respectively p and q, with coefficients in k. Call $x_1,\ldots,x_p$ the roots of P in K. Then the resultant R of P and Q is the product $Q(x_1)\ldots Q(x_p)$, which is a symmetric function of $Q(x_1),\ldots,Q(x_p)$, so can be expressed with the symmetric elementary functions of the x_i's, i.e. the coefficients of P up to the sign.

Furthermore this is done in a contracted way by SYM [V1] : the invariant product $Q(x_1),\ldots,Q(x_p)$ can be easily computed by the GIRARD formula from the power functions $\sum_{x_i\in r(P)} Q^r(x_i)$ $(1 \leq r \leq p)$. Now the computation can be realized on a contracted form, since $Q^r(a)$ is trivially expressed as linear combination of powers of a. Summing over the orbit of S_n, we see that the wanted power functions are linear combinations of power functions of the x_i's, again computed by the GIRARD formulas from the coefficients of P.

This (disymmetric !) method is available in SYM under the name "RESULSYM". It appears to be better than the classical subresultant algorithm when the coefficients of the polynomials are generic or involve complicated expressions [V1].

3 THE CONVERSE...

3.1 $(\mathcal{R}) \Longleftarrow (\mathcal{T})$

Elementary, my dear Watson : let us consider two univariate polynomials P and Q. Consider the transformation $f = x_1^{(1)} - x_1^{(2)}$ of type (1,1) : then the resultant of P and Q is the product $\prod_{a \in r(P), b \in r(Q)} (a-b)$, i.e. $f_*(P,Q)(0)$.

3.2 $(\mathcal{T}) \Longleftarrow (\mathcal{S})$

We have to look for the symmetric elementary functions of the roots of $f_*(P_1, \ldots, P_p)$, knowing those of the roots of the polynomials P_k from the coefficients of these polynomials.

We define the *monomial multi-form* $M_I(X)$ as the sum of the monomials of the orbit of some monomial under the action of S_D. Conversely, the *contracted form* of this monomial form is any monomial of the orbit.

3.2.1 Description of the algorithm

1. Compute the orbit $O = O_{S_D}(f)$, unfortunately by following a straightforward exhaustion algorithm. Let o the cardinal of this orbit.

2. Compute the o first power functions of the elements of this set, but in a contracted form in order to avoid space explosion.

3. Express the previous multi-symmetric polynomials as polynomials in the symmetric elementary functions of the roots of the polynomials $P_1, \ldots, P_p$, using the basic contracted algorithm described in SYM.

3bis. Similarly it is possible to express the previous multi-symmetric polynomials as polynomials in the power functions of the roots of the polynomials $P_1, \ldots, P_p$, themselves computed from the coefficients by the GIRARD-NEWTON formula . All can be performed in a contracted way by SYM [V1].

4. Obtain the first o symmetric elementary functions of the roots of the polynomial $f_*(P_1, \ldots, P_p)$ from the o first power functions computed in step 3. (or 3bis.), using again the GIRARD-NEWTON formula .

We refer to [V2] for more details.

3.2.2 Example

Let $p = 2$ with $x^{(1)} = (x, y)$, $x^{(2)} = (a, b)$ and $f(x, y, a, b) = a * x + b * y$. For a polynomial P we note $e_i(P)$ the i^{th} symmetric elementary function and $p_i(P)$ the i^{th} power function of its roots. We have :

$$\begin{aligned} O = O_{S_D}(f) &= (ax + by, ay + bx) \\ p_1(O) &= ax + by + ay + bx \\ p_2(O) &= (ax + by)^2 + (ay + bx)^2 \end{aligned}$$

Contracted forms p_1 and p_2 of the multi-symmetric polynomials $p_1(O)$ and $p_2(O)$ would be for example :

$$\begin{aligned} p_1 &= ax \\ p_2 &= a^2x^2 + 4axby \end{aligned}$$

and the multi-decompositions in symmetric elementary functions or power functions of the roots of P_1 and P_2 would give respectively :

$$\begin{aligned} p_1(f_*(P_1, P_2)) &= e_1(P_1)e_1(P_2) \\ &= p_1(P_1)p_1(P_2) \\ p_2(f_*(P_1, P_2)) &= (e_1(P_1)^2 - 2e_2(P_1))(e_1(P_2)^2 - 2e_2(P_2)) + 4e_2(P_1)e_2(P_2) \\ &= p_2(P_1)p_2(P_2) + (p_1(P_1)^2 - p_2(P_1))(p_1(P_2)^2 - p_2(P_2)) \end{aligned}$$

3.3 $(\mathcal{S}) \Longleftarrow (\mathcal{R})$

Let $A = (a_1, a_2, \ldots, a_d)$. Call P the polynomial such that $rac(P) = A$ and I the partition $I = (i_1, i_2, \ldots, i_r)$ where $i_r > 0$ and $r \leq d$. We want to rewrite the monomial form $M_I(A)$ with the elementary symmetric functions $e_i(A)$. Then we consider the affine subvariety of k^{r+1} defined by the $r+1$ following equations :

$$\begin{aligned} P(y - x_1^{i_1}x_2^{i_2}\ldots x_r^{i_r}) &= 0 \\ P(x_1) &= 0 \\ &\ldots \\ P(x_r) &= 0 \end{aligned}$$

and its projection on the y-axis, obtained by eliminating the variables x_i's by taking succesively the resultant of the only two polynomials where they occur. Let us call $R(y)$ the last resultant, i.e. :

$$\begin{aligned} R(y) &= \prod_{a \in rac(P)^r} P(y - a^I) \\ &= \prod_{a \in rac(P)^r, \beta \in rac(P)} (y - a^I - \beta) \end{aligned}$$

The degree of R is d^{r+1}, hence the opposite of the coefficient of the subdominant term is equal to $d^r e_1(A) + \#G_{S_d}(x^I)M_I + \ldots$, the remaining terms being similar products $\#G_{S_d}(x^L)M_L$, where the L's are the partitions of same weight obtained from I by equaling two or more variables. Thus we are reduced to compute the power function of degree $\mid I \mid$ knowing the symmetric elementary functions. But looking then to the resultant $Res_x(P(y - x^{|I|}), P(x))$, we see that the opposite of the coefficient of the subdominant term is $d(e_1(A) + p_{|I|})$, thus we extract easily the wanted power function.

4 An application

4.1 Introduction

To illustrate what we said before we shall treat an example, proposed by P. CARTIER.

Let us consider the polynomial $P(x) = x^7 - 7x + 3$. P. CARTIER suspected and proved that its GALOIS group is the group of the automorphisms of the projective plane over $\mathbf{Z}_2$. This finite projective plane contains 7 points and 7 lines, each line containing 3 points and through each point 3 lines are passing. The cardinal of this group is 168 instead of 7!=5040.

The first step is to prove that the GALOIS group is not the whole symmetric group S_7. Inspired by the geometric configuration, P. CARTIER considered the action on the triples of points, hence studied the polynomial of degree 35, whose roots are the sums 3 by 3 of the roots of P. If this polynomial is not irreducible, the action of the GALOIS group is not transitive and it cannot be the whole symmetric group. This sufficient assertion was checked by numerical computations.

4.2 Checking by Computer Algebra

We have p=1, and $f(x_1, x_2, x_3) = x_1 + x_2 + x_3$; we want to compute the direct image $f_*(P)$.

4.2.1 Brute force use of resultants

Trying to use the theorem 2.1.2 fails for space problems.

4.2.2 More subtle use of resultants

```
(c2) Cartier(pol):=
block([p,a,b,c,d,e,f,solp3,sol],
p[x] : pol,
a[z] : resultant(p[x],ev(p[x],x=z-x),x),
b[z] : rat(a[z]/ev(p[x],x=z/2),z),
c[z] : sqfr(b[z]),
p1[z] : part(c[z],2,1),
cartier[y] : resultant(ev(p[x],x=y-z),p1[z],z),
return(sqfr(cartier[y])))$

Time= 0 msec.

(c3) Cartier(x**7-7*x+3);

/systeme/macsyma.309/rat/result.o being loaded.

Totaltime= 1506133 msec.  GCtime= 700833 msec.

              35         29            28           23           22           21
(d3) - expt(y    - 280 y       + 906 y    - 79086 y    - 56826 y    + 34452 y

            17             16              15                17              11
 + 1739696 y    + 408366 y    + 1139292 y       + 978750 y    - 12357947 y

            10              9              8             7               5
 + 1393266 y    - 9345672 y   + 4975614 y   - 592029 y   + 29042496 y
```

```
            4            3            2
 - 12446784 y  - 2222640 y  + 1227744 y  + 36288 y - 31104, 3)/2187
```

4.2.3 Use of algorithm 3.2.1

To obtain the wanted polynomial, approximatively 8 hours are needed.

4.2.4 More subtle : use of the symmetry of f

In this particular case, it is possible to perform the step 1. and 2. of the algorithm 3.2.1, i.e. the computation of the power functions of the elements of the orbit of f in a contracted way. If we define $EXP(r,h) = \{I = (i_1, \ldots, i_h) \ / \ i_1 \geq \ldots \geq i_h > 0 \ et \ i_1 + \ldots + i_h = r\}$ we get :

$$P_r = \sum_{h=1}^{inf(k,r)} \binom{n-h}{k-h} \sum_{I \in EXP(r,h)} \binom{r}{i_1, \ldots, i_h} M_I(A)$$

where the $M_I(A)$ are monomial forms on A and $\binom{r}{i_1, \ldots, i_h}$ is the multinomial coefficient.

Now only 27 mn (through the elementary functions) and 2 mn 30 (through the power functions) are enough.

4.2.5 The end

```
(c2) factor(cartier);

Totaltime= 183733 msec.   GCTime= 60900 msec.


   7       4       2              28       25       23       22         21
 (x  + 14 x  - 42 x  - 21 x + 9) (x   - 14 x   + 42 x   - 63 x   + 897 x

         20        19          18          17          16          15
 - 1176 x   + 588 x   - 10668 x   + 14700 x   - 51345 x   + 62538 x

           14          13           12           11           10
 - 179421 x   + 293706 x   - 487452 x   + 760116 x   - 956137 x

           9            8            7            6            5
+ 1472415 x  - 1497363 x  + 1674299 x  - 1568784 x  + 1154832 x

          4         3           2
- 876624 x  - 7056 x  + 110880 x  - 4032 x - 3456)
```

REFERENCES

[V1] A. VALIBOUZE, Fonctions Symétriques et changements de bases, to be published in Proceedings of the European Conference on Computer Algebra EUROCAL 87 (Leipzig, RDA), 1987.

[V2] A. VALIBOUZE, Images directes de fonctions multisymétriques, Note interne du LITP $n°$8766, 1987.