J. Symbolic Computation (1989) **7**, 527–531

The Kernel of the Kodaira–Spencer Map of the Versal μ-Constant Deformation of an Irreducible Plane Curve Singularity with $\mathbb{C}^r$-action

BERND MARTIN AND GERHARD PFISTER

Humboldt-Universität zu Berlin, Sektion Mathematik, Unter den Linden 6, 1086 Berlin PSF 1297, DDR

(*Received 15 May 1986*)

An algorithm is described which gives a base of the kernel of the Kodaira–Spencer map of the versal μ-constant deformation of certain plane curve singularities. This is useful for computing the moduli of such singularities.

Let $\mathbb{C}[X, Y]$ be the polynomial ring over the field of complex numbers $\mathbb{C}$ considered as a graded ring with $\deg X = b$ and $\deg Y = a$, a and b being relatively prime. $f = X^a + Y^b$ is a homogeneous polynomial of degree $d = ab$ with respect to this graduation.

Let $B = \{m_1, \ldots, m_\mu\}$ be a monomial basis of $\mathbb{C}[X, Y]/(\partial f/\partial X, \partial f/\partial Y)$ ordered by degree ($\deg X^\alpha Y^\beta := \alpha b + \beta a$), such that $m_1 = 1$ and $m_\mu = X^{a-2}Y^{b-2}$, $\mu = (a-1)(b-1)$. Let $B_u = \{m_{\mu-r+1}, \ldots, m_\mu\}$ be the set of monomials of B of degree greater than d.

Let

$$F := f + \sum_{i=1}^{r} T_i m_{\mu-r+1} \in \mathbb{C}[X, Y, \mathbf{T}], \quad \mathbf{T} := (T_1, \ldots, T_r),$$

and let $\Delta F := (\partial F/\partial X, \partial F/\partial Y)$ be the Jacobian ideal of F, then $R := \mathbb{C}[X, Y, \mathbf{T}]/\Delta F$ is a free $\mathbb{C}[\mathbf{T}]$-module of rank μ generated by $m_1, \ldots, m_\mu$.

The multiplication by F in R, $F: R \to R$ is a $\mathbb{C}[\mathbf{T}]$-linear map. Denote by $C(\mathbf{T}) = (c_{ij})$ the matrix of F with respect to the basis B, i.e.

$$m_i F = \sum_j c_{ij} m_j \bmod \Delta F.$$

In section 1 we explain why we are interested in the matrix $C(\mathbf{T})$. In section 2 we give an algorithm to compute $C(\mathbf{T})$, one which does not use elimination theory.

1. The kernel of the Kodaira–Spencer map

Denote by $X \subseteq \mathbb{C}^2 \times \mathbb{C}^r$ the hypersurface defined by $F = 0$ and let $\pi: X \to \mathbb{C}^r$ be the canonical projection. We may consider $\pi: X \to \mathbb{C}^r$ to be a family of curves: The fibres $\pi^{-1}(t) \subseteq \mathbb{C}^2$ are curves having an isolated singular point at 0. Because of the choice of the family (the monomials $m_{\mu-r+i}$ have degree greater than d) the topological type of the singularity of $\pi^{-1}(t)$ does not change (cf. Teissier, 1976; Zariski, 1976). Furthermore, the family is versal with this property, i.e. any deformation of the singularity of $\pi^{-1}(0)$ having fibres of the same topological type can be induced by the family $\pi: X \to \mathbb{C}^r$. But this family

0747–7171/89/060527+05 \$03.00

is not universal, i.e. it contains analytically trivial subfamilies. These subfamilies are given by the integral manifolds of the kernel of the Kodaira–Spencer map: The Kodaira–Spencer map for the family $\pi : X \to \mathbb{C}^r$ is the map

$$\rho : \mathrm{Der}_{\mathbb{C}}\mathbb{C}[\mathbf{T}] \to \mathbb{C}[X, Y, \mathbf{T}]/(F, \Delta F)$$
$$\rho(\delta) := \text{class of } \delta F.$$

(cf. Laudal, 1979; Laudal & Pfister, 1983; Laudal *et al.*, 1986). The kernel L of the Kodaira–Spencer map is a Lie-algebra, finitely generated as $\mathbb{C}[\mathbf{T}]$-module. Along the integral manifolds of L the family $\pi : X \to \mathbb{C}^r$ is analytically trivial, i.e. a product.

Consider the matrix $C(\mathbf{T}) = (c_{ij})$ of the multiplication $F : R \to R$ with respect to the basis B:

$$m_i F = \sum_{j=1} c_{ij} m_j \bmod \Delta F \tag{I}$$

and let

$$\delta_i := \sum_{j=1}^{r} c_{i,\mu-r+j}\, \partial/\partial T_j \in \mathrm{Der}_{\mathbb{C}}\mathbb{C}[\mathbf{T}], \quad i = 1, \cdots, \mu.$$

LEMMA: (1) $\delta_i \in L, \quad i = 1, \cdots, \mu,$
(2) $\{\delta_i\}$ *generate* L *as* $\mathbb{C}[\mathbf{T}]$-*module*,
(3) $\delta_i = 0$ *if* $i > r$.

PROOF: (1) $\delta_i F = \sum_{j=1}^{r} c_{i,\mu-r+j}\, \partial F/\partial T_j = \sum_{j=1}^{\mu} c_{ij} m_j,$

because $c_{ij} = 0$ for $j < \mu - r + 1$ (compare the degrees in equation (I)). Hence we have $b_i F = m_i F \bmod \Delta F$, i.e. $\delta_i \in L$.

(2) Let $\delta = \sum_{j=1}^{r} w_j\, \partial/\partial T_j \in L$, i.e.

$$\delta F = \sum_{j=1}^{r} w_j m_{\mu-r+j} = H \cdot F \bmod \Delta F \text{ for suitable } H.$$

Let $H = \sum_{j=1}^{\mu} H_j m_j \bmod \Delta F$, $H_j \in \mathbb{C}[\mathbf{T}]$, then $\delta = \sum_{j=1}^{\mu} H_j \delta_j$.

(3) Holds because $\deg m_i F > \deg m_\mu$ if $i > r$.

EXAMPLE. $f = X^6 + Y^7$, $d = 42$, $\mu = 30$, $m_\mu = X^4 Y^5$, $r = 6$,
$F = X^6 + Y^7 + T_1 X^2 Y^5 + T_2 X^3 Y^4 + T_3 X^4 Y^3 + T_4 X^3 Y^5 + T_5 X^4 Y^4 + T_6 X^4 Y^5$.

$$\delta_1 = -1/42(2T_1\,\partial/\partial T_1 + 3T_2\,\partial/\partial T_2 + 4T_3\,\partial/\partial T_3 + 9T_4\,\partial/\partial T_4 + 10T_5\,\partial/\partial T_5 + 16T_6\,\partial/\partial T_6),$$
$$\delta_2 = -1/42(3T_2\,\partial/\partial T_4 + (4T_3 - 10/7\,T_1^2)\partial/\partial T_5$$
$$+ (10T_5 + 4/7\,T_3^2 T_1 - 10/49\,T_3 T_1^3 - 92/147\,T_2^2 T_1^2)\partial/\partial T_6),$$
$$\delta_3 = -1/42(2T_1\,\partial/\partial T_4 + 3T_2\,\partial/\partial T_5 + (9T_4 + 46/21\,T_3 T_2 T_1)\partial/\partial T_6),$$
$$\delta_4 = -1/42(4T_3 - 10/7\,T_1^2)\partial/\partial T_6),$$
$$\delta_5 = -3/42\,T_2\,\partial/\partial T_6,$$
$$\delta_6 = -2/42\,T_1\,\partial/\partial T_6.$$

Because of $c_{ij} = 0$ for $j < \mu - r + 1$ or $i > r$, it is enough to consider the multiplication by F on the submodule R_1 of R generated by $B_1 := \{m_1, \ldots, m_r\}$. For short let $n_i := m_{\mu - r + i}$, $i = 1, \ldots, r$. Let R_2 be the submodule generated by $B_u = \{n_1, \ldots, n_r\}$, then we have a map $F : R_1 \to R_2$. Let

$$m_i F = \sum_{j=1}^{r} \bar{c}_{ij} n_j \bmod \Delta \mathrm{F}.$$

Notice that $\bar{c}_{ij} = c_{i, \mu - r + j}$.

2. The algorithm

Choose u and v with the following properties:

(i) $1 \le u \le a - 1, \quad 1 \le v \le b - 1,$

(ii) $bu \equiv 1 \bmod a, \quad av \equiv 1 \bmod b.$

Let $e(n) \equiv nu \bmod a$ and $e'(n) \equiv nv \bmod b$.

Lemma

(1) *deg* $X^{e(n)} Y^{e'(n)} = \begin{cases} n & \text{if } e(n)b + e'(n)a < d, \\ n + d & \text{else;} \end{cases}$

(2) *if deg* $m_i = n$, $1 \le i \le r$, *then* $m_i = X^{e(n)} Y^{e'(n)}$;

(3) *if deg* $n_i = n + d$, $1 \le i \le r$, *then* $n_i = X^{e(n)} Y^{e'(n)}$;

(4) *if* $X^{e(n)} Y^{e'(n)} \notin B_1 \cup B_u$ *for* $n \le ab - 2(a+b)$, *then* $e(n) = a - 1$ *or* $e'(n) = b - 1$.

The proof is not difficult, we will omit it here.

The sequence $\{(e(n), e'(n)) \mid 0 \le n \le ab - 2(a+b)\}$ provides an effective way to construct the ordered monomial bases B_1 and B_u. The information we need can be arranged in a string S.

Part (i) of the algorithm

Compute u and v such that $bu \equiv 1 \bmod a$, $av \equiv 1 \bmod b$;

$$j := 1;\ i := e := e' := \emptyset;\ S(1) := \text{'}l\text{'};\ l(1) := \emptyset;$$

FOR $k := 1$ TO $d - 2(a+b)$ DO BEGIN
    $e := e + u \bmod a$; $e' := e' + v \bmod b$;
    IF $(e = a - 1)$ OR $(e' = b - 1)$ THEN $S(k+1) :=$ '▯' ELSE
        IF $(eb + e'a) > d$ THEN BEGIN
            $i := i + 1$; $l(i) := k$; $S(k+1) :=$ 'l' END
{characterises monomials of B_1, $l(i)$ is the degree of m_i}
        ELSE BEGIN $j := j + 1$; $u(j) := k$; $S(k+1) :=$ 'u' END
{characterises monomials of B_u, $u(j) + d$ is the degree of n_j}
    END

Example. $f = X^5 + Y^{14}$, $u = 4$, $v = 3$

$$S = \text{'}l▯uuul▯uu▯l▯uull▯uull▯u▯ll▯ulll▯u\text{'}$$

Notice that $S(n) = 'u'$ iff $S(N+1-n) = 'l'$, $N :=$ length of the string S, i.e. it is sufficient to compute only half the string. Now a monomial $T_{q(1)} \cdot \ldots \cdot T_{q(s)}$ occurs in $\bar{c}_{ij}$ of the matrix iff

(i) $m_i n_{q(1)} \ldots n_{q(s)} \equiv n_j$
(ii) $m_i n_{q(1)} \ldots n_{q(t)} \notin B_u$ for $t = 1, \ldots, s-1$.
(the exponents of X, resp. Y, are taken modulo a, resp. b).

This is equivalent (in the language of our string) to

(i') $l(i) + u(q(1)) + \ldots + u(q(s)) = u(j)$
(ii') $S(l(1) + u(q(1)) + \ldots + u(q(t))) \neq 'u'$ for $t = 1, \ldots, s-1$.

PART (II) OF THE ALGORITHM

Computing the monomials of $\bar{c}_{ij}$

```
PROCEDURE bracket (r, s);
BEGIN b := s−r; F := F+'(';
  REPEAT q := max i: u(i) ≤ b;
    WHILE q > ∅ DO BEGIN b := u(q)−1;
      IF u(q) = s−r THEN F := F+'T_q+'
        ELSE IF S(1+r+u(q)) ≠ 'u' THEN
        BEGIN F := F+'T_q'; bracket (r+u(q), s) END;
      q := max i: u(i) ≤ b END;
    IF (last character of F ≠ '+') THEN (delete last two char)
  UNTIL b = ∅;
  Replace last character of F by ')+'
END;
BEGIN u(∅) := ∅; F := '';
    bracket (l(i), u(j));
    delete last character of F
END.
```

EXAMPLE. $f = X^5 + Y^{14}$, $i = 2, j = 9$

$$F = '(T_7 + T_3(T_4(T_1) + T_1(T_4 + T_3(T_2) + T_2(T_3 + T_1(T_1)))))'$$

To finish this part of the algorithm we have to "solve" the brackets. In this example we get:

$$F = 'T_7 + T_3 T_4 T_1 + T_3 T_1 T_4 + T_3 T_1 T_3 T_2 + T_3 T_1 T_2 T_3 + T_3 T_1 T_2 T_1 T_1'$$

Now we have to compute the coefficients of the corresponding monomials. If c is the coefficient of the monomial $T_{q(1)} \cdot \ldots \cdot T_{q(s)}$ occurring in F, then $c = c_1 \cdot \ldots \cdot c_s$ and $c_1 = 1 - \deg n_{q(1)}/d$, c_i is one of the exponents of $n_{q(i)}$ divided by a or b depending on whether $\partial F/\partial X$, resp. $\partial F/\partial Y$ was involved in the reduction modulo ΔF in this step.

PART (III) OF THE ALGORITHM

Computing a coefficient

$c := (-1)^s u(q(1))/d$; $n := l(i)$;

```
FOR k := 2 TO s DO BEGIN
    e := e(n)+e(u(q(k-1))); e' := e'(n)+e'(u(q(k-1)));
    n := n+q(k-1);
    IF e ≥ a-1 THEN c := c*e(u(q(k)))/a;
    IF e' ≥ b-1 THEN c := c*e'(u(q(k)))/b
    END
```

EXAMPLE: $f = X^5 + Y^{14}$, $i = 2, j = 9$, $T_3 T_1 T_2 T_1 T_1$

$$c = -(\tfrac{4}{70})(\tfrac{6}{14})(\tfrac{2}{5})(\tfrac{6}{14})(\tfrac{3}{5}) = -\tfrac{108}{42875}$$

PART (IV) OF THE ALGORITHM

Finally, we have to order the variables in the monomials, to order the monomials in F lexicographically and to identify monomials of the same type by adding their coefficients.

EXAMPLE. $f = X^5 + Y^{14}$

$$\bar{c}_{2,9} = -13/70T_7 - 39/1225T_4T_3T_1 + 99/8575T_3^2T_2T_1 - 108/42875T_3T_2T_1^3$$

REMARK. The algorithm can be modified for more than two variables and without the restriction that the exponents are to be relatively prime. For instance, if $f = X_1^{a_1} + \cdots + X_n^{a_n}$, $d := a_1 \cdots a_n$, $b_i := d/a_i$ then choose u_i such that $u_i b_i \equiv 1 \bmod a_i$ and let $e_i(n) := nu_i$. If the a_i are not relatively prime add a small perturbation to each e_i and you will get similar results.

REMARK. The way of computing the $\bar{c}_{ij}$ by using the strings S and F allows one to compute fairly complicated examples on a 64K-computer in a short time (for instance, $\bar{c}_{57,157}$ of $X^{19} + Y^{29}$ in less than two seconds).

References

Laudal, O. A. (1979). *Formal moduli of algebraic structures, Lecture Notes in Math.* **754**.

Laudal, O. A., Pfister, G. (1983). The moduli problem: *Applications to the classification of isolated hypersurface singularities. Univ. of Oslo, Preprint Series No 11.*

Laudal, O. A., Martin, B., Pfister, G. (1986). Moduli of irreducible plane curve singularities with the semi-group $\langle a,b \rangle$, *Proc. Conf. on Algebraic Geometry*, (Berlin 1985), *Teubner-Texte zur Mathematik, Bd.92*, pp. 236–258. Leipzig: Teubner.

Teissier, B. (1976). *Appendice au cours de O. Zariski, Le problème des modules pour les branches planes.* Paris: Publ. du Centre de Mathématiques de l'Ecole Polytechnique.

Zariski, O. (1976). *Le problème des modules pour les branches planes.* Paris: Publ. du Centre de Mathématiques de l'Ecole Polytechnique.