J. Symbolic Computation (2001) **31**, 619–629
doi:10.1006/jsco.2001.0437
Available online at http://www.idealibrary.com on IDEAL®

A Direct Approach to Computing the μ-basis of Planar Rational Curves

JIANMIN ZHENG[†] AND THOMAS W. SEDERBERG[‡]

[†]*Department of Applied Mathematics, Zhejiang University, People's Republic of China*
[‡]*Department of Computer Science, Brigham Young University, U.S.A.*

This paper presents an $O(n^2)$ algorithm, based on Gröbner basis techniques, to compute the μ-basis of a degree n planar rational curve. The prior method involved solving a set of linear equations whose complexity by standard numerical methods was $O(n^3)$. The μ-basis is useful in computing the implicit equation of a parametric curve and can express the implicit equation in the form of a determinant that is smaller than that obtained by taking the resultant of the parametric equations.

1. Introduction

Consider a degree n planar rational curve $\mathbf{r}(t)$ in homogeneous form

$$(x, y, w) = (a(t), b(t), c(t)) \tag{1}$$

where $n = \max\{deg(a), deg(b), deg(c)\}$, and $a(t) = \sum_{i=0}^{n} a_i t^i$, $b(t) = \sum_{i=0}^{n} b_i t^i$, $c(t) = \sum_{i=0}^{n} c_i t^i$ $(\neq 0)$ are relatively prime. There always exists a non-zero homogeneous polynomial $f(x, y, w)$ for which $f(a(t), b(t), c(t))$ is identically zero. The equation $f(x, y, w) = 0$ is called the implicit equation of $\mathbf{r}(t)$, and (1) is called the parametric equation. The process of computing the implicit equation given the parametric equation is called *implicitization*, and the inverse process (i.e. computing the rational parametric equation of an implicitly defined curve) is known as *parameterization*. A plane implicit curve admits a rational parameterization if and only if its genus is zero. A few algorithms have been developed to check the rationality of an implicit curve, and to compute a rational parameterization if one exists (Abhyankar and Bajaj, 1988; Sendra and Winkler, 1991, 1998). Sendra and Winkler further studied the problem of computing an optimal parameterization (Sendra and Winkler, 1999).

The main algorithmic approaches to implicitization involve either resultants or Gröbner bases. The former is to take the resultant of

$$\rho_1 = c(t)x - a(t)w \qquad \text{and} \qquad \rho_2 = c(t)y - b(t)w \tag{2}$$

with respect to t (Goldman *et al.*, 1984). This yields $f(x, y, w)$ in the form of the determinant of an $n \times n$ matrix if Bezout's resultant is used, or a $2n \times 2n$ matrix if Sylvester's resultant is used (Sederberg *et al.*, 1997). The latter considers the ideal $\langle c(t)x - a(t), c(t)y - b(t)\rangle \subset K[x, y, t]$ where K is a computable field of characteristic zero. The Rational Implicitization Theorem (Cox *et al.*, 1992) states that if $GCD(a, b, c) = 1$, then the variety of $\langle cx - a, cy - b\rangle \cap K[x, y]$ is the smallest variety in K^2 containing

0747–7171/01/050619 + 11 $35.00/0

the parametric curve. Thus Gröbner bases can be used to implicitize a curve. Extensions based on Gröbner basis techniques have also been proposed (Gao and Chou, 1992; Alonso *et al.*, 1995; Gonzalez-Vega, 1997). In general, resultant-based methods are more efficient than Gröbner basis methods from a computational point of view, but Gröbner basis methods provide more insights, and deal with curves and surfaces with base points.

A recent development in the problem of implicitizing planar rational parametric curves is the "moving curve" method (Sederberg and Chen, 1995; Sederberg *et al.*, 1994, 1997; Cox *et al.*, 1998b). A moving curve is defined as

$$C(x, y, w; t) := \sum_{i=0}^{m} f_i(x, y, w)t^i,$$

where $f_i(x, y, w)$ is a homogeneous polynomial of degree d. Thus $C(x, y, w; t) = 0$ is a family of algebraic curves, with one curve corresponding to each t. When $d = 1$, $C(x, y, w; t) = 0$ is a family of lines and is called a *moving line* of degree m. Likewise, $C(x, y, w; t) = 0$ is called a *moving conic* of degree m when $d = 2$. A moving curve $C(x, y, w; t) = 0$ is said to *follow* a planar rational curve $\mathbf{r}(t) = (x(t), y(t), w(t))$ if $C(x(t), y(t), w(t); t)$ is identically zero, that is, if for all values of t, the point $\mathbf{r}(t)$ lies on the moving curve $C(x, y, w; t) = 0$. Each row of Bezout's matrix or Sylvester's matrix corresponds to a moving line following the curve.

It has been proven (Cox *et al.*, 1998b) that for any planar rational curve $\mathbf{r}(t)$, there exist two moving lines p and q of degree $\mu(\leq n/2)$ and $n - \mu$ respectively, which follow the curve $\mathbf{r}(t)$ and satisfy:

1. p has the lowest degree in t among the moving lines following the curve $\mathbf{r}(t)$.
2. Any moving line $A(t)x + B(t)y + C(t)w = 0$ that follows the curve $\mathbf{r}(t)$ can be written in the form
$$Ax + By + Cw = h_1(t)p + h_2(t)q$$
where h_1 and h_2 are polynomials in t.

Two such moving lines p and q are referred to as a *μ-basis* of the curve $\mathbf{r}(t)$, because they generate the ideal $\langle \rho_1, \rho_2 \rangle$ (that is, $\langle p, q \rangle = \langle \rho_1, \rho_2 \rangle$).

The μ-basis is very useful for performing implicitization. Using a variant of Bezout's resultant, the implicit equation of $\mathbf{r}(t)$ can be written as the determinant of an $(n - \mu) \times (n - \mu)$ matrix with μ rows whose elements are quadratic in x, y and w, and the remaining $n - 2\mu$ rows with elements that are linear in x, y and w. In the generic case, $\mu = \lfloor n/2 \rfloor$, and the determinant has dimension $\frac{n}{2} \times \frac{n}{2}$ if n is even and $\frac{n+1}{2} \times \frac{n+1}{2}$ if n is odd—a significant improvement over the $n \times n$ matrix generated by conventional implicitization methods (De Montaudouin and Tiller, 1984).

A μ-basis has practical value in addition to performing curve implicitization. For example, it can help reveal the singular locus of the rational curve, and the parametric equation is easily found from the μ-basis by doing a single cross product (Cox *et al.*, 1998b).

The previous known algorithm for computing the μ-basis of a curve is the method in Cox *et al.* (1998b) which proceeds in the following manner.

Observe that a degree m moving line

$$C(x, y, w; t) = \sum_{i=0}^{m} (A_i x + B_i y + C_i w)t^i = 0 \tag{3}$$

follows a rational curve (1) if

$$\sum_{i=0}^{m}(A_i a(t) + B_i b(t) + C_i c(t))t^i \equiv 0$$

which can be expressed as $Mv = 0$, where M is the $(n+m+1) \times (3m+3)$ matrix

$$M = \begin{bmatrix} a_0 & b_0 & c_0 & 0 & \cdots & 0 & 0 & 0 \\ a_1 & b_1 & c_1 & a_0 & \cdots & 0 & 0 & 0 \\ a_2 & b_2 & c_2 & a_1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_n & b_n & c_n & a_{n-1} & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & a_n & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & a_{n-1} & b_{n-1} & c_{n-1} \\ 0 & 0 & 0 & 0 & \cdots & a_n & b_n & c_n \end{bmatrix} \tag{4}$$

and

$$v = [A_0 \;\; B_0 \;\; C_0 \;\; \cdots \;\; A_m \;\; B_m \;\; C_m]^T.$$

Solving $Mv = 0$ for v yields a degree m moving line (3) that follows the curve.

In the generic case, $\mu = \lfloor \frac{n}{2} \rfloor$. It is easy to determine if the generic case holds by simply checking how many solutions $Mv = 0$ has if we choose $m = \lfloor \frac{n}{2} \rfloor - 1$. If $Mv = 0$ has no solution, then it is true that $\mu = \lfloor \frac{n}{2} \rfloor$. Otherwise, μ can be determined by computing $Rank(M)$ when $m = \lfloor \frac{n}{2} \rfloor - 1$ as follows. Let N_m be the number of the linearly independent solutions of $Mv = 0$, where $N_m = 3m + 3 - Rank(M)$. As is shown in Cox *et al.* (1998b), $\mu \leq \lfloor \frac{n}{2} \rfloor$, and

$$N_m = \begin{cases} 0, & 0 \leq m < \mu \\ m - \mu + 1, & \mu \leq m < n - \mu - 1 \\ 2m + 2 - \mu, & n - \mu - 1 \leq m. \end{cases} \tag{5}$$

Since we have chosen $m = \lfloor \frac{n}{2} \rfloor - 1$, we have $\mu \leq m < n - \mu - 1$ (i.e. $N_m = m - \mu + 1$) or $m = \mu - 1 < \mu$ (i.e. $N_m = 0$) corresponding to $\mu < \lfloor \frac{n}{2} \rfloor$ or $\mu = \lfloor \frac{n}{2} \rfloor$, respectively. Therefore $\mu = m + 1 - N_m$ holds for both cases.

Once the value of μ is obtained, we can solve $Mv = 0$ with $m = \mu$ for a moving line p. The moving line q can be obtained by solving the equation with $m = n - \mu$ and choosing a solution which is not of the form $h(t)p$ where $h(t)$ is a polynomial in t.

In this paper, we present a new algorithm for computing a μ-basis that is more straightforward and faster than the method just reviewed. Section 2 provides the relevant mathematical background. The new theory and method are developed in Section 3, along with a numerical example. Section 4 analyzes the computational complexity, concluding that the new algorithm is $O(n^2)$ while the previous method is $O(n^3)$.

2. Preliminary

We assume that the coefficients of the polynomials defining the rational curve belong to Q, the field of rational numbers. Let $Q[t]^3$ be the set of 3-dimensional row vectors with entries in the polynomial ring $Q[t]$. $Q[t]^3$ is a module over $Q[t]$ (Cox *et al.*, 1998a). Denote the standard basis vectors in $Q[t]^3$ by

$$E_1 = (1, 0, 0), \qquad E_2 = (0, 1, 0), \qquad E_3 = (0, 0, 1).$$

Then any element $f \in Q[t]^3$ can be written

$$f = f_1(t)E_1 + f_2(t)E_2 + f_3(t)E_3$$

with $f_i(t) \in Q[t]$. Furthermore,

$$f = \sum_{i=0}^{deg(f_1)} e_{1,i}t^iE_1 + \sum_{i=0}^{deg(f_2)} e_{2,i}t^iE_2 + \sum_{i=0}^{deg(f_3)} e_{3,i}t^iE_3$$

where $e_{k,i} \in Q$ and $e_{k,deg(f_k)} \neq 0$. The element t^iE_k is a *monomial* in $Q[t]^3$.

Now we define an ordering relation $>_M$ on the monomials of $Q[t]^3$: we say $t^iE_j >_M t^kE_l$ if $i > k$, or if $i = k$ and $j < l$. This order sorts the monomials first by the degrees, and then breaks ties using the position within the vector in $Q[t]^3$. It is easy to show that (Cox *et al.*, 1998a):

1. $>_M$ is a total ordering relation, which means the terms appearing within any $f \in Q[t]^3$ can be uniquely listed in increasing or decreasing order under $>_M$.
2. if $t^iE_j >_M t^kE_l$, then $t^{i+\alpha}E_j >_M t^{k+\alpha}E_l$ for any non-negative integer α.
3. $>_M$ is well-ordering, i.e. every non-empty collection of monomials has a smallest element under $>_M$.

Once we have the ordering $>_M$ on the monomials of $Q[t]^3$, we can express each $f \in Q[t]^3$ as a sum of monomials m_i

$$f = \sum_{i=1}^{l} e_i m_i$$

with $e_i \neq 0$ and $m_1 >_M m_2 >_M \cdots >_M m_l$. We define the *leading coefficient*, *leading monomial*, and *leading term* of f:

$$LC(f) = e_1, \qquad LM(f) = m_1, \qquad LT(f) = e_1m_1.$$

The leading monomial m_1 is of the form t^dE_k. We say the degree of f is d, and the leading term *contains* the standard basis vector E_k or the leading term is located in the kth component of the vector in $Q[t]^3$. Obviously, $deg(f) = \max\{deg(f_1), deg(f_2), deg(f_3)\}$.

3. Algorithm for Computing a μ-basis

This section presents a new algorithm for computing the μ-basis of a parametric curve. The algorithm is in the spirit of Buchberger's algorithm for computing the Gröbner basis of a polynomial ideal. However, unlike Buchberger's algorithm which adds remainders of S-polynomials, our algorithm never needs to keep track of more than three generators for the module. This is the key to the efficiency of our algorithm.

3.1. GENERATORS FOR MOVING LINES

Consider the module $P = \{Ax + By + Cw : A(t), B(t), C(t) \in Q[t]\}$ over $Q[t]$. Since the map $\varphi : Ax + By + Cw \to (A, B, C)$ from the set P to $Q[t]^3$ is an *isomorphism*, in the following we use the triple (A, B, C) or $Ax + By + Cw$ for moving lines without distinction. When we refer to "the moving line (A, B, C)", we mean the line defined by the equation $Ax + By + Cw = 0$.

For a planar rational curve $\mathbf{r}(t) = (a(t), b(t), c(t))$, let

$$M = \{(A(t), B(t), C(t)) : A(t)a(t)+B(t)b(t)+C(t)c(t) \equiv 0, A(t), B(t), C(t) \in Q[t]\}. \quad (6)$$

Geometrically, M consists of all moving lines that follow the curve $\mathbf{r}(t)$. Given the equation of a parametric curve, we have three trivial moving lines that can serve as the generators of M. In fact,

PROPOSITION 1. *Let $\mathbf{r}(t)$ be a planar rational curve defined by (1), and let M be the set of all moving lines that follow $\mathbf{r}(t)$. Then M can be generated by $(b, -a, 0)$, $(c, 0, -a)$ and $(0, c, -b)$.*

PROOF. It is obvious that moving lines $(b, -a, 0)$, $(c, 0, -a)$ and $(0, c, -b)$ follow the curve $\mathbf{r}(t)$. If an additional moving line $(A(t), B(t), C(t))$ follows the curve, then

$$A(t)a(t) + B(t)b(t) + C(t)c(t) \equiv 0.$$

Since $a(t)$, $b(t)$ and $c(t)$ are relatively prime, there exist three polynomials $k_1(t)$, $k_2(t)$, $k_3(t) \in Q[t]$ such that $k_1a + k_2b + k_3c = 1$. Therefore

$$\begin{aligned}(A, B, C) &= (k_1a + k_2b + k_3c)(A, B, C) = k_1(Aa, Ba, Ca) \\ &\quad +k_2(Ab, Bb, Cb) + k_3(Ac, Bc, Cc) \\ &= k_1(-Bb - Cc, Ba, Ca) + k_2(Ab, -Aa - Cc, Cb) + k_3(Ac, Bc, -Aa - Bb) \\ &= (k_2A - k_1B)(b, -a, 0) + (k_3A - k_1C)(c, 0, -a) + (k_3B - k_2C)(0, c, -b). \ \square\end{aligned}$$

A set of moving lines L_i, $i = 1, \ldots, \lambda$ is said to be *linearly dependent over $Q[t]$* if there exist polynomials $h_i(t) \in Q[t]$ (not all zero) such that $\sum_{i=1}^{\lambda} h_i(t)L_i \equiv 0$. The generators $(b, -a, 0)$, $(c, 0, -a)$ and $(0, c, -b)$ are linearly dependent over $Q[t]$, because we can choose $h_1 = c$, $h_2 = -b$ and $h_3 = a$. In addition, it can be shown that two of $(b, -a, 0)$, $(c, 0, -a)$ and $(0, c, -b)$ have leading terms containing the same standard basis vector. Actually, we can make a more general statement:

PROPOSITION 2. *Let $p(t) = (p_1, p_2, p_3)$, $q(t) = (q_1, q_2, q_3)$, $g(t) = (g_1, g_2, g_3) \in Q[t]^3$ be linearly dependent over $Q[t]$. Then at least two of the three triples p, q, and g have leading terms that contain the same standard basis vector.*

PROOF. Suppose the leading terms of $p(t)$, $q(t)$, and $g(t)$ contain different basis vectors. Without loss of generality, we can assume they contain E_1, E_2, and E_3 respectively. Then for any polynomial $h(t) \in Q[t]$, we have

$$\begin{aligned} deg(hp_1) &\geq deg(hp_2), deg(hp_3); \\ deg(hq_1) &< deg(hq_2) \geq deg(hq_3); \\ deg(hg_1), deg(hg_2) &< deg(hg_3). \end{aligned} \quad (7)$$

Since $p(t)$, $q(t)$, and $g(t)$ are linearly dependent over $Q[t]$, there exist three polynomials $h_1(t)$, $h_2(t)$, $h_3(t) \in Q[t]$ such that

$$h_1(t)(p_1, p_2, p_3) + h_2(t)(q_1, q_2, q_3) + h_3(t)(g_1, g_2, g_3) \equiv 0. \quad (8)$$

Now consider the first component. We try to derive a contradiction. From $h_1p_1 + h_2q_1 + h_3g_1 = 0$, we know that among the three polynomials h_1p_1, h_2q_1 and h_3g_1, at least two

of them have the same degree. The degree of the third is less than or equal to the degree of the other two. Thus there are only four cases:

Case 1 $deg(h_1p_1) < deg(h_2q_1) = deg(h_3g_1)$.
From (7), $deg(h_1p_2) \le deg(h_1p_1) < deg(h_2q_1) < deg(h_2q_2)$. Consider the second component of the equation (8). As discussed above for the first component, $deg(h_2q_2)$ must equal $deg(h_3g_2)$ from equation $h_1p_2 + h_2q_2 + h_3g_2 = 0$.
Similarly, $deg(h_1p_3) \le deg(h_1p_1) < deg(h_3g_1) < deg(h_3g_3)$, so $deg(h_2q_3) = deg(h_3g_3)$ by analyzing the third component of (8).
Thus we obtain $deg(h_3g_3) > deg(h_3g_2) = deg(h_2q_2) \ge deg(h_2q_3) = deg(h_3g_3)$, a contradiction.

Case 2 $deg(h_2q_1) < deg(h_1p_1) = deg(h_3g_1)$.
In this case, we have $deg(h_3g_3) > deg(h_3g_1) = deg(h_1p_1) \ge deg(h_1p_3)$. Therefore $deg(h_3g_3) = deg(h_2q_3)$. This leads to $deg(h_2q_2) \ge deg(h_2q_3) = deg(h_3g_3) > deg(h_3g_2)$. Thus the equation $h_1p_2+h_2q_2+h_3g_2 = 0$ implies $deg(h_2q_2) = deg(h_1p_2)$. Combining these results, we have $deg(h_3g_1) = deg(h_1p_1) \ge deg(h_1p_2) = deg(h_2q_2) \ge deg(h_2q_3) = deg(h_3g_3) > deg(h_3g_1)$, which cannot be true.

Case 3 $deg(h_3g_1) < deg(h_1p_1) = deg(h_2q_1)$.
In a similar fashion we can obtain

$$\begin{aligned} deg(h_3g_2) = deg(h_2q_2) > deg(h_2q_1) = deg(h_1p_1) \ge deg(h_1p_3) \\ = deg(h_3g_3) > deg(h_3g_2) \end{aligned}$$

which is impossible.

Case 4 $deg(h_1p_1) = deg(h_2q_1) = deg(h_3g_1)$.
Likewise, we can obtain in this case $deg(h_3g_2) = deg(h_2q_2) > deg(h_2q_1) = deg(h_1p_1) \ge deg(h_1p_3) = deg(h_3g_3) > deg(h_3g_2)$. This cannot occur.

Therefore the assumption does not hold and the proposition is proved. □

3.2. ALGORITHM

Starting with the three moving lines $(b, -a, 0)$, $(c, 0, -a)$, and $(0, c, -b)$, we now devise an algorithm to produce two generators for M with lowest degree. First we outline the algorithm.

Input: (a, b, c)—the parametric equation of a planar rational curve $\mathbf{r}(t)$
Output: two moving lines p and q that form a μ-basis of $\mathbf{r}(t)$
Procedure:

step 1. (initialize) Set $v_1 = (b, -a, 0)$, $v_2 = (c, 0, -a)$, $v_3 = (0, c, -b)$.
step 2. The coefficients of a, b, and c are rational numbers. Step 4 requires that those coefficients are integers. Therefore, multiply v_1, v_2, and v_3 by the least common multiple of the denominators of the coefficients of a, b, and c. Then set $S = \{v_1, v_2, v_3\}$.
step 3. Choose v_i, v_j from S so that $LT(v_i)$ and $LT(v_j)$ contain the same basis vector, and $deg(v_i) \ge deg(v_j)$.
step 4. Replace v_i with

$$v_i \leftarrow \frac{LCM(LC(v_i), LC(v_j))}{LC(v_i)} v_i - \frac{LCM(LC(v_i), LC(v_j))}{LC(v_j)} t^{deg(v_i)-deg(v_j)} v_j \tag{9}$$

where $LCM(i, j)$ is the least common multiple of i and j.

step 5. If $v_i = 0$, remove v_i from S.
step 6. If the leading term of each non-zero element in S has a different basis vector, then output S;
else goto step 3.

The algorithm works based on the following observations:
1. The set S contains at most three moving lines at any time.
2. Each replacement can be written

$$v_i^{(l+1)} = d_1 v_i^{(l)} + d_2 t^\alpha v_j^{(l)}, \qquad v_j^{(l+1)} = v_j^{(l)}, \qquad v_k^{(l+1)} = v_k^{(l)}$$

where $\alpha(\geq 0)$, $d_1(\neq 0)$ and d_2 are integers, and l denotes the iteration number. Therefore when $v_i^{(l+1)} \neq 0$, the elements $v_i^{(l+1)}$, $v_j^{(l+1)}$ and $v_k^{(l+1)}$ generate M since $v_i^{(l)}$, $v_j^{(l)}$ and $v_k^{(l)}$ also generate M. If $v_i^{(l)}$, $v_j^{(l)}$ and $v_k^{(l)}$ are linearly dependent over $Q[t]$, then so are $v_i^{(l+1)}$, $v_j^{(l+1)}$ and $v_k^{(l+1)}$. When $v_i^{(l+1)} = 0$, then $v_j^{(l+1)}$ and $v_k^{(l+1)}$ generate M if $v_i^{(l)}$, $v_j^{(l)}$ and $v_k^{(l)}$ are a generator set. Since the initial three moving lines serve as a generating set for M and are linearly dependent over $Q[t]$, we know by induction that at each stage S is a generating set for all moving lines following the curve, and as long as there are three elements in S, they are also linearly dependent over $Q[t]$.

3. The algorithm terminates after a finite number of steps, because each update lowers the ordering of the leading term of one moving line.

4. At the final stage, S contains only two elements. S cannot contain three elements when the algorithm terminates because in that case, those three elements must be linearly dependent over $Q[t]$. But by Proposition 2, at least two of them have leading terms containing the same basis vector and therefore the algorithm will continue to iterate.

On the other hand, S cannot consist of a single element (g_1, g_2, g_3) because since that one element must generate M, there exist three non-zero polynomials $h_1(t)$, $h_2(t)$, $h_3(t) \in Q[t]$ such that

$$(b, -a, 0) = h_1(t)(g_1, g_2, g_3), \qquad (c, 0, -a) = h_2(t)(g_1, g_2, g_3),$$
$$(0, c, -b) = h_3(t)(g_1, g_2, g_3).$$

This leads to $g_1 = g_2 = g_3 = 0$, a contradiction.

3.3. THE OUTPUTS ARE THE μ-BASIS

Section 3.2 shows that the algorithm finally outputs two elements. Denote those two elements by $p(t)$ and $q(t)$ with $deg(p) \leq deg(q)$. Since p and q generate M, any moving line following the curve $\mathbf{r}(t)$ can be generated by $p(t)$ and $q(t)$. Now we prove that $p(t)$ and $q(t)$ are a μ-basis. This can be shown by the following two propositions.

PROPOSITION 3. *The moving line $p(t)$ has the lowest degree in t.*

PROOF. By the construction of p and q, we know that the leading terms of p and q contain different basis vectors. Denote these basis vectors by E_i and E_j $(i \neq j)$. Without loss of generality, we assume that $i > j$. For any moving line $L = (l_1, l_2, l_3)$ following the curve $\mathbf{r}(t)$, there exist two polynomials $h_1(t), h_2(t) \in Q[t]$ such that $L = h_1 p + h_2 q$, i.e. $(l_1, l_2, l_3) = h_1(p_1, p_2, p_3) + h_2(q_1, q_2, q_3)$. Thus if $deg(h_1 p_j) \neq deg(h_2 q_j)$, then

$$deg(L) \geq deg(l_j) \geq \max\{deg(h_1 p_j), deg(h_2 q_j)\} \geq deg(q_j) = deg(q) \geq deg(p).$$

Otherwise, if $deg(h_1 p_j) = deg(h_2 q_j)$, then

$$deg(h_1 p_i) > deg(h_1 p_j) = deg(h_2 q_j) \geq deg(h_2 q_i)$$

and thus $deg(L) \geq deg(l_i) = deg(h_1 p_i) \geq deg(p_i) = deg(p)$. Therefore the degree of L is never less than the degree of p in either case. This completes the proof. □

PROPOSITION 4. *The sum of $deg(p)$ and $deg(q)$ is equal to the degree of the curve $\mathbf{r}(t)$.*

PROOF. Let $deg(p) = \mu$ and $deg(q) = \eta$. Then $\mu \leq \eta$.

First, $(b, -a, 0)$, $(c, 0, -a)$ and $(0, c, -b)$ are moving lines following the curve, so they can be represented by p and q. Thus

$$(b, -a, 0) = h_1 p + h_2 q, \tag{10}$$

$$(c, 0, -a) = h_3 p + h_4 q, \tag{11}$$

$$(0, c, -b) = h_5 p + h_6 q \tag{12}$$

with some polynomials $h_i(t) \in Q[t]$. Taking the cross product of (10) and (11) gives

$$a(a, b, c) = (h_1 h_4 - h_2 h_3) p \times q.$$

Similarly,

$$b(a, b, c) = (h_1 h_6 - h_2 h_5) p \times q,$$
$$c(a, b, c) = (h_3 h_6 - h_4 h_5) p \times q.$$

Since a, b and c are relatively prime, there exist three polynomials $k_1(t)$, $k_2(t)$, $k_3(t) \in Q[t]$ satisfying $k_1 a + k_2 b + k_3 c = 1$. Hence

$$(a, b, c) = [k_1(h_1 h_4 - h_2 h_3) + k_2(h_1 h_6 - h_2 h_5) + k_3(h_3 h_6 - h_4 h_5)](p \times q). \tag{13}$$

Recall that $LT(p)$ and $LT(q)$ contain different basis vectors. Thus $deg(p \times q) = deg(p) + deg(q) = \mu + \eta$. The degree of the expression on the left side of equation (13) is n, while the degree on the right side of equation (13) is at least $deg(p \times q) = \mu + \eta$. This gives $\eta \leq n - \mu$.

Second, by Proposition 1, $(b, -a, 0)$, $(c, 0, -a)$ and $(0, c, -b)$ also serve as a generating set. Therefore we can find polynomials $d_i(t) \in Q[t]$, $i = 1, \ldots, 6$ such that

$$p = d_1(t)(b, -a, 0) + d_2(t)(c, 0, -a) + d_3(t)(0, c, -b),$$
$$q = d_4(t)(b, -a, 0) + d_5(t)(c, 0, -a) + d_6(t)(0, c, -b).$$

Taking the cross product we arrive at

$$\begin{aligned} p \times q &= d_1 d_5 a(a, b, c) + d_1 d_6 b(a, b, c) - d_2 d_4 a(a, b, c) \\ &\quad + d_2 d_6 c(a, b, c) - d_3 d_4 b(a, b, c) - d_3 d_5 c(a, b, c) \\ &= [(d_1 d_5 - d_2 d_4)a + (d_1 d_6 - d_3 d_4)b + (d_2 d_6 - d_3 d_5)c](a, b, c). \end{aligned}$$

Hence we have the degree estimation:

$$\mu + \eta = deg(p) + deg(q) = deg(p \times q) \geq deg((a, b, c)) = n$$

i.e. $\eta \geq n - \mu$.

Therefore we obtain $deg(q) = \eta = n - \mu$. □

3.4. NUMERICAL EXAMPLE

We now present an example to illustrate how the algorithm works.

EXAMPLE. A planar rational curve is defined by $(x, y, w) = (2t^2+4t+5, 3t^2+t+4, t^2+2t+3)$. We apply the algorithm to this curve. The elements in S at each iteration are listed below:

$$\begin{cases} v_1^{(0)} &=& (3t^2+t+4, -2t^2-4t-5, 0) \\ v_2^{(0)} &=& (t^2+2t+3, 0, -2t^2-4t-5) \\ v_3^{(0)} &=& (0, t^2+2t+3, -3t^2-t-4) \end{cases}$$

$$\Longrightarrow \begin{cases} v_1^{(1)} &=& v_1^{(0)} - 3v_2^{(0)} &=& (-5t-5, -2t^2-4t-5, 6t^2+12t+15) \\ v_2^{(1)} &=& v_2^{(0)} &=& (t^2+2t+3, 0, -2t^2-4t-5) \\ v_3^{(1)} &=& v_3^{(0)} &=& (0, t^2+2t+3, -3t^2-t-4) \end{cases}$$

$$\Longrightarrow \begin{cases} v_1^{(2)} &=& v_1^{(1)} + 2v_3^{(1)} &=& (-5t-5, 1, 10t+7) \\ v_2^{(2)} &=& v_2^{(1)} &=& (t^2+2t+3, 0, -2t^2-4t-5) \\ v_3^{(2)} &=& v_3^{(1)} &=& (0, t^2+2t+3, -3t^2-t-4) \end{cases}$$

$$\Longrightarrow \begin{cases} v_1^{(3)} &=& v_1^{(2)} &=& (-5t-5, 1, 10t+7) \\ v_2^{(3)} &=& 5v_2^{(2)} + tv_1^{(2)} &=& (5t+15, t, -13t-25) \\ v_3^{(3)} &=& v_3^{(2)} &=& (0, t^2+2t+3, -3t^2-t-4) \end{cases}$$

$$\Longrightarrow \begin{cases} v_1^{(4)} &=& v_1^{(3)} &=& (-5t-5, 1, 10t+7) \\ v_2^{(4)} &=& v_2^{(3)} + v_1^{(3)} &=& (10, t+1, -3t-18) \\ v_3^{(4)} &=& v_3^{(3)} &=& (0, t^2+2t+3, -3t^2-t-4) \end{cases}$$

$$\Longrightarrow \begin{cases} v_1^{(5)} &=& v_1^{(4)} &=& (-5t-5, 1, 10t+7) \\ v_2^{(5)} &=& v_2^{(4)} &=& (10, t+1, -3t-18) \\ v_3^{(5)} &=& v_3^{(4)} - tv_2^{(4)} &=& (-10t, t+3, 17t-4) \end{cases}$$

$$\Longrightarrow \begin{cases} v_1^{(6)} &=& v_1^{(5)} &=& (-5t-5, 1, 10t+7) \\ v_2^{(6)} &=& v_2^{(5)} &=& (10, t+1, -3t-18) \\ v_3^{(6)} &=& -v_3^{(5)} + 2v_1^{(5)} &=& (-10, -t-1, 3t+18) \end{cases}$$

$$\Longrightarrow \begin{cases} v_1^{(7)} &=& v_1^{(6)} &=& (-5t-5, 1, 10t+7) \\ v_2^{(7)} &=& v_2^{(6)} &=& (10, t+1, -3t-18) \\ v_3^{(7)} &=& v_3^{(6)} + v_2^{(6)} &=& (0, 0, 0) \qquad \text{(to be removed from } S\text{)}. \end{cases}$$

Thus we obtain the μ-basis in Cartesian coordinates:

$$p(t) = (-5t-5)x + y + (10t+7) = (-5x+10)t + (-5x+y+7)$$
$$q(t) = 10x + (t+1)y + (-3t-18) = (y-3)t + (10x+y-18).$$

The implicit equation is thus the resultant of $p(t)$ and $q(t)$ with respect to t:

$$(-5x+10)(10x+y-18)-(-5x+y+7)(y-3)=-50x^2+175x-y^2+6y-159=0.$$

4. Computational Complexity

We now discuss the performance of the algorithm. Assume the curve $\mathbf{r}(t)=(a,b,c)$ has degree n. The algorithm starts with moving lines $v_1^{(0)}=(b,-a,0)$, $v_2^{(0)}=(c,0,-a)$ and $v_3^{(0)}=(0,c,-b)$, and eventually yields a μ-basis, i.e. a moving line p of degree μ and a moving line q of degree $n-\mu$. Without loss of generality, we assume that $v_1^{(k)}=p$, $v_2^{(k)}=q$ and $v_3^{(k)}=0$ after k iterations. First let us derive an upper bound on k, the number of iterations. Note that each iteration performs one update (9), and each update operation lowers the ordering of the leading term of a certain v_i. For each v_i, to decrease its degree by one, at most three updates are required. Since p has degree μ, $v_1^{(k)}$ is obtained from $v_1^{(0)}$ by performing at most $3(n-\mu)+2$ update operations, where number "2" is counted for the case where the leading terms of $v_i^{(0)}$ and $v_i^{(k)}$ are located in the first and third components, respectively. In the same way, $v_2^{(k)}$ comes from $v_2^{(0)}$ through at most $3\mu+2$ operations. However, the leading terms of p and q lie in the different components within the vector in $Q[t]^3$, so at most $3n+3(=(3(n-\mu)+2)+(3\mu+2)-1)$ updates are needed to produce $v_1^{(k)}$ and $v_2^{(k)}$. For $v_3^{(k)}$, suppose it comes from $v_3^{(j)}$ by performing one update operation. Then $v_3^{(j)}$ is of degree at least μ since μ is the lowest degree of any moving line following the curve. Also notice that the first component of $v_3^{(0)}$ is 0. Therefore, $v_3^{(k)}$ can be obtained from $v_3^{(0)}$ by at most $3(n-\mu)+2$ updates. In consequence, we have $k\le 6n-3\mu+5(=3n+3+3(n-\mu)+2)$.

Next we estimate the number of arithmetic operations required for producing $v_i^{(k)}$ which is obtained from $v_i^{(0)}$ with λ updates. If a moving line $v_i^{(j)}$ has N coefficients, the update (9) requires at most $2(N-1)$ multiplications and $N-1$ additions, producing a new element with at most $N-1$ coefficients. The initial moving line $v_i^{(0)}$ can be considered to have at most $3(n+1)$ coefficients (actually at most $2(n+1)$ since one component is zero). Then the λ updates need

$$\sum_{i=1}^{\lambda}2(3(n+1)-i-1)=2\lambda\left(3n+2-\frac{\lambda+1}{2}\right)$$

multiplications and $\lambda\big(3n+2-\frac{\lambda+1}{2}\big)$ additions. Applying these formulas to $v_1^{(k)}$, $v_2^{(k)}$ and $v_3^{(k)}$ and summing them up, we find that the k iterations need at most $18n^2+30n+18n\mu+6+3\mu(1-9\mu)$ multiplications and $9n^2+15n+9n\mu+3+\frac{3\mu(1-9\mu)}{2}$ additions. Therefore the computational complexity of the proposed algorithm is $O(n^2)$.

By comparison, we briefly consider the previously published method of computing a μ-basis by solving linear equations. The coefficient matrix of the equations for a degree m moving line has dimension $(n+m+1)\times(3m+3)$ (see (4)). Solving this set of equations using Gaussian elimination is $O(n^3)$ (Kronsjö, 1987) for the generic case of $\mu=\lfloor\frac{n}{2}\rfloor$.

Our analysis shows that this new method is straightforward and needs much less computation than the previous method. This leads to a simpler algorithm for performing implicitization of curves and for solving other problems related to the μ-basis.

Acknowledgements

The authors were supported in part by NSF grant number CCR-9712407. Jianmin Zheng is also supported by the National Natural Science Foundation of China (69973042) and Foundation of State Key Basic Research 973 Development Programming Item (G1998030600). David Cox provided the initial idea of studying the μ-basis in terms of modules.

References

Abhyankar, S., Bajaj, C. (1988). Automatic parameterization of rational curves and surfaces III: algebraic plane curves. *Comput.-Aided Geom. Des.*, **5**, 309–321.

Alonso, C., Gutierrez, J., Recio, T. (1995). An implicitization algorithm with fewer variables. *Comput.-Aided Geom. Des.*, **12**, 251–258.

Cox, D., Little, J., O'Shea, D. (1992). *Ideals, Varieties and Algorithms*, 2nd edn, 1996, Berlin, Springer.

Cox, D., Little, J., O'Shea, D. (1998a). *Using Algebraic Geometry*. Berlin, Springer.

Cox, D., Sederberg, T., Chen, F. (1998b). The moving line ideal basis of planar rational curves. *Comput.-Aided Geom. Des.*, **15**, 803–827.

De Montaudouin, Y., Tiller, W. (1984). The Cayley method in computer aided geometric design. *Comput.-Aided Geom. Des.*, **1**, 309–326.

Gao, X., Chou, S. (1992). Implicitization of rational parametric equations. *J. Symbolic Computation*, **14**, 459–470.

Goldman, R., Sederberg, T., Anderson, D. (1984). Vector elimination: a technique for the implicitization, inversion and intersection of planar parametric rational polynomial curves. *Comput.-Aided Geom. Des.*, **1**, 327–356.

Gonzalez-Vega, L. (1997). Implicitization of parametric curves and surfaces by using multidimensional Newton formulae. *J. Symbolic Computation*, **23**, 137–151.

Kronsjö, L. (1987). *Algorithms: Their Complexity and Efficiency*, 2nd edn, New York, Wiley-Interscience.

Sederberg, T., Chen, F. (1995). Implicitization using moving curves and surfaces. In *SIGGRAPH 95 Conference Proceedings, Annual Conference Series*, pp. 301–308. Reading, MA, Addison Wesley.

Sederberg, T., Goldman, R., Du, H. (1997). Implicitizing rational curves by the method of moving algebraic curves. *J. Symbolic Computation*, **23**, 153–175.

Sederberg, T., Saito, T., Qi, D., Klimaszewski, K. (1994). Curve implicitization using moving lines. *Comput.-Aided Geom. Des.*, **11**, 687–706.

Sendra, J., Winkler, F. (1991). Symbolic parametrization of curves. *J. Symbolic Computation*, **6**, 607–632.

Sendra, J., Winkler, F. (1998). Real parametrization of algebraic curves, *Artificial Intelligence and Symbolic Computation*, pp. 284–295. Plattsburgh, NY.

Sendra, J., Winkler, F. (1999). Algorithms for rational real algebraic curves. *Fundam. Inform.*, **39**, 211–228.

Originally Received 18 February 2000
Accepted 18 January 2001
Published electronically 20 March 2001