Monomial Bases and Polynomial System Solving

(Extended Abstract)

Ioannis Z. Emiris * Ashutosh Rege*

Abstract

This paper addresses the problem of efficient construction of monomial bases for the coordinate rings of zero-dimensional varieties. Existing approaches rely on Gröbner bases methods – in contrast, we make use of recent developments in sparse elimination techniques which allow us to strongly exploit the structural sparseness of the problem at hand. This is done by establishing certain properties of a matrix formula for the sparse resultant of the given polynomial system. We use this matrix construction to give a simpler proof of the result of Pedersen and Sturmfels [22] for constructing monomial bases. The monomial bases so obtained enable the efficient generation of multiplication maps in coordinate rings and provide a method for computing the common roots of a generic system of polynomial equations with complexity singly exponential in the number of variables and polynomial in the number of roots. We describe the implementations based on our algorithms and provide empirical results on the well-known problem of cyclic n-roots; our implementation gives the first known upper bounds in the case of $n = 10$ and $n = 11$. We also present some preliminary results on root finding for the Stewart platform and motion from point matches problems in robotics and vision respectively.

1 Introduction

The study of coordinate rings of varieties in K^n, where K is a (algebraically closed) field, has been shown to be particularly useful in studying systems of polynomial equations. For a 0-dimensional variety it is well known that the coordinate ring forms a finite-dimensional vector space (in fact an algebra) over K. An important algorithmic question is the construction of an explicit monomial K-basis for such a space. Among other things, the monomial basis allows us to generate endomorphisms or "multiplication maps" for any given polynomial and thus to be able to compute in the coordinate ring. Furthermore, this leads to an algorithm for determining the common roots of a system of nonlinear polynomial equations.

Previous approaches to computing such bases and multiplication maps have relied on the use of Gröbner bases [19]. The algorithms obtained from these approaches have complexity exponential in the number of variables n. The main drawback of Gröbner bases algorithms is that usually they cannot take advantage of the inherent *sparseness* of the system of polynomial equations. In contrast, *sparse elimination* methods are designed with exactly this goal in mind.

In this paper, we reconsider the above questions for regular sequences of polynomials, in light of recent work on sparse resultant theory [9, 24]. The notion of sparseness is captured very effectively in terms of the concepts of *Newton polytopes* and *Mixed Volumes* and these methods provide us with algorithms whose practical complexity depends on the inherent sparseness of the system as measured by these concepts. Using such techniques, we provide constructive methods for computing monomial bases, linear multiplication maps and the solutions to a system of polynomials; the latter is achieved by reduction to an eigenvalue problem. Our approach is based on the matrix construction for the sparse resultant in [9]. This gives us an algorithm that is particularly simple and efficient in practice although its asymptotic worst case complexity is singly exponential in n, the dimension of the problem.

We have implemented our algorithm and applied it to the well-known problem of cyclic n-roots [5]. In the cases $n = 10$ and $n = 11$, our implementation provides the first known upper bounds on the number of isolated roots. For smaller n our bounds usually agree exactly with the known ones. The performance of our method for $n = 7, 8$ over zero characteristic is at least one order of magnitude superior to several Gröbner bases implementations over finite fields; the speed-up is even greater when compared to the Gröbner implementations over

*Address: Computer Science Division, University of California, Berkeley, CA 94720, U.S.A.
E-mail: {emiris,rege}@cs.Berkeley.edu. Fax: +1-510-642-5775.
Supported by a David and Lucile Packard Foundation Fellowship and by NSF P.Y.I. Grant IRI-8958577.

zero characteristic. We have also implemented the algorithm for solving systems of polynomial equations : some encouraging preliminary results are presented.

The paper is organized as follows : In section 2, we give a brief overview of related work. Section 3 summarizes the sparse resultant construction of [9] which we use in Section 4 to obtain a monomial basis. Section 5 specifies the algorithm for constructing a monomial K-basis for a given coordinate ring and assesses, on the one hand, its asymptotic complexity and on the other hand its practical performance based on our implementation. Section 6 deals with the construction of the multiplication map and root finding for polynomial systems. Section 7 shows how to obtain a Poisson formula for the sparse resultant.

2 Related Work

A method for constructing vector bases of coordinate rings as monomials indexed by the lattice points in the mixed cells of a mixed subdivision was first demonstrated by Pedersen and Sturmfels [22]. Our approach is based on a matrix formula for the sparse resultant [9] which leads to a simple proof and algorithm for the construction of monomial bases.

The classical resultant provides a means for root-finding by the use of U-resultants [26, 16, 23, 7]. Related to this approach, the reduction of this problem to an eigenvalue problem was formalized in [2] and, independently, in [18, 17]. The latter methods led to impressive results on certain problems from graphics and kinematics, thus motivating our work on resultants.

The recent interest in sparse methods is founded on the observation that the number of affine roots is typically significantly less than that predicted by the Bezout bound. Various problems from robot kinematics and computer vision illustrate this statement, such as the forward kinematics of the Stewart platform. This problem reduces to solving a system of polynomials that has, generically, 40 complex roots; see [20] and its references. Equivalent formulations of the problem give different Bezout and Bernstein bounds, on the number of projective and affine roots respectively; the lowest bounds we have obtained are, respectively, 256 and 54.

3 Matrix Formulae for the Resultant

This section summarizes the construction of matrix M in [9], whose determinant is a nontrivial multiple of the sparse resultant. A new variant of this algorithm yields a smaller matrix with the same properties.

Consider $n+1$ *Laurent* polynomials $f_0, \ldots, f_n \in K[x_1, x_1^{-1}, \ldots, x_n, x_n^{-1}] = K[x, x^{-1}]$, where the base field is $K = \mathbb{Q}(\{c_{ij}\})$ and $\{c_{ij}\}$ is the set of all coefficients in the polynomials. We use x^e to denote the monomial $x_1^{e_1} \ldots x_n^{e_n}$, where $e = (e_1, \ldots, e_n) \in \mathbb{Z}^n$ is an exponent vector. Let $\mathcal{A}_i = \{a_{i1}, \ldots, a_{im_i}\} \subset \mathbb{Z}^n$ denote the set of exponent vectors corresponding to the monomials of f_i with non-zero coefficients. $\mathcal{A}_i$ is called the *support* of f_i.

Let $Q_i \subset \mathbb{R}^n$ denote the convex hull of the points in $\mathcal{A}_i$. Q_i is called the *Newton polytope* of f_i. A technical assumption is that $\cup_{i=0}^n \mathcal{A}_i$ spans the affine lattice $\mathbb{Z}^n$; otherwise, a coordinate transformation by means of a Smith normal form can guarantee this [24]. The algorithm selects $n+1$ linear lifting functions $l_i : \mathbb{R}^n \to \mathbb{R}$ for $0 \le i \le n$. It constructs the lifted Newton polytopes

$$\hat{Q}_i \triangleq \{(p_i, l_i(p_i)) : p_i \in Q_i\} \subset \mathbb{R}^{n+1}, \qquad 0 \le i \le n$$

and takes their Minkowski sum $\hat{Q} = \hat{Q}_0 + \cdots + \hat{Q}_n \subset \mathbb{R}^{n+1}$. The projection of all facets on the lower envelope of $\hat{Q}$ onto

$$Q = Q_0 + \cdots + Q_n \subset \mathbb{R}^n$$

induces a *mixed subdivision* Δ of the latter. The linear lifting functions l_i are chosen to be *sufficiently generic*, such that every point in the mixed subdivision is uniquely expressed as the sum of $n+1$ points, one from each Q_i. Thus, each maximal cell σ in Δ is uniquely expressed as the Minkowski sum $F_0 + \cdots + F_n$, where each F_i is a face of Q_i, for $0 \le i \le n$. is the sum of the dimensions of the faces in its expression. Cells are either *mixed* or *unmixed*, mixed cells being Minkowski sums such that exactly one face is a vertex and all others are edges. This construction is essentially due to Sturmfels [24].

Definition 3.1 *A mixed maximal cell of the induced mixed subdivision of Q is i-*mixed *if, in its unique expression as a Minkowski Sum, the summand from Q_i is a vertex.*

It follows from the construction above that $MV(Q_0, \ldots, Q_{i-1}, Q_{i+1}, \ldots, Q_n)$ equals the sum of volumes of all i-mixed cells.

Canny and Emiris [9] construct a matrix M whose rows and columns are indexed by the integer lattice points

$$\mathcal{E} = (Q + \delta) \cap \mathbb{Z}^n,$$

where $Q + \delta$ is a polytope obtained by perturbing Q by some arbitrarily small $\delta \in \mathbb{Q}^n$, chosen to be sufficiently generic to ensure that every perturbed lattice point lies strictly inside a neighbouring maximal cell. The bijective correspondence between the integer lattice and the set of Laurent monomials allows us to consider $\mathcal{E}$ either as a point set or a monomial set. Each row of M contains the coefficients of $x^p f_i$, for some $0 \le i \le n$ and some $p \in Q_0 + \cdots + Q_{i-1} + Q_{i+1} + \cdots + Q_n$.

Theorem 3.1 [9] *The matrix M, described above, is generically nonsingular and its determinant is divisible* *by the sparse resultant $R(f_0,\ldots,f_n)$. Moreover, the degree of* $\det M$ *in the coefficients of* f_0 *equals* $MV(f_1, \ldots, f_n)$*, which also equals the degree of* $R(f_0,\ldots,f_n)$ *in these coefficients.*

Let matrix $\hat{M}$ be obtained from M by specializing all coefficients to powers of a new variable t and denote by $\hat{M}_{pq}$ the entry of $\hat{M}$ with row index p and column index q, for some $p, q \in \mathcal{E}$, then

Lemma 3.2 [9, Lemma 16] *For all non-zero elements* $\hat{M}_{pq}$ *with* $p \neq q$, $\deg_t(\hat{M}_{pq}) > \deg_t(\hat{M}_{qq})$.

Lemma 3.3 *Every principal minor of* M *is generically nonzero.*

Proof By the previous lemma. □

An improved version of this algorithm, proposed by J. Canny [8], is presented below; it constructs a matrix N which is at most as large as M and possesses the same properties as M.

Definition 3.2 *A* closed *submatrix* N *of* M *is any square proper submatrix of* M *such that the sets of monomials indexing the rows and columns of* N *are identical and, for every row included in* N*, the entries of this row that are not in* N *are all zero.*

Theorem 3.4 *For any closed submatrix* N *of* M *that includes at least* $MV(f_0,\ldots,f_{i-1},f_{i+1},\ldots,f_n)$ *rows containing multiples of* f_i*, the determinant* $\det N$ *is a nontrivial multiple of the resultant. Moreover, the degree of* $\det N$ *in the coefficients of* f_i *lies between the respective degrees of the resultant and of* $\det M$ *in these coefficients.*

Proof By Lemma 3.3, N is generically nonsingular. The proof of Theorem in [9] also works for N to show $R \,|\, \det N$. The relation on the degrees follows from the divisibility of $\det N$ by R and the fact that the rows of N containing f_i form a subset of those in M. □

It is straightforward to construct matrix N by a *greedy* variant of the previous algorithm, once we have the subdivision Δ of $Q + \delta$. The only difference from the previous algorithm is the way the row and column monomials, denoted respectively by $\mathcal{R}$ and $\mathcal{C}$, are specified. Let B_i contain all monomials x^p for which matrix N has a row filled by the coefficients of $x^p f_i$, for $0 \leq i \leq n$. Initially, $\mathcal{R}$ contains one monomial from some 0-mixed cell of Δ. At any step,

$$\mathcal{C} = \cup_{i=0}^{n} (B_i + \mathcal{A}_i)$$

contains all column monomials required for closedness. In subsequent steps, $\mathcal{R}$ is incremented to include $\mathcal{C}$ and the algorithm iterates as long as $\mathcal{R}$ and $\mathcal{C}$ are different. In the rest of this paper we use a matrix obtained by either algorithm and denote it by M.

4 Monomial Bases for Coordinate Rings

In this section, we use the matrix construction of [9], outlined in Section 3, to obtain a monomial basis for the coordinate ring generated by the given polynomials. The fact that the coordinate ring has a vector space basis consisting of the monomials indexed by the lattice points in the mixed cells of a mixed subdivision was first demonstrated by Pedersen and Sturmfels [22]. Their proof relies on reducing the general problem to binomial systems via Puiseux series. Here, we obtain the result in a more straightforward fashion by using the above construction and certain well-known matrix techniques [2].

Consider a set of n *generic* Laurent polynomials $f_1, \ldots, f_n$ in n variables; let $\mathcal{I} = \mathcal{I}(f_1,\ldots,f_n)$ be the ideal that they generate and $V = V(f_1,\ldots,f_n) \in (\overline{K}^*)^n$ their variety, where $\overline{K}$ is the algebraic closure of K. Assume that V has dimension zero. Then its coordinate ring $K[x, x^{-1}]/\mathcal{I}$ is an m-dimensional vector space over K, where $m = MV(f_1,\ldots,f_n)$ equals the Mixed Volume of the respective Newton polytopes [4]. In addition, the ideal $\mathcal{I} = \mathcal{I}(f_1,\ldots,f_n)$ is assumed to be radical which is equivalent to saying that all roots in V are distinct. This section proves that a specific subset of the monomials indexing the sparse resultant matrix constitutes a K-basis of $K[x,x^{-1}]/\mathcal{I}$.

We add an appropriate f_0 to the set $f_1,\ldots,f_n$ and construct the Minkowski sum $Q + \delta$ and its mixed subdivision as in the previous section. Without loss of generality we can choose f_0 such that it has the constant monomial 1 as one of its monomials. This follows easily from the fact that given an arbitrary f_0 in $K[x,x^{-1}]$, we can divide it by one of its monomials without changing its roots in $(\overline{K}^*)^n$.

Let $\mathcal{B} \subset \mathcal{E} \subset \mathbb{Z}^n$ be the set of all integer lattice points that lie in 0-mixed cells, in the subdivision of $Q+\delta$. Equivalently, $\mathcal{B}$ is the set of all Laurent monomials with exponent vectors in the 0-mixed cells of $Q+\delta$. By Theorem 3.1, we can write $\mathcal{B} = \{b_1,\ldots,b_m\}$. We will show that $\mathcal{B}$ is in fact a monomial K-basis for $K[x,x^{-1}]/\mathcal{I}$.

An important property of the matrix construction of the previous section is that postmultiplication with a column vector expresses evaluation of the polynomials whose coefficients have filled in the rows of the matrix. More precisely, for an arbitrary $\alpha \in K^n$,

$$M \begin{bmatrix} \vdots \\ \alpha^q \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ \alpha^p f_{i_p}(\alpha) \\ \vdots \end{bmatrix}, \tag{1}$$

where $p \in \mathcal{E}$ indexes the row of M that contains the coefficients of $x^p f_{i_p}(x)$ and $q \in \mathcal{E}$ indexes the column corresponding to monomial x^q. Since $\mathcal{A}_0$ contains $0^n \in$

$\mathbb{Z}^n$ we can always pick, without loss of generality, lifting function l_0 such that Q_0 contributes only its zero vertex 0^n as a summand to the 0-mixed cells. By definition ([9]), every row indexed by a monomial in $\mathcal{B}$ contains the coefficients of $x^{b-0^n} f_0 = x^b f_0$, for some $b \in \mathcal{B}$.

The partition of $\mathcal{E}$ into $\mathcal{B}$ and $\mathcal{E} \backslash \mathcal{B}$ defines four blocks in M shown below, where the rightmost set of columns and bottom set of rows are indexed by $\mathcal{B}$. Relation (1) becomes

$$M \begin{bmatrix} \vdots \\ \alpha^{q_c} \\ \vdots \\ \alpha^{b_i} \\ \vdots \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \vdots \\ \alpha^{q_c} \\ \vdots \\ \alpha^{b_i} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ 0 \\ \vdots \\ \alpha^{b_i} f_0(\alpha) \\ \vdots \end{bmatrix} \tag{2}$$

where q_c ranges over $\mathcal{E} \setminus \mathcal{B}$, b_i ranges over $\mathcal{B}$ and $\alpha \in V$ is a fixed common root.

Theorem 4.1 *Assume that variety V has dimension zero and ideal $\mathcal{I}$ is radical. Then, the set of monomials $\mathcal{B}$ form a vector-space basis for the coordinate ring $K[x, x^{-1}]/\mathcal{I}$ over K.*

Proof By Lemma 3.3 every principal minor of M is generically non-zero. Then, we can define the $m \times m$ matrix

$$M' = M_{22} - M_{21} M_{11}^{-1} M_{12}. \tag{3}$$

We premultiply both sides of (2) with the non-singular matrix

$$\begin{bmatrix} I & 0 \\ -M_{21} M_{11}^{-1} & I \end{bmatrix}, \tag{4}$$

where I is the identity matrix, and obtain

$$\begin{bmatrix} M_{11} & M_{12} \\ 0 & M' \end{bmatrix} \begin{bmatrix} \vdots \\ \alpha^{q_c} \\ \vdots \\ \alpha^{b_i} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ 0 \\ \vdots \\ \alpha^{b_i} f_0(\alpha) \\ \vdots \end{bmatrix}. \tag{5}$$

Let $f_0(x) = c_{00} + \sum_{j=1}^{n} c_{0j} x_j \in K[x, x^{-1}]$ with c_{0j} being generic indeterminates. Let v_α be the column subvector $[\alpha^{b_i}]$, with b_i ranging over all elements of $\mathcal{B}$. Then, (5) gives an eigenvector equation

$$M' v_\alpha = f_0(\alpha) v_\alpha \Rightarrow (M' - f_0(\alpha) I)\, v_\alpha = 0.$$

Since $\alpha \in (\overline{K}^*)^n$, every v_α is nonzero. Furthermore, the roots α are distinct and, by the genericity of c_{0j}, all eigenvalues $f_0(\alpha)$ are distinct. This implies that all eigenvectors v_α are linearly independent.

If the monomials in $\mathcal{B}$ are not a basis, then a non-trivial linear combination of them over K must belong to $\mathcal{I}$. Hence, there are elements $k_1, \ldots, k_m \in K$ not all zero such that, for every $\alpha \in V$, $\sum_{i=1}^{m} k_i \alpha^{b_i} = 0$. Therefore, the square matrix with columns v_α has dependent rows, which contradicts the independence of vectors v_α. $\square$

In other words, we have defined a canonical surjective homomorphism

$$K[x, x^{-1}] \rightarrow K[x, x^{-1}]/\mathcal{I} : \quad g \mapsto g \bmod \mathcal{I} = \sum_{b_i \in \mathcal{B}} c_{b_i} x^{b_i},$$

$$\text{such that } g \in \mathcal{I} \Leftrightarrow c_{b_i} = 0, \forall b_i \in \mathcal{B}.$$

5 An Algorithm for Constructing Monomial Bases

We have shown that the set of monomials $\mathcal{B}$ corresponding to the 0-mixed cells constitutes a basis for $K[x, x^{-1}]/\mathcal{I}$. It turns out that we can actually compute the basis in a simpler fashion, without going through the resultant matrix construction because the set $\mathcal{B}$ is defined independently of f_0. In that sense our results verify those of [22].

Consider a mixed subdivision of the perturbed Minkowski sum

$$Q^0 + \delta = Q_1 + \cdots + Q_n + \delta$$

where δ is the same as in the previous section. The maximal cells in the subdivision are again either mixed, when they are the Minkowski sum of n edges, or unmixed. The sum of all mixed cell volumes is $m = MV(f_1, \ldots, f_n)$.

Lemma 5.1 *Consider the mixed subdivision Δ^0 of $Q^0 + \delta$ induced by lifting functions $l_1, \ldots, l_n$. Then $\mathcal{B}$ equals the set of all integer lattice points in the mixed cells of this subdivision.*

Proof Recall that Q is the Minkowski sum of $n+1$ Newton polytopes, $\mathcal{A}_0$ contains the zero exponent 0^n and Δ is the mixed decomposition of $Q + \delta$ induced by $l_0, \ldots, l_n$. Let $\hat{Q}^0$ and $\hat{Q}$ be the Minkowski sums of the respective lifted Newton polytopes.

Consider a lower envelope facet $\hat{\sigma}$ of $\hat{Q}^0$, where its perturbed projection $\sigma + \delta$ is a mixed cell in Δ^0. We can pick l_0 so that its value is so much smaller at 0^n than at other elements of $\mathcal{A}_0$ that, for every facet $\hat{\sigma}$, the sum $(0^n, l_0(0^n)) + \hat{\sigma}$ is a lower envelope facet on $\hat{Q}$. Then the total volume of all cells in Δ of the form $0^n + \sigma + \delta$, where $\sigma + \delta$ is a mixed cell of Δ^0, is m. All of these cells are 0-mixed by construction, hence there are no more 0-mixed cells in Δ.

An appropriate choice of l_0, therefore, establishes a bijective correspondence between mixed cells of Δ^0 and

0-mixed cells of Δ. The proof is completed by noting that the integer points in the latter cells are of the form $0^n + p$, where $p \in Q^0$ and, actually, p belongs to a mixed cell of Δ^0. □

This gives rise to the following improved algorithm for computing monomial bases:

Input : n polynomials in n variables.
Output : a monomial basis for the coordinate ring corresponding to these polynomials.

1. Compute the respective Newton polytopes $Q_1, \ldots, Q_n$.
2. Pick sufficiently generic lifting functions $l_1, \ldots, l_n$ and compute the induced mixed subdivision Δ^0 of $Q^0 + \delta$.
3. Identify all mixed maximal cells σ of Δ^0.
4. For each σ, enumerate all lattice points $\sigma \cap \mathbb{Z}^n$. Each of these lattice points is the exponent of a unique monomial in the basis.

5.1 Complexity and Implementation

We examine the asymptotic worst-case complexity of the algorithm, as well as its empirical complexity as demonstrated in a series of experiments using our implementation.

The asymptotic complexity is clearly dominated by that of computing a mixed decomposition of Q^0. Let μ bound the number of vertices in every Newton polytope and

$$\mathcal{E}^0 = (Q^0 + \delta) \cap \mathbb{Z}^n.$$

Then the lifting method in [9] has total worst-case bit complexity, if we ignore the polylogarithmic factor, of $\mathcal{O}^*((n\mu)^{5.5}|\mathcal{E}^0|)$. Computing the mixed decomposition reduces to linear programming tests, for which any polynomial-time algorithm may be used; the above bound was based on Karmarkar's algorithm [15].

For the important class of *unmixed* polynomial systems, i.e. systems of polynomials with identical Newton polytopes, [9] proves $|\mathcal{E}^0| = \mathcal{O}(2^n m)$, where m is the Mixed Volume of the system. The same bound obviously holds for *mixed* systems where the Newton polytopes, though arbitrary, do not differ significantly. In the general case, though,

Lemma 5.2 *For arbitrary systems of n polynomials in n variables, $|\mathcal{E}^0| = \mathcal{O}(m^n)$.*

Proof By definition, m depends linearly on the scalar factor s of any Newton polytope, while the volume of the Minkowski sum, and hence $|\mathcal{E}^0|$, grows with s^n. Imagine that each Newton polytope Q_i starts at fixed sizes and is scaled by s_i to attain its actual size, then $m = \Omega(\prod_{i=1}^n s_i)$, while $|\mathcal{E}^0| = \mathcal{O}((\prod_{i=1}^n s_i)^n)$. □

Table 1: **Algorithm performance for the cyclic n-roots problem on an ALPHA DECSTATION.**

n	bounds: known	bounds: Bernstein	computing time
5	70	70	0s
6	156	156	2s
7	924	924	27s
8	1152	2560	4m 19s
9	-	11016	40m 59s
10	-	35940	4h 50m 14s
11	-	184756	38h 26m 44s

This bound is tight as shown by the example of $n-1$ hypercubes of fixed side length and a single hypercube of length side proportional to n, where $m = \mathcal{O}(n)$ and $|\mathcal{E}^0| = \Theta(n^n)$.

Theorem 5.3 *The worst-case bit complexity of our algorithm for computing monomial bases is exponential in n and polynomial in μ and m.*

Gröbner bases methods exhibit the same asymptotic complexity, namely single exponential in n and polynomial in m. The merit of all sparse elimination methods, though, including our monomial bases algorithm, lies in the fact that their complexity is directly related to the sparseness of the given system and, hence, they are expected to perform better for several problems in practice.

We present an implementation for computing the Mixed Volume and a monomial basis by J. Canny and the first author, which is the fastest to the best of our knowledge. The main idea is to minimize the number of large edge tuples that must be checked, by performing several tests with small tuples. The basic fact behind this idea of *pruning* [8] is that an edge tuple $(e_{i_1}, \ldots, e_{i_k})$ which does not give rise to any mixed cell in the decomposition of $Q_{i_1} + \cdots + Q_{i_k}$, cannot contribute to any mixed cell of Δ^0. The pruning algorithm identifies the mixed cells of partial Minkowski sums $Q_{i_1} + \cdots + Q_{i_k}$ where k ranges from 2 to n. At the last step we add the volumes of all mixed cells to find the Mixed Volume or enumerate the lattice points in the 0-mixed cells to obtain a monomial basis.

Table 5.1 reports the running times of our program applied to the benchmark problem of cyclic n-roots [5, 6] on an ALPHA DECSTATION. The Bernstein bound is the Mixed Volume of the system, which provides an upper bound on the number of isolated roots as well as the cardinality of the monomial basis. In certain cases, e.g. for $n = 8, 9$, the variety has positive dimension.

For $n \geq 9$ our program has produced the first bounds on the cardinality of isolated roots. Since it ignores the polynomial coefficients, its results are tight for those values of n for which the system is sufficiently generic. Our implementation improves tremendously upon the performance of existing Gröbner bases programs, while, of course, it provides less information than a Gröbner bases algorithm: For $n = 7$, MACAULAY requires 30 minutes on a SUN 4 [5]. For $n = 8$, Faugère's GB over a finite characteristic takes more than 3 hours on a SUN SPARC 10 [12], while Backelin's BERGMAN [3] consumes more than 15 hours of a SUN 490 [6], running over zero characteristic.

6 Multiplication Maps

This section shows how matrix M', defined in (3), is the matrix of the endomorphism in $K[x, x^{-1}]/\mathcal{I}$ which expresses multiplication by polynomial f_0, hence it provides a *multiplication map* in $K[x, x^{-1}]/\mathcal{I}$. Again, we are assuming that $\mathcal{I}$ is radical, the corresponding variety V zero-dimensional, m denotes the cardinality of V and $K[x, x^{-1}]/\mathcal{I}$ is an m-dimensional vector space over K.

Lemma 6.1 *The rows of M' contain the coefficients of polynomials $x^{b_i} f_0 \bmod \mathcal{I}$, for some $b_i \in \mathcal{B}$.*

Proof Premultiplication of M by matrix (4) has the effect of adding scalar multiples of the rows indexed by $\mathcal{E} \setminus \mathcal{B}$ to those indexed by $\mathcal{B}$. Hence, the row of M indexed by $b_i \in \mathcal{B}$ now contains the coefficients of

$$x^{b_i} f_0 + \sum_{p \in \mathcal{E} \setminus \mathcal{B}} k_p x^p f_{j_p} \qquad \text{for some } k_p \in K.$$

On the other hand, (5) shows that each such polynomial is a linear combination over K of the monomials in $\mathcal{B}$. Thus the lemma is proven. □

Since $\mathcal{B}$ provides a vector space basis for $K[x, x^{-1}]/\mathcal{I}$ over K, every polynomial $g \in K[x, x^{-1}]/\mathcal{I}$ can be expressed as a vector $v_g \in K^m$, whose entries are indexed by $\mathcal{B}$ and contain the respective coefficients.

Theorem 6.2 *Let M' denote both the matrix and the associated endomorphism in $K[x, x^{-1}]/\mathcal{I}$ with respect to basis $\mathcal{B}$. Then this endomorphism expresses multiplication by polynomial $f_0 \in K[x, x^{-1}]/\mathcal{I}$,*

$$M' : K[x, x^{-1}]/\mathcal{I} \to K[x, x^{-1}]/\mathcal{I} : g \mapsto g f_0 \bmod \mathcal{I}.$$

In other words, if vector v_g expresses polynomial $g \in K[x, x^{-1}]/\mathcal{I}$, with respect to basis $\mathcal{B}$, then vector $v_g M'$ expresses polynomial $g f_0 \in K[x, x^{-1}]/\mathcal{I}$ with respect to the same basis.

Proof From the previous lemma row b_i of M' contains the coefficients of polynomial $x^{b_i} f_0 \bmod \mathcal{I}$. Let $g = \sum_{i=1}^m c_i x^{b_i}$, then

$$\begin{aligned} g f_0 \bmod \mathcal{I} &= \sum_{i=1}^m c_i (x^{b_i} f_0 \bmod \mathcal{I}) \\ &= \sum_{i=1}^m c_i (\sum_{j=1}^m M'_{ij} x^{b_j}) = \sum_{j=1}^m x^{b_j} (\sum_{i=1}^m c_i M'_{ij}). \end{aligned}$$

If $b_j \in \mathcal{B}$ indexes the j-th column of M', then the last polynomial is written as a row vector indexed by $\mathcal{B}$. Clearly, this vector is the product $v_g M'$. □

Matrix M' essentially allows computation within the coordinate ring. This is used to outline an algorithm for finding all roots of the given system of polynomials.

6.1 Polynomial System Solving

Root finding reduces to computing eigenvectors by an approach introduced in [2] and further discussed, in the context of Gröbner bases, in [19]. This section proves that the same approach is possible in the context of sparse elimination. Note that the two additional methods surveyed in [19, Sect. 5, 6] can be combined with our construction, first for finding the roots by means of the minimal polynomial and, second, for counting the number of real roots.

In computing matrix M by the algorithm in [9], f_0 is linear with generic coefficients, as in the proof of Theorem 4.1. In practice, we pick random coefficients c_{0j}, for $0 \leq j \leq n$, from some range of possible integer values of size r. A bad choice is one that will result in the same value of f_0 at two distinct roots α_1 and α_2. Assume that α_1 and α_2 differ in their i-th coordinate for some $i > 0$, then fix all choices of c_{0j} for $j \neq i$; the probability of a bad choice for c_{0i} is $1/r$, and since there are $\binom{m}{2}$ pairs of roots, the total probability of failure for this scheme is

$$\binom{m}{2} / r.$$

It suffices, therefore, to pick c_{0j} from a sufficiently large range in order to make the probability of success arbitrarily high. Moreover, it is clear that any choice of f_0 coefficients can be tested deterministically at the end of the algorithm.

We have seen that each eigenvector v' of M' contains the values of monomials $\mathcal{B}$ at some common root $\alpha \in (\overline{K}^*)^n$. Since there are exactly m eigenvalues as well as roots, by the Pigeonhole Principle every eigenvector will correspond to the values of the basis monomials at a distinct root. Define vector

$$v = -M_{11}^{-1} M_{12} v'$$

of size $|\mathcal{E}| - m$, indexed by $\mathcal{E} \setminus \mathcal{B}$. Vector v lies in the kernel of the homomorphism defined by the top $|\mathcal{E}| - m$ rows of M in (5), i.e. $[M_{11} M_{12}]v = 0$, where 0 here is the corresponding zero vector. Therefore the element of v indexed by $p \in \mathcal{E} \setminus \mathcal{B}$ is the value of monomial x^p at root α. Vectors v and v' together contain the values of every monomial in $\mathcal{E}$ at α.

Now, a set of monomials has to be found, such that their values at a root determine the root coordinates. Using v and v', the problem reduces to finding a subset of $\mathcal{E}$ that defines a simplex in n-space. The lattice spanned by $\mathcal{E}$ has dimension n, otherwise every Newton polytope would have positive codimension which implies that Q has positive codimension. Therefore there exist $n+1$ points in $\mathcal{E}$ forming a simplex.

A simple procedure to find such a set of points is the following: Select any set of n points from $\mathcal{E}$ and consider them as column vectors of a matrix. While this matrix does not have full rank, add the minimum number of points from $\mathcal{E}$ so that the matrix may achieve full rank. Continue until a full-rank matrix is obtained, which is guaranteed to happen after selecting at most $|\mathcal{E}|$ lattice points. This gives a set of n independent vectors; picking an additional distinct point produces a simplex.

Theorem 6.3 *Given matrix M, all common zeros of polynomials $f_1, \ldots, f_n$ are found in time asymptotically bounded by $|\mathcal{E}|^3$ and a polynomial in n and m. Hence the overall bit complexity is exponential in n and polynomial in m.*

Proof The main steps of this algorithm are, given matrix M, to compute matrix M', find its eigenvectors, compute the respective vectors v, find a subset of points in $\mathcal{E}$ forming a simplex and, finally, recover the root coordinates. The first three steps involve linear algebra operations on matrices of size at most $|\mathcal{E}|$, hence the rough upper bound $\mathcal{O}(|\mathcal{E}|^3)$. Finding the simplex involves a series of rank tests on an $n \times k$ matrix, where $n \leq k \leq |\mathcal{E}|$. If this test is implemented incrementally, the overall complexity to find the simplex is $\mathcal{O}(n|\mathcal{E}|^2)$.

The overall complexity bounds are obtained by applying Lemma 5.2 and Theorem 5.3. Dealing with asymptotic complexities, we can ignore the difference in cardinality between $\mathcal{E}$ and $\mathcal{E}^0$. □

It is important to note that in using resultants, much of the computation needs to be conducted only once for a system with given supports. In particular, constructing the Newton polytopes, finding the appropriate lattice points and computing M all belong to a preprocessing step, while computing M' and finding eigenvectors v' and kernel vectors v are done on-line for the specified coefficients.

In practice, the algorithm of [10] is preferred, which is expected to construct smaller resultant matrices. For certain classes of systems, including multigraded ones, it produces optimal resultant matrices, called Sylvester-type formulae, by using the results of [25]. In the rest of this section we describe the application of our C implementation of this method for root finding to the motion from point-matches problem [11] and the forward kinematics problem of the Stewart platform [20] from vision and robotics respectively. The results presented are preliminary and we expect to improve upon them in the near future.

Given a square polynomial system of n equations in n unknowns, instead of adding an extra linear polynomial, we hide one of the variables in the coefficient field, thus obtaining an over-constrained system. By arguments similar to those above, we reduce root-finding to an eigenvector problem, where the eigenvalues correspond to the values of the hidden variable at the different solutions and the eigenvectors give the values of monomials at the roots. Typically, the lattice points in $\mathcal{E}$ required to recover the roots are found by walking in the lattice, thus avoiding any rank tests. For both problems under examination this is possible and, further, most of the lattice points needed lie in $\mathcal{B}$; hence only a few entries of v must be computed.

For the motion from point-matches problem, also referred to as relative orientation, we use the quaternion formulation of [13]. After hiding one variable in the coefficient field we obtain a bilinear system of 6 equations in 5 unknowns, which has a Sylvester-type formula for its sparse resultant. All preprocessing, including construction of this formula, consumes 23 seconds on a Sun Sparc 10. After specializing the coefficients to their given values, Gaussian elimination on M produces a 20×20 matrix M', which is of optimal size since the Mixed Volume of the original 6×6 system is 20; this Mixed Volume is known to be exact [11]. It takes currently 1 second to produce M' from M, reduce to an eigenvector problem, solve the latter numerically by the appropriate LAPACK library routines [1] and recover the actual solution vectors.

The general Stewart platform problem is expressed as an 8×8 polynomial system, for which we obtain matrix M' of less than twice the optimal size and expect that the last phase of the program will run in real time.

7 A Poisson Formula

A Poisson formula for the sparse resultant was given in [21], where the extraneous factor was described. In this section we show how the matrix M' can be used to obtain a Poisson formula for the sparse resultant. Again, we shall focus on the evaluations of the $\mathcal{B}$ monomials at the roots; let $v_i = [\alpha_i^{b_1}, \ldots, \alpha_i^{b_m}]^T$.

Lemma 7.1 *If the monomials in $\mathcal{B} = \{x^{b_1}, \ldots, x^{b_m}\}$ are linearly independent in $K[x, x^{-1}]/\mathcal{I}$ over K, then the vectors v_i are linearly independent over K.*

Proof If the vectors v_i are not independent, then the $m \times m$ matrix that has them as columns is singular. Therefore there exist $k_1, \ldots, k_m$ in K which are not all zero, such that

$$\sum_{i=1}^{m} k_i [\alpha_1^{b_i}, \ldots, \alpha_m^{b_i}] = 0 \Rightarrow \sum_{i=1}^{m} k_i \alpha_j^{b_i} = 0, \forall \alpha_j \in V. \quad (6)$$

Let $g(x) = \sum_{i=1}^{m} k_i x^{b_i}$ be a polynomial in $K[x, x^{-1}]/\mathcal{I}$ which is not identically zero because the k_i cannot be all zero. On the other hand, (6) implies that g vanishes on V, thus $g \in \mathcal{I}$ and hence g should be identically zero in $K[x, x^{-1}]/\mathcal{I}$, under the canonical basis $\mathcal{B}$. Thus we arrive at a contradiction. $\square$

Theorem 7.2 *Letting $V = \{\alpha_1, \ldots, \alpha_m\}$, we have*

$$\det M' = \prod_{i=1}^{m} f_0(\alpha_i).$$

Proof Recall that $\{x^{b_1}, \ldots, x^{b_m}\}$ is a vector space basis of $K[x, x^{-1}]/\mathcal{I}$ over K. We have seen that each root α_i corresponds to an eigenvector $v_i = [\alpha_i^{b_1}, \ldots, \alpha_i^{b_m}]^T$ of M', with associated eigenvalue $f_0(\alpha_i)$. By the previous lemma, all m vectors v_i are independent over K.

From linear algebra we know that if the eigenvectors span the domain and range of square matrix M', then M' is similar to a diagonal matrix D whose diagonal entries are the eigenvalues of M'; see e.g. [14, Thm. VII.5.5]. The determinant of M' equals that of D which is equal to the product $f_0(\alpha_1) \cdots f_0(\alpha_m)$. $\square$

A direct corollary is that the determinant of the matrix M equals the Poisson expression of the sparse resultant multiplied by $\det M_{11}$, which is thus shown to be the extraneous factor in $\det M$.

8 Conclusions and Open Questions

In this paper, we demonstrated a practical method for computing the monomial basis for the coordinate ring of a 0-dimensional variety, whose ideal is radical. The key idea was to make use of the matrix formula for the sparse resultant of the system of polynomial equations defining the variety. The use of sparse elimination techniques enabled us to provide an algorithm which exploits the sparseness of the system under consideration and which runs efficiently under empirical tests. An open question that arises immediately is whether the algorithm above can be extended to the case where the common roots of the given system of polynomial equations are not all distinct.

Based on the monomial basis construction, we can obtain multiplication maps for polynomial equivalence classes in the coordinate ring. Further, these ideas could be extended to actually determine the common roots of the given polynomial system. Another important open question that comes up is extending the above analysis to the case where the number of polynomials is larger than the dimension of the problem.

We applied the above algorithms to compute the monomial bases for the cyclic n-roots problem. In the case of $n = 10$ and $n = 11$ we obtain the first known upper bounds on the number of isolated roots. Further, our implementation shows a speed-up of at least one order of magnitude over several Gröbner bases packages.

Acknowledgments

We acknowledge lengthy and constructive discussions with John Canny and thank Carlo Traverso for pointing out some related work. We also thank the referees for their detailed comments.

References

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen. *LAPCK Users' Guide.* SIAM, Philadelphia, 1992.

[2] W. Auzinger and H.J. Stetter. An Elimination Algorithm for the Computation of all Zeros of a System of Multivariate Polynomial Equations. In *Proc. Intern. Conf. on Numerical Math., Intern. Series of Numerical Math., 86*, pages 12–30. Birkhäuser Verlag, Basel, 1988.

[3] J. Backelin and R. Fröberg. How we Proved that there are exactly 924 Cyclic 7-Roots. In *Proc. ACM Intern. Symp. on Symbolic and Algebr. Computation*, pages 103–111, Bonn, 1991.

[4] D.N. Bernstein. The Number of Roots of a System of Equations. *Funct. Anal. and Appl.*, 9(2):183–185, 1975.

[5] G. Björck and R. Fröberg. A Faster Way to Count the Solutions of Inhomogeneous Systems of Algebraic Equations, with Applications to Cyclic n-roots. *J. Symbolic Computation*, 12:329–336, 1991.

[6] G. Björck and R. Fröberg. Methods to "Divide out" certain Solutions from Systems of Algebraic Equations, Applied to Find all Cyclic 8-Roots.

Manuscript, Dept. of Math., Stockholm University, 1994.

[7] J. Canny. Generalised Characteristic Polynomials. *J. Symbolic Computation*, 9:241–250, 1990.

[8] J. Canny, 1993. Personal Communication.

[9] J. Canny and I. Emiris. An Efficient Algorithm for the Sparse Mixed Resultant. In G. Cohen, T. Mora, and O. Moreno, editors, *Proc. 10th Intern. Symp. on Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, Lect. Notes in Comp. Science 263*, pages 89–104, Puerto Rico, May 1993. Springer Verlag. Submitted to *SIAM J. Computing*.

[10] I. Emiris and J. Canny. A practical method for the sparse resultant. In *Proc. ACM Intern. Symp. on Symbolic and Algebr. Computation*, pages 183–192, Kiev, July 1993.

[11] O.D. Faugeras and S. Maybank. Motion from Point Matches: Multiplicity of Solutions. *Intern. J. Comp. Vision*, 4:225–246, 1990.

[12] J.-C. Faugère, 1994. Personal Communication.

[13] B.K.P. Horn. Relative Orientation Revisited. *J. Opt. Soc. Am.*, 8(10):1630–1638, October 1991.

[14] T.W. Hungerford. *Algebra.* Graduate Texts in Mathematics, 73. Springer-Verlag, New York, 1974.

[15] N. Karmarkar. A New Polynomial-Time Algorithm for Linear Programming. *Combinatorica*, 4:373–395, 1984.

[16] D. Lazard. Résolution des systèmes d'équations algébriques. *Theor. Comp. Science*, 15:77–110, 1981.

[17] D. Manocha and J. Canny. Multipolynomial Resultants and Linear Algebra. In *Proc. ACM Intern. Symp. on Symbolic and Algebr. Computation*, pages 96–102, Berkeley, July 1992.

[18] D. Manocha and J. Canny. Real Time Inverse Kinematics of General 6R Manipulators. In *Proc. IEEE Intern. Conf. Robotics and Automation*, pages 383–389, Nice, May 1992.

[19] H. M. Möller. Systems of Algebraic Equations Solved by Means of Endomorphisms. In G. Cohen, T. Mora, and O. Moreno, editors, *Proc. 10th Intern. Symp. on Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, Lect. Notes in Comp. Science, 263*, pages 43–56, Puerto Rico, May 1993. Springer Verlag.

[20] B. Mourrain. The 40 "Generic" Positions of a Parallel Robot. In *Proc. ACM Intern. Symp. on Symbolic and Algebr. Computation*, pages 173–182, Kiev, July 1993.

[21] P. Pedersen and B. Sturmfels. Product Formulas for Resultants and Chow Forms. *Math. Zeitschrift*, 214:377–396, 1993.

[22] P. Pedersen and B. Sturmfels. Mixed Monomial Bases. In *MEGA '94*, April 1994. To Appear.

[23] J. Renegar. On the Computational Complexity of the First-Order Theory of the Reals, parts I, II, III. *J. Symbolic Computation*, 13(3):255–352, 1992.

[24] B. Sturmfels. On the Newton Polytope of the Resultant. *J. of Algebr. Combinatorics*, 3:207–236, 1994.

[25] B. Sturmfels and A. Zelevinsky. Multigraded Resultants of Sylvester Type. *J. of Algebra*, 163(1):115–127, 1994.

[26] B.L. van der Waerden. *Modern Algebra.* Ungar Publishing Co., New York, 3rd edition, 1950.