Sparsity Considerations in Dixon Resultants*

Deepak Kapur and Tushar Saxena
Institute for Programming and Logics
Department of Computer Science
State University of New York at Albany
Albany, NY 12222

September 23, 1995

Abstract

New results relating the sparsity of non-homogeneous polynomial systems and computation of their projection operator (i.e., a non-trivial multiple of multivariate resultant) using Dixon's method are developed. It is proved that the Dixon formulation of resultants, despite being classical, implicitly exploits the structure of the Newton polytopes of input polynomials; the complexity of computing Dixon resultant is not determined by the total degree of the polynomial system. Bound on the size of the Dixon matrix of unmixed polynomial systems is derived in terms of their Newton polytopes. This bound is used to prove that for a multi-homogeneous system, the size of its Dixon matrix is of a smaller order than its n-fold mixed volume. Using dense multivariate polynomial interpolation techniques, it is shown that for a fixed number of variables, Dixon matrix of multi-homogeneous polynomial systems can be constructed using $\mathcal{O}(\mathcal{M}^3)$ arithmetic operations, where $\mathcal{M}$ is the n-fold mixed volume of the input system. The Dixon matrix is found to be smaller than the sparse and Macaulay resultant matrices by a factor exponential in the number of variables, and the Dixon formulation yields a faster algorithm to compute the resultant. This work links the classical Dixon formulation (developed in 1908) to the modern line of sparsity analysis based on Newton polytopes.

1 Introduction

Eliminating n variables from $n+1$ multivariate polynomials results in the *resultant* – a quantity which has proved useful in many applications [8]. Most efficient ways to compute the resultant express it as a determinant. However, a pure determinantal expression for the resultant is rare (see [13] for some classes of polynomials which allow a pure determinantal formulation), and in these cases, one can still express some nontrivial multiple of the resultant (called a *projection operator*) as a determinant.

There are three major methods to compute the resultant or a projection operator – the classical formulations by Dixon [4] and Macaulay [12] developed early this century, and the recently developed sparse resultant formulation [11, 6]. All three methods construct matrices whose determinant is either the resultant, or a projection operator.

The Macaulay formulation uses the traditional Bezout bound on the number of solutions of a polynomial system to construct the Macaulay matrix. The size of this matrix depends on the total degree of the input polynomials. However, recently, sharper bounds have been derived on the number

*Supported in part by a grant from United States Air Force Office of Scientific Research AFOSR-91-0361.

of solutions, known as BKK bounds [1], which exploit the structure and sparsity of the polynomials as determined by their Newton polytopes. The sparse resultant formulation is a modification of the Macaulay formulation which exploits the modern BKK bound and Newton polytopes of the input polynomials to construct smaller matrices [7]. The size of the sparse resultant matrix depends on the volume of the polytopes of input polynomials, and is not governed by their total degree. Size of sparse resultant matrix is small because it does not assume any monomials, besides the ones inside the Newton polytope, have non-zero coefficients – unlike the Macaulay matrix, which is much larger because it is constructed under the crude assumption that all monomials of total degree less than or equal to the input polynomial have non-zero coefficients, without regard to the more refined structure of the polynomial in terms of its Newton polytope.

Methods based on the Dixon formulation have been empirically found to be more efficient than the Macaulay and the sparse resultant formulations [9, 10]. However, being a classical approach (developed in 1908), the relationship between the Dixon formulation and the modern BKK bounds, or the structure of the input polytopes, is not well understood. In [11], Sturmfels gave the bracket representation of the Dixon resultant for bi-homogeneous systems; however the Dixon resultant was not analyzed in its full generality. Moreover, it has been unclear whether the Dixon formulation exploits the sparsity of input polynomials, and if so, to what extent (quantitatively). Little has been known about the size of the Dixon matrix and its relationship to the Newton polytopes of the input polynomials.

This paper makes three contributions. The first contribution is to show that the Dixon formulation of multivariate resultants implicitly exploits the structure of Newton polytopes of input polynomials and is, thus, not governed by their total degree (or the classical Bezout bound). Using this relationship, an explicit bound on the size of Dixon matrix is derived in terms of the volume of polytopes of input polynomials, and this is the second and main contribution of the paper. It is shown that the size of the Dixon matrices of unmixed systems is bounded by the number of integral points inside the Minkowski sum of *successive projections* of their Newton polytopes (Theorem 3.4 and Corollary 3.5). And, the third contribution of the paper is an upper bound on the complexity of constructing Dixon matrices. This is especially significant since entries in a Dixon matrix are more complex in contrast to entries in a Macaulay matrix or a sparse resultant matrix, which are simply the coefficients of terms in polynomials. This bound on the construction of Dixon matrix shows that despite more complicated entries, it can be constructed fast. These contributions are further elaborated below.

It is shown in this paper that, much like the sparse resultant formulation, the Dixon formulation assumes that all monomials outside the polytopes have zero coefficients. If the polytopes of two polynomial systems have the same volumes and projections, then no matter how different their total degrees, their Dixon matrices, like their sparse resultant matrices, would be of the same size. The size of the Dixon matrix is much smaller than the size of Macaulay matrix. It is also shown that the size of the Dixon matrix is smaller than the size of sparse resultant matrix, which is bounded by the number of integral points inside the Minkowski sum of the Newton polytopes (not their successive projections) of the input polynomials. This is analogous to the Dixon matrix whose size is bounded by the number of integral points inside the Minkowski sum of the *successive projections* of Newton polytopes of input polynomials. However, since the projections are of successively lower dimension than the polytopes themselves, the Dixon matrix is smaller. Specifically, the size of the Dixon matrix of multi-homogeneous polynomials is proved to be of smaller order than their n-fold mixed volume (Theorem 3.8), whereas the size of the sparse resultant matrix is larger than the n-fold mixed volume by an exponential multiplicative factor (Theorem 3.9). If the n-fold mixed volume of a multi-homogeneous polynomial system is $\mathcal{M}$ and the number of variables being eliminated is n

then for asymptotically large number of partitions of variables (approaching n),

$$\text{Size of its Dixon matrix} = \mathcal{O}\left(\frac{1}{n+1}\mathcal{M}\right), \tag{1}$$

$$\text{Size of its Sparse Resultant matrix} = \mathcal{O}\left(e^{n+1}\mathcal{M}\right). \tag{2}$$

This is actually congruent with the intuition Dixon discussed in his paper, because he mainly developed this formulation as an efficient method for n-degree polynomial systems[1], and for asymptotically large number of partitions, multi-homogeneous systems approach the n-degree case.

Regarding the construction of the Dixon matrix, it is shown that it can be constructed fast using a variant of the dense multivariate polynomial interpolation algorithm, which exploits the bounds on the size of the Dixon matrix. Specifically, for multi-homogeneous systems with a fixed number of variables, the Dixon matrix can be constructed using $O(\mathcal{M}^3)$ arithmetic operations (Theorem 4.1).

Putting these results together, the cost of computing a projection operator using Dixon formulation is the sum of (i) the cost of constructing the Dixon matrix and (ii) the cost of computing its determinant. The number of arithmetic operations required for (ii) is $\mathcal{O}(\mathcal{M}^3)$ (see Equation 1). Consequently, for a fixed number of variables, the computation of a projection operator using the Dixon formulation takes $\mathcal{O}(\mathcal{M}^3)$ arithmetic operations. This is in contrast to the sparse resultant formulation which, if the polynomials are *full*[2], then for a fixed number of variables, takes $\mathcal{O}(\mathcal{M}^{6.5})$ arithmetic operations [7]. If the polynomials are *hollow*[3], the construction time of the sparse resultant reduces considerably and the determinant computation dominates, but, it still takes longer than the Dixon formulation because the matrix size is larger (Equation 2).

To the best of our knowledge, this is the first time such a relationship has been shown between the classical Dixon formulation and the modern line of analysis based on Newton polytopes of polynomials. The analysis based on this relationship reveals that the Dixon formulation does take into account the sparsity of input polynomials, creates smaller resultant matrices, and has lower complexity than the sparse resultant formulation.

2 Dixon Method for Computing Projection Operators

Let $X = \{x_1, \ldots, x_n\}$ and $\bar{X} = \{\bar{x}_1, \ldots, \bar{x}_n\}$ be two sets of n variables each. Given a polynomial $p \in \mathbb{Q}[X]$, $p(\bar{x}_1, \ldots, \bar{x}_k,\ x_{k+1},\ \ldots, x_n)$ stands for uniformly replacing x_j by $\bar{x}_j$ for $1 \leq j \leq k$ in p. For a given polynomial $q \in \mathbb{Q}[X, \bar{X}]$, $V_X(q)$ and $V_{\bar{X}}(q)$, the **monomial vectors** of q wrt. X and $\bar{X}$ respectively, are defined to be the vectors of all monomials appearing in q in the variables X and $\bar{X}$ respectively. The **exponent vector** of the monomial $x_1^{e_1} \cdots x_n^{e_n} \bar{x}_1^{\bar{e}_1} \cdots \bar{x}_n^{\bar{e}_n}$ wrt. the set X of variables is defined to be the n-tuple $(e_1, \ldots, e_n) \in \mathbb{Z}^n$. The **support**, $supp(q) \subset \mathbb{Z}^n$, of a polynomial q is the set of exponent vectors of all the monomials, wrt. X, with non-zero coefficients in q. The determinant of a square matrix A is denoted by $|A|$.

Example: If $X = \{x_1, x_2\}$ and $p = 2x_1^3x_2\bar{x}_1^3 + x_2^4\bar{x}_1\bar{x}_2^2 + \bar{x}_1^2 + \bar{x}_1\bar{x}_2 + 1 \in \mathbb{Q}[X, \bar{X}]$, then $V_X(p) = [x_1^3x_2,\ x_2^4,\ 1]$ and $V_{\bar{X}}(p) = [\bar{x}_1^3,\ \bar{x}_1\bar{x}_2^2,\ \bar{x}_1^2,\ \bar{x}_1\bar{x}_2,\ 1]$. $supp(p) = \{(3,1), (0,4), (0,0)\}$. □

---

[1] $n+1$ nonhomogeneous polynomials $p_1, \ldots, p_{n+1}$ in $x_1, \ldots, x_n$ are called n-degree there exist nonnegative integers $m_1, \ldots, m_n$ such that each
$p_j = \sum_{i_1=0}^{m_1} \cdots \sum_{i_n=0}^{m_n} a_{j,i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$ *for* $1 \leq j \leq n+1$.

[2] In a full polynomial, all terms inside its Newton polytope have non-zero coefficients.

[3] In a hollow polynomial, only the terms on the boundary of its Newton polytope have non-zero coefficients.

Definition 2.1 *Let $P = \{p_1, \ldots, p_{n+1}\} \subset \mathbb{Q}[X]$ be a set of $n+1$ polynomials in n variables $x_1, \ldots, x_n$. The* **cancellation matrix** *of P, denoted C_P, is*

$$C_P = \begin{bmatrix} p_1(x_1, x_2, x_3, \ldots, x_n) & \cdots & p_{n+1}(x_1, x_2, x_3, \ldots, x_n) \\ p_1(\bar{x}_1, x_2, x_3, \ldots, x_n) & \cdots & p_{n+1}(\bar{x}_1, x_2, x_3, \ldots, x_n) \\ p_1(\bar{x}_1, \bar{x}_2, x_3, \ldots, x_n) & \cdots & p_{n+1}(\bar{x}_1, \bar{x}_2, x_3, \ldots, x_n) \\ \vdots & \vdots & \vdots \\ p_1(\bar{x}_1, \bar{x}_2, \bar{x}_3, \ldots, \bar{x}_n) & \cdots & p_{n+1}(\bar{x}_1, \bar{x}_2, \bar{x}_3, \ldots, \bar{x}_n) \end{bmatrix}.$$

The **Dixon polynomial** *of P, denoted $\delta_P \in \mathbb{Q}[X, \bar{X}]$, is*

$$\delta_P = \frac{|C_P|}{\prod_{i=1}^{n}(x_i - \bar{x}_i)},$$

and the **Dixon matrix** *of P, denoted D_P, is the matrix for which $\delta_P = V_{\bar{X}}(\delta_P)\, D_P\, V_X(\delta_P)$.*

If the Dixon matrix is square, its determinant is a projection operator whose vanishing is a necessary condition for the system P to have a common solution in the *algebraic torus* $(\mathbb{C} - \{0\})^n$. However, if the Dixon matrix is rectangular then the determinant (and the projection operator) cannot be directly computed. Even if it is square, it may be singular, resulting in a trivial projection operator, which is undesirable. The following theorem given in [9], provides a method to extract a non-trivial projection operator in such instances.

Theorem 2.2 *If D_P has a column which is linearly independent of all other columns, then the determinant of any non-singular rank submatrix of D_P is a non-trivial projection operator.*

This method of computing the projection operator is quite efficient in practice as demonstrated in [10]. As can be noticed from the theorem, this method may fail if there is no column in the Dixon matrix which is linearly independent of all other columns. However, such failure seems very rare; in fact, it has never occurred on the numerous problems that we have tried from different application domains. (For systems with two variables, it can be proved that this method always works [4].)

Below we establish a relationship between the Newton polytopes of the input polynomials and the size of the Dixon matrix, and compare it with resultant matrices in other formulations.

3 Newton Polytopes and the Dixon Matrix

Let the convex hull of a set $\mathcal{A} \subset \mathbb{Z}^n$ be denoted by $conv(\mathcal{A}) \subset \mathbb{R}^n$. The **Newton polytope** of a polynomial p is $\mathcal{N}(p) = conv(supp(p)) \subset \mathbb{R}^n$, where as defined above, $supp(p)$ is the set of exponent vectors of all the monomials wrt. X in p. For a polynomial p, $supp(p) \subseteq \mathcal{N}(p) \cap \mathbb{Z}^n$.

Given a set of points $\mathcal{A} \subset \mathbb{R}^n$ and $0 \leq i \leq n$, $\pi_i(\mathcal{A}) \subset \mathbb{R}^n$ stands for the set obtained by replacing the first i coordinates of all points in $\mathcal{A}$ by 0 – a projection of $\mathcal{A}$ to $n-i$ dimensions. We will abuse the terminology and call this projection as the $\mathbf{i^{th}}$ **projection** of $\mathcal{A}$. $\pi_0(\mathcal{A}) = \mathcal{A}$ and $\pi_n(\mathcal{A}) = \{0, \ldots, 0\}$.

The **Minkowski sum** of sets $\mathcal{A}, \mathcal{B} \in \mathbb{R}^n$ is $\mathcal{A} + \mathcal{B} = \{a + b \mid a \in \mathcal{A}, b \in \mathcal{B}\} \subset \mathbb{R}^n$, or equivalently,

$$\mathcal{A} + \mathcal{B} = \bigcup_{a \in \mathcal{A}} (a + \mathcal{B}). \tag{3}$$

Fact 3.1 *Given a set $\mathcal{A} \subset \mathbb{R}^n$ and two polynomials p and q, if $\mathcal{N}(p) \subseteq \mathcal{A}$ and $\mathcal{N}(q) \subseteq \mathcal{A}$, then $\mathcal{N}(p+q) \subseteq \mathcal{A}$.*

Fact 3.2 *Given two polynomials, p and q, $\mathcal{N}(p\,q) = \mathcal{N}(p) + \mathcal{N}(q)$.*

For simplicity, it is assumed from now onwards that the input system is **unmixed**, i.e., the Newton polytopes of all polynomials are equal. For such a set, P, let $\mathcal{N}(P)$ denote the Newton polytope of any polynomial in P. Following lemma relates the Newton polytope of the determinant of cancellation matrix to the polytopes of input polynomials.

Lemma 3.3 *For an unmixed set, $P \subset \mathbb{Q}[X]$, of $n+1$ polynomials, $\mathcal{N}(|C_P|) \subseteq \sum_{i=0}^{n} \pi_i(\mathcal{N}(P))$, where C_P is the cancellation matrix of P.*

Proof: The Newton polytope of all entries in row $i+1$ of the cancellation matrix is $\pi_i(\mathcal{N}(P))$. If $|C_P|$ is expanded as a sum of products, then each summand is a product of exactly $n+1$ polynomials, one from each row. Form Fact 3.2, the polytope of this summand is $\sum_{i=0}^{n} \pi_i(\mathcal{N}(P))$, so the result follows from Fact 3.1. □

We can relate the Dixon polynomial to the input polytopes as follows:

Theorem 3.4 *For an unmixed set, $P \subset \mathbb{Q}[X]$, of $n+1$ polynomials*[4]*,*

$$supp(\delta_P) \subseteq \sum_{i=0}^{n} \pi_i(\mathcal{N}(P)) \cap \mathbb{Z}^n.$$

Proof: Let $q = \prod_{i=1}^{n}(x_i - \bar{x}_i)$. Polynomial q contains the term $\bar{x}_1 \cdots \bar{x}_n$ which is a constant wrt. the set X of variables, so $(0,0,\ldots,0) \in \mathcal{N}(q)$. Since $|C_P| = q \times \delta_P$,

$$\begin{aligned}
\sum_{i=0}^{n} \pi_i(\mathcal{N}(P)) &\supseteq \mathcal{N}(|C_P|) = \mathcal{N}(q \times \delta_P) = \mathcal{N}(q) + \mathcal{N}(\delta_P) && \text{(by Lemma 3.3 and Fact 3.2)}\\
&= (\mathcal{N}(q) \cup \{(0,0,\ldots,0)\}) + \mathcal{N}(\delta_P) = (\mathcal{N}(q) + \mathcal{N}(\delta_P)) \cup \mathcal{N}(\delta_P) && \text{(by Equation 3)}\\
&\supseteq \mathcal{N}(\delta_P) \supseteq supp(\delta_P). && \square
\end{aligned}$$

The number of columns in the Dixon matrix equals the cardinality of support of the Dixon polynomial. The cardinality of an integer point set is asymptotically bounded by the volume of their convex hull [5]. Let $vol(\mathcal{A})$ denote the n-dimensional **volume** of a convex set $\mathcal{A} \subset \mathbb{R}^n$. We get the following bound on the size of Dixon matrix as a function of the Newton polytopes of input polynomials.

Corollary 3.5 *The number of columns in the Dixon matrix of an unmixed set, $P \subset \mathbb{Q}[X]$, of $n+1$ polynomials is*

$$\mathcal{O}\left(vol\left(\sum_{i=0}^{n} \pi_i(\mathcal{N}(P))\right)\right).$$

We believe this is the first time that such a bound on the size of the Dixon matrix has been derived in terms of the input polytopes. We now compare this to similar results on the size of the Macaulay and sparse resultant matrices.

[4] A slightly stronger version can be proved, viz., support of the Dixon polynomial is, in fact, the number of integer lattice points inside the polytope obtained *after perturbing* $\sum_{i=0}^{n} \pi_i(\mathcal{N}(P))$ just a little bit in the negative octant. Note that this number is strictly less than the number determined by Theorem 3.4 because it moves some of the boundary points out of the polytope. But this decrease does not change the analysis and bounds in the rest of the paper, so we adhere to the theorem above.

3.1 Comparison with Macaulay Matrix

If the total degree of each input polynomials is d, then the size of the Macaulay matrix is $\mathcal{O}\left(\frac{(n+1)^n d^n}{n!}\right)$. So the total degree completely determines the size of Macaulay matrix, no matter how many terms are missing from the polynomials. If a polynomial system has the same number of variables, but polynomials of higher total degree than another, the Macaulay matrix of the former is larger, even though their Newton polytopes may have the same shape and volume.

Corollary 3.5 shows that for the Dixon matrix, this is not the case at all. In fact, the size of the Dixon matrix has no dependence on the total degree of the polynomials, only on their Newton polytopes. Sizes of the Dixon matrices of two different polynomial systems are the same as long as their Newton polytopes have the same shape, no matter what the total degrees of the polynomials.

The Dixon formulation *does* consider the sparsity in the input polynomials and implicitly exploits the structure of the Newton polytopes of input polynomials.

3.2 Comparison with Sparse Resultant Matrix

A result analogous to Corollary 3.5 exists for the sparse resultant matrix [7]:

Fact 3.6 *Size of the sparse resultant matrix of an unmixed set, $P \subset \mathbb{Q}[X]$, of $n+1$ polynomials is*

$$\mathcal{O}\left(vol\left(\sum_{i=0}^{n} \mathcal{N}(P)\right)\right).$$

The reader will notice the surprising similarity between Fact 3.6 and Corollary 3.5. While the sparse resultant matrix considers integral points inside the Minkowski sum of the polytopes, the Dixon matrix considers integral points inside the Minkowski sum of *successive projections* of polytopes. Since the projections are of successively lower dimensions than the polytopes themselves, the Dixon matrix is smaller than the sparse resultant matrix.

As an example of how these various polytopes look, see Figure 1. P is the polytope of the unmixed system $p_i = a_i\, x^2 y + b_i\, x^3 y + c_i\, x^2 y^2 + d_i\, x^3 y^2 + e_i\, x^3 y^3$ for $1 \le i \le 3$. The number of integral points in D and S are the sizes of the Dixon and sparse matrices, respectively. The Macaulay matrix is so large that the region corresponding to terms in its columns cannot be shown in this figure – both, the sparse and Dixon polytopes, are subsets of this region. To see just how much the quantitative difference in sizes of the three resultant matrices is, we analyze the case of multi-homogeneous polynomial systems, which forms an important class with a large number of applications such as robotics, vision etc.

3.3 Multi-Homogeneous Systems

Definition 3.7 *A set of polynomials is called* **multi-homogeneous of type $(\mathbf{l_1}, \ldots, \mathbf{l_r}; \mathbf{d_1}, \ldots, \mathbf{d_r})$** *if it is unmixed and the set of variables can be partitioned into r subsets, $x_1, \ldots, x_r$ such that, for $1 \le i \le r$, the number of variables in the set x_i is l_i and the total degree of each polynomial in P in the variable set x_i is d_i.*

In the rest of the paper, a multi-homogeneous system should be understood to mean a multi-homogeneous polynomial set of type $(l_1, \ldots, l_r; d_1, \ldots, d_r)$ with $n+1$ polynomials, where $n = \sum_{i=1}^{r} l_i$ is the total number of variables. We also assume that, for $1 \le i \le r$, the number of partition variables $l_i = \mathcal{O}(l)$, and the partition degree $d_i = \mathcal{O}(d)$, and consequently, $n = \mathcal{O}(r\, l)$.

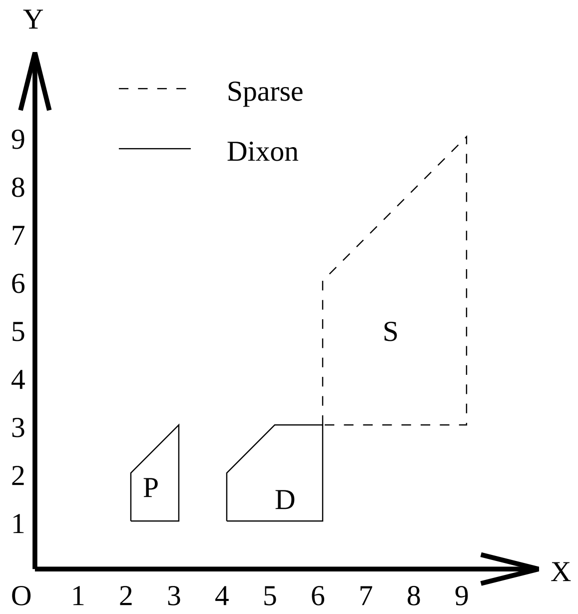

Figure 1: Various polytopes for the system P

The concept of an n-fold mixed volume provides a good quantitative basis to compare different resultant formulations. Given an unmixed set $P \subset \mathbb{Q}[X]$, its **n-fold mixed volume**, denoted $mvol(P)$, is $mvol(P) = n!\ vol(\mathcal{N}(P))$. Following theorem relates the size of the Dixon matrix to the n-fold mixed volume. Its proof is in Appendix A.

Theorem 3.8 *For asymptotically large r, the size of the Dixon matrix of a multi-homogeneous system P is $\mathcal{O}\left(\frac{mvol(P)}{n+1}\right)$.*

Following can be analogously derived for the sparse resultant matrix from Fact 3.6:

Theorem 3.9 *For asymptotically large r, size of the sparse resultant matrix of a multi-homogeneous system P is $\mathcal{O}\left(e^{n+1}\ mvol(P)\right)$.*

For multi-homogeneous systems, for asymptotically large r, the Dixon matrix is smaller than the sparse resultant matrix by a factor exponential in n. It can be shown that the sparse resultant matrix is, in turn, smaller than the Macaulay matrix by a factor exponential in n.

The entries in the Dixon matrix are however complicated in contrast to the entries in the sparse resultant matrix and Macaulay matrix, for which the entries are either 0 or coefficients of the monomials in the polynomials. In the next section, we analyze the complexity of constructing a Dixon matrix.

4 Construction of the Dixon Matrix

Dixon matrix can be constructed by expanding the determinant of the cancellation matrix and dividing it by $\prod_{i=1}^{n}(x_i - \bar{x}_i)$. Since the entries of the cancellation matrix are multivariate polynomials, direct expansion can be quite expensive, and give rise to large intermediate expressions. This can be avoided by using dense multivariate polynomial interpolation [14].

Interpolation works in stages. In each stage, polynomial is interpolated in one more variable using bounds on the number of terms in the polynomial (when viewed as a polynomial in the variables which have already been lifted) and its degree in the variable being lifted. Since the Dixon polynomial has $2n$ variables, namely X and $\bar{X}$, $2n$ stages are needed. Using analysis in Section 3 (Corollary 3.5), tight bounds on the number of terms in the Dixon polynomial, and its degree in different variables, can be derived and used in each stage of interpolation. The complexity of our variant of interpolation depends on the volume of the polytopes since that is the quantity which determines the number of terms in the Dixon polynomial, and hence, the cost of each stage. The complexity of construction for multi-homogeneous systems is given by the following theorem whose proof is in Appendix B.

Theorem 4.1 *The construction of the Dixon matrix of a multi-homogeneous system, P, requires $\mathcal{O}\left(\frac{n^3\ l!^{3r}}{n!}\ mvol\,(P)^3\right)$ arithmetic operations.*

The use of interpolation to construct Dixon matrices is possible because its entries are coefficients of a polynomial, which can be interpolated from input polynomials. This is not possible for sparse resultant matrix since its entries do not have such a relationship with the input polynomials.

4.1 Total Cost of Computing the Resultant using Dixon Formulation

Computing the resultant using Dixon formulation involves two steps: (i) constructing the Dixon matrix, and (ii) computing the determinant of the Dixon matrix. For a fixed number of variables, the size of the Dixon matrix is proportional to the n-fold mixed volume, thus, computing its determinant takes at most $\mathcal{O}(mvol(P)^3)$ arithmetic operations. From the above theorem, the construction cost, for a fixed number of variables, is $\mathcal{O}(mvol(P)^3)$. Consequently, for a fixed number of variables, the total cost of computing the resultant of a multi-homogeneous system using the Dixon formulation is $\mathcal{O}(mvol(P)^3)$.

4.2 Comparison with Sparse Resultant Formulation

The construction cost of sparse resultant matrix involves two measures of sparsity. First is the mixed volume, which we have used throughout this paper, and the other is the number of terms with non-zero coefficients in the polynomials. The following theorem from [7] gives the cost of constructing the sparse resultant matrix.

Fact 4.2 *If the number of monomials with non-zero coefficients in each polynomial of a multi-homogeneous system P is m, then the complexity of constructing its sparse resultant matrix is $\mathcal{O}\left(e^n\,(m\,n)^{5.5}\ mvol\,(P)\right)$.*

The sparse resultant formulation has an advantage that its construction cost is sensitive to both, the mixed volume, as well as the cardinality of the support of input polynomials (the traditional measure of sparsity). The construction of sparse resultant matrix can, in certain cases, take less time than the construction of the Dixon matrix. However, even in these cases, total cost of computing the resultant using sparse resultant formulation is more than Dixon formulation because the size of sparse resultant matrix is unaffected by the cardinality of support.

We consider two extreme cases: (i) *full* polynomials, in which all terms inside the Newton polytope have non-zero coefficients, and (ii) *hollow* polynomials, in which only the terms on the boundary of the polytopes have non-zero coefficients.

Ex	Application	No. of Polys	Dixon			Macaulay			Sparse		
			Matrix Size	Cnstr. Time	RSC Time	Matrix Size	GCP Time	RSC Time	Matrix Size	Cnstr. Time	RSC Time
1	Geom. Reason.	3	13	0.18	342	55	*	4608	34	0.3	566
2	Formula Deriv.	5	14	0.36	76	330	*	*	86	9.8	4094
3	Formula Deriv.	3	5	0.08	*	15	N/R	*	14	0.2	*
4	Geometry	3	4	0.08	1021	15	*	*	14	0.2	*
5	Random	4	6	0.06	*	84	N/R	*	27	0.7	*
6	Implicitization	3	10	0.1	0.58	45	3741	13.15	16	0.2	0.71
7	Implicitization	3	18	0.45	47	66	N/R	604	55	2	519
8	Implicitization	3	18	9.33	126	153	*	94697	54	1.9	1369
9	Random	4	7	0.21	220	56	*	9972	24	1.2	412
10	Random	5	36	1.1	8130	1365	*	*	151	40	201558
11	Implicitization	3	11	0.06	3.01	36	N/R	32.58	33	0.7	31.04
12	Equilibrium	4	5	0.05	0.24	20	N/R	1.53	20	0.7	2.36
13	Vision	6	6	3.3	0.05	792	*	*	60	248	10.53
14	Comput. Bio.	3	8	0.067	0.27	28	11.18	0.68	16	0.2	0.33

Table 1: Empirical Data

Full Systems: In a full system P, $m = \mathcal{O}(vol(P))$. Using this, the complexity of constructing its sparse resultant matrix is $\mathcal{O}\left(\frac{e^{6.5n}}{n^{5.5(n-1)}}\ (mvol(P))^{6.5}\right)$. For a fixed number of variables, since the cost of computing the determinant of the sparse resultant matrix is $\mathcal{O}(mvol(P))^3$ (from Theorem 3.9), the construction cost dominates the total cost of computing the resultant. Hence, the total cost of computing the resultant of a *full* multi-homogeneous system using sparse resultant formulation is $\mathcal{O}(mvol(P)^{6.5})$. However, the Dixon formulation (Section 4.1) takes $\mathcal{O}(mvol(P)^3)$ arithmetic operations, which is better than sparse.

Hollow Systems: In this case, we can assume that $m = \mathcal{O}(1)$ in Fact 4.2. So the construction cost of the sparse resultant matrix, in this case, is proportional to the n-fold mixed volume. This is better than the construction cost of the Dixon matrix, and if only a resultant matrix is desired, the sparse formulation is preferable. However, while computing the resultant, the determinant computation step dominates. Because of larger size of the sparse resultant matrix, the Dixon formulation again computes the resultant faster.

In both cases, more so in the full case, the Dixon formulation is less expensive than the sparse resultant formulation. To see typical timings in practice, see Table 4.2 (from [10]) in which the timings of Dixon, sparse and Macaulay formulations are given on 14 examples, mostly from robotics, vision, geometry and graphics. The coefficients in these systems are polynomials in parameters and not just integers. For sparse and Dixon formulations, the matrix construction times as well as the determinant computation time (called RSC, which uses interpolation to compute the determinant of the resultant matrices) are separately given. We wish to point out that for implementational simplicity, we used direct matrix expansion of the cancellation matrix for constructing Dixon matrix, and this has worse complexity than the interpolation method discussed in this paper. For Macaulay formulation, we give two timings, one using the Generalized Characteristic Method of Canny [2], and the other using a method similar to one in Theorem 2.2. As can be observed, Dixon matrix is much smaller and the Dixon formulation performs much better than others and was able to solve all 14 problems reasonably well. This work is the result of an effort to seek theoretical justification for these empirical observations.

References

[1] Bernshtein D.N., The Number of Roots of a System of Equations, *Funktsional'nyi Analiz i Ego Prilozheniya,* 9(3):1-4.

[2] Canny J., Generalized Characteristic Polynomials, *Journal of Symbolic Computation*, 9:241-250.

[3] Canny J., Pedersen P., An Algorithm for Newton Resultant, Technical Report, *Cornell University*, CornellCS:TR93-1394, 1993.

[4] Dixon A.L., The eliminant of three quantics in two independent variables. *Proc. London Mathematical Society,* 6, 1908, 468-478.

[5] Ehrart E. Sur un problème de géométrie diophantienne, I. Polyèdres et réseaux., *J. Reine Angew. Math.*, 226:1-29, 1967.

[6] Gelfand I.M., Kapranov M.M., Zelevinsky A.V., *Discriminants, Resultants and Multidimensional Determinants*, Birkhäuser, Boston, 1994.

[7] Emiris I., *Sparse Elimination and Applications in Kinematics*, Doctoral dissertation thesis, Department of Computer Science, U of Calif., Berkeley, 1994.

[8] Hoffman C.M., Algebraic and Numerical Techniques for Offsets and Blends, *Computation of Curves and Surfaces* W. Dahmen et. al. (eds.), Kluwer Acad. Pub., 1990, 499-528.

[9] Kapur D., Saxena T., Yang L., Algebraic and Geometric Reasoning using Dixon Resultants, Proc. *ACM ISSAC 94,* Oxford, England, July 1994.

[10] Kapur D., Saxena T., Comparison of Various Multivariate Resultants, To appear in Proc. *ACM ISSAC 95*, Montreal, Canada, July 1995.

[11] Sturmfels B., Sparse Elimination Theory, *Proc. Computat. Algebraic Geom. and Commut. Algebra*, D. Eisenbud and L. Robbiano, eds., Cortona , Italy, June 1991.

[12] Macaulay F.S., *The Algebraic Theory of Modular Systems*, Cambridge Tracts in Math. and Math. Phys., 19, 1916.

[13] Weyman J. and Zelevinsky A., Determinantal Formulas for Multigraded Resultants, *Journal of Algebraic Geometry*, pp 569-597, 1994.

[14] Zippel R., *Effective Polynomial Computation*, Kluwer Academic Publishers, Boston, 1993.

A Proof of Theorem 3.8

Let P be a multi-homogeneous polynomial set of type $(l_1, \ldots, l_r; d_1, \ldots, d_r)$ with $n+1$ polynomials, where $n = \sum_{i=1}^{r} l_i$ is the total number of variables. Then,

$$\begin{aligned} vol\,(\mathcal{N}\,(P)) &= \prod_{i=1}^{r} \frac{d_i^{l_i}}{l_i!}, \\ mvol\,(P) &= n!\; vol\,(\mathcal{N}\,(P)) \\ \text{Size of Dixon matrix} &= \mathcal{O}\left(\frac{\prod_{i=1}^{r} \left(\sum_{j=1}^{i} l_j + 1\right)^{l_i}}{n+1}\, vol\,(\mathcal{N}\,(P)) \right). \end{aligned}$$

Assuming $l_i = \mathcal{O}(l)$, using the Stirling approximation for asymptotically large r, the proof of Theorem 3.8 follows:

$$\begin{aligned} \frac{\text{Size of Dixon matrix}}{mvol\,(P)} &= \frac{\prod_{i=1}^{r} (i\,l+1)^l}{(rl+1)\,(r\,l)!} = \mathcal{O}\left(\frac{e^{rl}}{rl+1} \left(\prod_{i=1}^{r} \frac{il+1}{rl} \right)^l \right) = \mathcal{O}\left(\frac{e^{rl}}{rl+1} \left(\prod_{i=1}^{r} \frac{i}{r} \right)^l \right) \\ &= \mathcal{O}\left(\frac{e^{rl}}{rl+1} \left(\frac{r!}{r^r} \right)^l \right) = \mathcal{O}\left(\frac{e^{rl}}{rl+1e^{rl}} \right) = \mathcal{O}\left(\frac{1}{n+1} \right). \end{aligned}$$

B Proof of Theorem 4.1

The polytopes of all multi-homogeneous systems with n variables with partition degree d are subsets of the polytope of the system $(\underbrace{1, \ldots, 1}_{n}; \underbrace{d, \ldots, d}_{n})$. Since the size of the Dixon matrix depends only on the input polytopes, the Dixon matrix is the largest for $(1, \ldots, 1; d, \ldots, d)$. Consequently, the cost of construction the Dixon matrix for such a system bounds the cost of constructing the Dixon matrix for all other systems with n variables and partition degree d.

Let the n variables of the system $(1, \ldots, 1; d, \ldots, d)$ be $x_1, \ldots, x_n$ and the corresponding substitution variables be $\bar{x}_1, \cdots, \bar{x}_n$. For $1 \le i \le n$, one can easily derive from the analysis in section 3 that for the system $(1, \ldots, 1; d, \ldots, d)$:

1. The degree of $\delta_P$in x_i is $id - 1$, and hence, degree in $\bar{x}_{n-i+1}$ is also $id - 1$.

2. The number of terms in the variables $x_1, \ldots, x_i$ (regarding all other variables as constants) are bounded by $i!\, d^i$.

We use dense multivariate interpolation to compute the Dixon polynomial (which will directly give us the Dixon matrix). Given $D+1$ values of a univariate polynomial, interpolating them to produce the polynomial takes $I(D) = \mathcal{O}(D^2)$ time [14]. Let the cost of computing a single value of the univariate polynomial from the cancellation matrix be B. Computing a value requires (i) substituting values for each variable in each entry of the cancellation matrix which takes $\mathcal{O}(n^3 d^n)$ arithmetic operations and (ii) computing its determinant, which takes $\mathcal{O}(n^3)$ operations. Hence, $B = \mathcal{O}(n^3 d^n)$.

Multivariate interpolation works in stages; in each stage one variable is lifted. Since the Dixon polynomial has $2n$ variables, namely X and $\bar{X}$, we need $2n$ stages to interpolate it. We lift the variables in the order $x_1, \bar{x}_n, x_2, \bar{x}_{n-1}, \ldots, x_n, \bar{x}_1$. We call the k^{th} variable in this sequence, y_k, eg.

$y_3 = x_2$ and $y_4 = \bar{x}_{n-1}$. Let N_k be the complexity of the first k stages. Now, the k^{th} stage involves (i) interpolating the polynomial in $k-1$ variables as many times as the $deg(\delta_P, y_k)+1$ and (ii) interpolating as many univariate polynomials (in y_k) as the number of terms in δ_P in variables $y_1, \ldots, y_{k-1}$. Since, in the first stage, one needs to compute the values of the Dixon polynomial directly by computing the determinant of the Dixon matrix, we get the following recurrence relations for our $2n$ stages:

$$\begin{aligned} N_1 &= d\,B + I(d), \\ N_k &= k_c\, d\, N_{k-1} + \left(k_f!\, d^{k_f}\right) \left((k_c-1)!\, d^{k_c-1}\right) I\left(k_c d\right), \end{aligned}$$

where, $k_c = \lceil k/2 \rceil$ and $k_f = \lfloor k/2 \rfloor$. The solution to this recurrence is

$$N_{2n} = n!^2 d^{2n} \left(B + \frac{2}{d} \sum_{i=1}^{n} \frac{I\left(id\right)}{i} \right) = \mathcal{O}\left(n!^2 d^{2n} \left(n^3 d^n + \frac{2}{d} \sum_{i=1}^{n} \frac{i^2 d^2}{i} \right) \right) = \mathcal{O}\left(n^3 n!^2 d^{3n} \right),$$

therefore,

$$\frac{N_{2n}}{mvol(P)^3} = n^3 n!^2 d^{3n} \frac{l!^{3r}}{n!^3 d^{3n}} = \frac{n^3 l!^{3r}}{n!},$$

and the theorem is proved.