Constructing Sylvester-Type Resultant Matrices using the Dixon Formulation *

Arthur Chtcherba Deepak Kapur

Department of Computer Science
University of New Mexico
Albuquerque, NM 87131
e-mail: {*artas,kapur*}*@cs.unm.edu*

August 23, 2002

Abstract

A new method for constructing Sylvester-type resultant matrices for multivariate elimination is proposed. Unlike sparse resultant constructions discussed recently in the literature or the Macaulay resultant construction, the proposed method does not explicitly use the support of a polynomial system in the construction. Instead, a multiplier set for each polynomial is obtained from the Dixon formulation using an arbitrary term for the construction. As shown in [KS96], the generalized Dixon resultant formulation implicitly exploits the sparse structure of the polynomial system. As a result, the proposed construction for Sylvester-type resultant matrices is **sparse** in the sense that the matrix size is determined by the support structure of the polynomial system, instead of the total degree of the polynomial system.

The proposed construction is a generalization of a related construction proposed by the authors in which the monomial 1 is used [CK00b]. It is shown that any monomial (inside or outside the support of a polynomial in the polynomial system) can be used instead insofar as that monomial does not vanish on any of the common zeros of the polynomial system. For generic unmixed polynomial systems (in which every polynomial in the polynomial system has the same support, i.e., the same set of terms), it is shown that the choice of a monomial does not affect the matrix size insofar as it is in the support.

The main advantage of the proposed construction is for mixed polynomial systems. Supports of a mixed polynomial system can be translated so as to have a maximal overlap, and a term is selected from the overlapped subset of translated supports. Determining an appropriate translation vector for each support and a term from overlapped support can be formulated as an optimization problem. It is shown that under certain conditions on the supports of polynomials in a mixed polynomial system, a monomial can be selected leading to a Dixon multiplier matrix of the smallest size, thus implying that the projection operator computed using the proposed construction is either the resultant or has an extraneous factor of minimal degree.

The proposed construction is compared theoretically and empirically, on a number of examples, with other methods for generating Sylvester-type resultant matrices.

KEYWORDS: Resultant, Dixon Method, Bezoutians, Sylvester-type matrices, Dialytic Method, Multiplier Matrices, BKK Bound, Support

1 Introduction

Resultant matrices based on the Dixon formulation have turned out to be quite efficient in practice for simultaneously eliminating many variables on a variety of examples from different application domains; for details and comparison with other resultant formulations and elimination methods, see [KS95, CK02b] and `http://www.cs.unm.edu/~artas`. Necessary conditions can be derived on parameters in a problem formulation under which the associated polynomial system has a solution.

*This research is supported in part by NSF grant nos. CCR-0203051, CDA-9503064, and a grant from the Computer Science Research Institute at Sandia National Labs.

Sylvester-type multiplier matrices based on the Dixon formulation are introduced using a general construction which turns out to be effective especially for mixed polynomial systems. This construction generalizes a construction discussed in our previous work [CK00b]. Multiplier sets for each polynomial in a given polynomial system are computed, generating a matrix whose determinant (or the determinant of a maximal minor) includes the resultant. Unlike other Sylvester-type matrix constructions which explicitly use the support of a polynomial system, the proposed construction uses the support only implicitly insofar as the Dixon formulation implicitly exploits the sparse support structure of a polynomial system as proved in [KS96]. The proposed construction for Sylvester-type resultant matrices is thus **sparse** in the sense that the matrix size is determined by the support structure of the polynomial system, instead of the total degree of the polynomial system.

It is shown that an arbitrary monomial can be used to do the proposed construction; the only requirement is that the monomial does not vanish on any of the common zeros of the polynomial system. For the generic unmixed case (in which each polynomial in the polynomial system has the same support, i.e., the same set of terms), this construction is shown to be *optimal* if the monomial used is from the support of the polynomial system. To be precise, given a generic unmixed polynomial system, if the Dixon formulation produces a Dixon matrix whose determinant is the resultant, then the Sylvester-type multiplier matrices (henceforth, called the *Dixon multiplier* matrices) based on the proposed construction also have the resultant as their determinants. In case the Dixon matrix is such that the determinant of the maximal minor has an extraneous factor besides the resultant, the Dixon multiplier matrix does not have an extraneous factor of higher degree. Thus, no additional extraneous factor is contributed to the result by the proposed construction.

For mixed polynomial systems, the proposed construction works especially well. Conditions are identified on the supports of the polynomials in a mixed polynomial system which enable the selection of a term producing a Dixon multiplier matrix of the smallest size. The projection operator computed from this matrix is either the resultant or has an extraneous factor of minimal degree. Heuristics are developed for selecting an appropriate monomial for the construction in case of mixed polynomial systems which do not satisfy such conditions. Supports are first translated so that they have maximal overlap, providing a large choice of possible terms to be used for generating multiplier sets. Determining translation and selecting a term from the translated supports are formulated as an optimization problem.

The main advantage of using the Dixon multiplier matrices over the associated Dixon matrices is (i) in the mixed case, the Dixon multiplier matrices can have resultants as their determinants, whereas the Dixon matrices often have determinants which includes along with the resultants, extraneous factors; (ii) if the determinant of a Dixon multiplier matrix has an extraneous factor with the resultant, the degree of the extraneous factor is lower than the degree of the extraneous factor appearing in the determinant of the Dixon matrix; (iii) the Dixon multiplier matrices can be stored and computed more efficiently, given that the entries are either zero or the coefficients of the monomials in the polynomials; this is in contrast to the entries of the Dixon matrices which are determinants in the coefficients.

The next section discusses preliminaries and background – the concept of a multivariate resultant of a polynomial system, the support of a polynomial, the degree of the resultant as determined by the bound developed in a series of papers by Kouchnirenko, Bernstein and Khovanski (also popularly known as the **BKK** bound), based on the mixed volume of the Newton polytopes of the supports of the polynomials in a polynomial system, Sylvester-type resultant matrices. Section 3 is a review of the generalized Dixon formulation; the Dixon polynomial and Dixon matrix are defined; using the Cauchy-Binet expansion of determinants, the Dixon polynomial and its support are related to the support of the polynomials in the polynomial system.

Section 4 gives the construction for Sylvester-type resultant matrices using the Dixon formulation. Theorem 4.1 serves as the basis of this construction. As the reader would notice, this construction uses an arbitrary monomial, instead of a construction in [CK00b] where the particular monomial 1 was used; the only requirement on the selected monomial is that it should not vanish on any of the common zeros of the polynomial system. In the case a Dixon multiplier matrix is not exact, i.e., its determinant is a nontrivial multiple of the resultant, it is shown how a maximal minor of the matrix can be used for computing the projection operator. It is also proved that whenever the Dixon matrix obtained from the generalized Dixon formulation can be used to compute the resultant exactly (up to a sign), the Dixon multiplier matrix can also be used to compute the resultant exactly.

Section 5 discusses how an appropriate monomial can be chosen for the construction so as to minimize the Dixon multiplier matrix for a given polynomial system and consequently, the degree of the extraneous factor. For unmixed polynomial systems, it is shown that choosing any monomial in the support will lead to the

Dixon multiplier matrices of the same size. The heuristic for selecting a monomial for constructing the Dixon multiplier matrix is especially effective in the case of mixed systems. It is shown that monomials common to all the polynomials in a given polynomial system are good candidates for the construction. Supports of polynomials of a polynomial system can be translated so as to maximize the overlap among them. An example is discussed illustrating why translation of the supports of the polynomials in a polynomial system is crucial for getting Dixon multiplier matrices of smaller size.

The construction is compared theoretically and empirically with other methods for generating sparse resultant matrices, including the subdivision method [CE00] and the incremental method [EC95].

Section 7 discusses an application of the Dixon multiplier construction to multi-graded systems. It is proved that the proposed construction generates exact matrices for families of generic unmixed systems including multi-graded systems, without any a priori knowledge about the structure of such polynomial systems.

2 Multivariate Resultant of a Polynomial System

Consider a system of polynomial equations $\mathcal{F} = \{f_0, \ldots, f_d\}$,

$$f_0 = \sum_{\alpha\in\mathcal{A}_0} c_{0,\alpha}\mathbf{x}^\alpha, \quad f_1 = \sum_{\alpha\in\mathcal{A}_1} c_{1,\alpha}\mathbf{x}^\alpha, \quad \cdots, \quad f_d \;=\; \sum_{\alpha\in\mathcal{A}_d} c_{d,\alpha}\mathbf{x}^\alpha,$$

and for each $i = 0, \ldots, d$, $\mathcal{A}_i \subset \mathbb{N}^d$ and $k_i = |\mathcal{A}_i| - 1$, $\mathbf{x}^\alpha = (x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_d^{\alpha_d})$ where $(c_{i,\alpha})$ are parameters. We will denote by $\mathcal{A} = \langle \mathcal{A}_0, \ldots, \mathcal{A}_d\rangle$, the **support** of the polynomial system $\mathcal{F}$.

The goal is to derive condition on parameters $(c_{i,\alpha})$ so that the polynomial system $\mathcal{F} = 0$ has a solution. One can view this problem as the elimination of variables from the polynomial system. Elimination theory tells that such a condition exists for a large family of polynomial systems, and is called the *resultant* of the polynomial system. Since the number of equations is more than the number of variables, in general, for arbitrary values of $c_{i,\alpha}$, the polynomial system $\mathcal{F}$ does not have any solution. The resultant of the above polynomial system can be defined as follows [EM99]. Let

$$W_V = \Big\{(\mathbf{c}, \mathbf{x}) \in \mathbb{P}^{k_0} \times \cdots \times \mathbb{P}^{k_d} \times V \;\mid\; f_i(\mathbf{c}, \mathbf{x}) = 0 \quad \text{for all} \quad i = 0, 1, \ldots, d\Big\},$$

where $\mathbf{c} = \langle c_{0,\alpha_0}, \ldots, c_{0,\alpha_{k_0}}, \ldots, c_{d,\alpha_0}, \ldots, c_{d,\alpha_{k_d}}\rangle$, and V is a projective subvariety of $\underline{\dim d}$. W_V is an algebraic set, or a projective variety. Consider the following projections of this variety:

$$\begin{aligned} \pi_1 : W_V &\rightarrow \mathbb{P}^{k_0} \times \mathbb{P}^{k_1} \cdots \times \mathbb{P}^{k_d}, \\ \pi_2 : W_V &\rightarrow V, \end{aligned}$$

$\pi_1(W_V)$ is the set of all values of the parameters such that the above system of polynomial equations has a solution. Since W_V is a projective variety, any projection of it is also a projective variety (see [Sha94]). Therefore, there exists a set of polynomials defining $\pi_1(W_V)$. If there is only one such polynomial, then $\pi_1(W_V)$ is a hypersurface, and its defining equation is called the **resultant**. If V is d dimensional, then for generic coefficients, any d equations have a finite number of solutions; consequently $\pi_1(W_V)$ is a hypersurface (see [EM99] and [BEM00]).

Definition 2.1 *If variety $\pi_1(W_V)$ is a hypersurface, then its defining equation will be called the resultant of $\{f_0, f_1, \ldots, f_d\}$ over V, denoted as $\mathrm{Res}_V(f_0, \ldots, f_d)$.*

In the above definition, the resultant is dependent on the choice of the variety V. Different resultant construction methods do not define explicitly the variety V, and we assume it to be the projective closure of some affine set.

The variety $\pi_1(W_V)$ is a hypersurface whenever V is d dimensional; for generic values of the coefficients, any d equations have a finite number of solutions (see discussion in [EM99] and [BEM00]).

The degree of the resultant is determined by the number of roots the polynomial system has in a given variety V. For simplicity, in this article, we will assume that V is a projective closure of $(\mathbb{C}^*)^d$ or toric variety[1]. Results of [BEM00] extend these notions to projective closures of affine open spaces, which can be parameterized, hence giving much more general applicability of the method.

[1] The set $(\mathbb{C}^*)^d$ is d-dimensional set where coordinates cannot have zero values, that is $\mathbb{C}^* = \mathbb{C} - \{0\}$.

2.1 Support and Degree of the Resultant

The convex hull of the support of a polynomial f is called its Newton polytope, and will be denoted as $\mathcal{N}(f)$. One can relate the Newton polytopes of a polynomial system to the number of its roots.

Definition 2.2 ([GKZ94],[CLO98]) *The mixed volume function $\mu(\mathcal{Q}_1, \ldots, \mathcal{Q}_d)$, where $\mathcal{Q}_i$ is a convex hull, is a unique function which is multilinear with respect to Minkowski sum and scaling, and is defined to have the multilinear property*

$$\mu(\mathcal{Q}_1, \ldots, a\mathcal{Q}_k + b\mathcal{Q}'_k, \ldots, \mathcal{Q}_d) = a\,\mu(\mathcal{Q}_1, \ldots, \mathcal{Q}_k, \ldots, \mathcal{Q}_d) + b\,\mu(\mathcal{Q}_1, \ldots, \mathcal{Q}'_k, \ldots, \mathcal{Q}_d);$$

to ensure uniqueness, $\mu(\mathcal{Q}, \ldots, \mathcal{Q}) = d!\mathsf{Vol}(\mathcal{Q})$, where $\mathsf{Vol}(\mathcal{Q})$ is the Euclidean volume of the polytope $\mathcal{Q}$.

Theorem 2.1 (BKK Bound) *Given a polynomial system $\{f_1, \ldots, f_d\}$ in d variables $\{x_1, \ldots, x_d\}$ with the support $\langle \mathcal{A}_1, \ldots, \mathcal{A}_d \rangle$, the number of roots in $(\mathbb{C}^*)^d$, counting multiplicities, of the polynomial system is either infinite or $\leq \mu\,(\mathcal{A}_1, \ldots, \mathcal{A}_d)$; furthermore, the inequality becomes equality when the coefficients of polynomials in the system satisfy genericity requirements.*

Since we are interested in overconstrained polynomial systems, usually consisting of $d+1$ polynomials in d variables, the BKK bound also tells us the degree of the resultant.

In the resultant, the degree of the coefficients of f_0 is equal to the number of common roots the rest of polynomials have. It is possible to choose any f_i and the resultant expression can be expressed by substituting in f_i the common roots of the remaining polynomial system, [PS93]. This implies that the degree of the coefficients of f_i in the resultant equals the number of roots of the remaining set of polynomials. We denote the BKK bound of a $d+1$ polynomial system by $\langle b_0, b_1, \ldots, b_d \rangle$ as well as B, where $b_i = \mu(\mathcal{A}_0, \ldots, \mathcal{A}_{i-1}, \mathcal{A}_{i+1}, \ldots, \mathcal{A}_d)$ and $B = \sum_{i=0}^{d} b_i$.

2.2 Resultant Matrices

One way to compute the resultant of a given polynomial system is to construct a matrix with a property that whenever the polynomial system has a solution, such a matrix has a deficient rank, thereby implying that determinant of any maximal minor is a multiple of the resultant. The BKK bound imposes a lower bound on the size of such a matrix.

A simple way to construct a resultant matrix is to use the *dialytic* method, i.e., multiply each polynomial with a finite set of monomials, and rewrite the resulting system in the matrix form. We call such a matrix the *multiplier matrix*. This alone, however, does not guarantee that a matrix so constructed is a resultant matrix.

Definition 2.3 *Given a set of polynomials $\{f_1, \ldots, f_k\}$ in variables $x_1, \ldots, x_d$ and finite monomials sets $X_1, \ldots, X_k$, where $X_i = \{\ \mathbf{x}^\alpha \mid \alpha \in \mathbb{N}^d\ \}$, denote by $X_i f_i = \{\ \mathbf{x}^\alpha f_i \mid \mathbf{x}^\alpha \in X_i\}$. The matrix representing the polynomial system $X_i f_i$ for all $i = 1, \ldots, k$, can be written as*

$$\begin{pmatrix} X_1 f_1 \\ X_2 f_2 \\ \vdots \\ X_k f_k \end{pmatrix} = M \times X = 0,$$

*where $X^T = \left(\mathbf{x}^{\beta_1}, \ldots, \mathbf{x}^{\beta_l}\right)$ such that $\mathbf{x}^\beta \in X$ if there exist i such that $\mathbf{x}^\beta = \mathbf{x}^\alpha \mathbf{x}^\gamma$ where $\mathbf{x}^\alpha \in X_i$ and $\mathbf{x}^\gamma \in f_i$. Such matrices will be called as the **multiplier matrices**.*

If a given multiplier matrix is non-singular, and its corresponding polynomial system has a solution which does not make X identically zero, then its determinant is a multiple of the resultant. Furthermore, the requirement on the matrix to be non-singular (or even square) can be relaxed, as long as it can be shown that its rank becomes deficient whenever there exist a solution; in such cases, the multiple of the resultant can be extracted from a maximal minor of this matrix.

Note that such matrices are usually quite sparse: matrix entries are either zero or coefficients of the polynomials in the original system. Good examples of resultant multiplier matrices are *Sylvester* [Syl53] for the univariate case, and *Macaulay* [Mac16] as well Newton sparse matrices of [CE00] for the multivariate case; they all differ only in the selection of multiplier sets X_i.

If the BKK bound of a given polynomial system is $\langle b_0, b_1, \ldots, b_d \rangle$, then $|X_i| \geq b_i$. The matrix size must be at least B (the sum of all the b_i's) for it to be a candidate for the resultant matrix of the polynomial system.

In the following sections, we show how the Dixon formulation can be used to construct multiplier matrices for the multivariate case. We first give a brief overview of the Dixon formulation, define the concepts of the Dixon polynomial and the Dixon matrix of a given polynomial system. Expressing the Dixon polynomial using the Cauchy-Binet expansion of determinants of a matrix turns out to be useful for illustrating the dependence of the construction on the support of a given polynomial system.

3 The Dixon Matrix

In [Dix08], Dixon generalized Bezout-Cayley's construction for computing resultant of two univariate polynomials to the bivariate case. In [KSY94], Kapur, Saxena and Yang further generalized this construction to the general multivariate case; the concepts of a Dixon polynomial and a Dixon matrix were introduced as well. Below, the generalized multivariate Dixon formulation for simultaneously eliminating many variables from a polynomial system and computing its resultant is reviewed. More details can be found in [KS95].

In contrast to multiplier matrices, the Dixon matrix is dense since its entries are determinants of the coefficients of the polynomials in the original polynomial system. It has the advantage of being an order of magnitude smaller in comparison to a multiplier matrix, which is important as the computation of the determinant of a matrix with symbolic entries is sensitive to its size, see table 2 in section 6. The Dixon matrix is constructed through the computation of the Dixon polynomial, which is expressed in matrix form.

Let $\pi_i(\mathbf{x}^\alpha) = \overline{x}_1^{\alpha_1} \cdots \overline{x}_i^{\alpha_i} x_{i+1}^{\alpha_{i+1}} \cdots x_d^{\alpha_d}$, where $i \in \{0, \ldots, d\}$, and $\overline{x}_i$'s are new variables; $\pi_0(\mathbf{x}^\alpha) = \mathbf{x}^\alpha$. π_i is extended to polynomials in a natural way as: $\pi_i(f(x_1, \ldots, x_d)) = f(\overline{x}_1, \ldots, \overline{x}_i, x_{i+1}, \ldots, x_d)$.

Definition 3.1 *Given a polynomial system $\mathcal{F} = \{f_0, f_1, \ldots, f_d\}$, where $\mathcal{F} \subset \mathbb{Q}[\mathbf{c}][x_1, \ldots, x_d]$, define its **Dixon polynomial** as*

$$\theta(f_0, \ldots, f_d) = \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \begin{vmatrix} \pi_0(f_0) & \pi_0(f_1) & \cdots & \pi_0(f_d) \\ \pi_1(f_0) & \pi_1(f_1) & \cdots & \pi_1(f_d) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \pi_d(f_1) & \cdots & \pi_d(f_d) \end{vmatrix}.$$

Hence $\theta(f_0, f_1, \ldots, f_d) \in \mathbb{Q}[\mathbf{c}][x_1, \ldots, x_d, \overline{x}_1, \ldots, \overline{x}_d]$, where $\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_d$ are new variables.

The order in which original variables in $\mathbf{x}$ are replaced by new variables in $\overline{\mathbf{x}}$ is significant in the sense that the Dixon polynomial computed using two different variable orderings may be different.

Definition 3.2 *A Dixon polynomial $\theta(f_0, \ldots, f_d)$ can be written in bilinear form as*

$$\theta(f_0, f_1, \ldots, f_d) = \overline{X}\, \Theta\, X^T,$$

*where $\overline{X} = (\overline{\mathbf{x}}^{\beta_1}, \ldots, \overline{\mathbf{x}}^{\beta_k})$ and $X = (\mathbf{x}^{\alpha_1}, \ldots, \mathbf{x}^{\alpha_l})$ are row vectors. The $k \times l$ matrix Θ is called the **Dixon matrix**.*

Each entry in Θ is a polynomial in the coefficients of the original polynomials in $\mathcal{F}$; moreover its degree in the coefficients of any given polynomial is at most 1. Therefore, the projection operator computed using the Dixon formulation can be of at most of degree $|X|$ in the coefficients of any single polynomial.

We will relate the support of a given polynomial system $\mathcal{A} = \langle \mathcal{A}_0, \ldots, \mathcal{A}_d \rangle$ to the support of its Dixon polynomial.

3.1 Relating Size of the Dixon Matrix to Support of Polynomial System

Given a collection of supports, it is often useful to construct a support which contains a single point from each support in the collection. A special notation is introduced for this purpose.

Definition 3.3 *Given a polynomial system support $\mathcal{A} = \langle \mathcal{A}_0, \ldots, \mathcal{A}_d \rangle$, let $\sigma = \langle \sigma_0, \sigma_1, \ldots, \sigma_d \rangle$ such that $\sigma_i \in \mathcal{A}_i$ for $i = 0, \ldots, d$. Denote this relation as $\sigma \Subset \mathcal{A}$; clearly $\{\sigma_0, \sigma_1, \ldots, \sigma_d\} \subset \mathbb{N}^d$; abusing the notation, σ is also treated a simplex.*

Using the above notation, we can express the support of the Dixon polynomial in terms of a sum of smaller Dixon polynomials.

Theorem 3.1 *[CK00a] Let $\mathcal{F} = \{f_0, f_1, \ldots, f_d\}$ be a polynomial system and let $\mathcal{A} = \langle \mathcal{A}_0, \ldots, \mathcal{A}_d \rangle$ be the support of $\mathcal{F}$. Then*

$$\theta(f_0, f_1, \ldots, f_d) = \sum_{\sigma \Subset \mathcal{A}} \sigma(\mathbf{c})\ \sigma(\mathbf{x}) = \sum_{\sigma \Subset \mathcal{A}} \theta_\sigma,$$

where $\theta_\sigma = \sigma(\mathbf{c})\ \sigma(\mathbf{x})$ and

$$\sigma(\mathbf{c}) = \begin{vmatrix} c_{0,\sigma_0} & c_{0,\sigma_1} & \cdots & c_{0,\sigma_d} \\ c_{1,\sigma_0} & c_{1,\sigma_1} & \cdots & c_{1,\sigma_d} \\ \vdots & \vdots & \ddots & \vdots \\ c_{d,\sigma_0} & c_{d,\sigma_1} & \cdots & c_{d,\sigma_d} \end{vmatrix} \quad \text{and} \quad \sigma(\mathbf{x}) = \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \begin{vmatrix} \pi_0(\mathbf{x}^{\alpha_{\sigma_0}}) & \pi_0(\mathbf{x}^{\alpha_{\sigma_1}}) & \cdots & \pi_0(\mathbf{x}^{\alpha_{\sigma_d}}) \\ \pi_1(\mathbf{x}^{\alpha_{\sigma_0}}) & \pi_1(\mathbf{x}^{\alpha_{\sigma_1}}) & \cdots & \pi_1(\mathbf{x}^{\alpha_{\sigma_d}}) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_d(\mathbf{x}^{\alpha_{\sigma_0}}) & \pi_d(\mathbf{x}^{\alpha_{\sigma_1}}) & \cdots & \pi_d(\mathbf{x}^{\alpha_{\sigma_d}}) \end{vmatrix}.$$

Proof: Let $\breve{\mathcal{A}} = \bigcup_{i=0}^{d} \mathcal{A}_i$ and $\breve{\mathcal{A}} = \{a_1, \ldots, a_n\}$. Let c_{i,a_j} be the coefficient of monomial $\mathbf{x}^{a_j}$ in polynomial f_i for $a_j \in \breve{\mathcal{A}}$, if monomial $\mathbf{x}^{a_j}$ does not appear in f_i then $c_{i,a_j} = 0$. Consider

$$\begin{aligned} \theta(f_0, \ldots, f_d) &= \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \begin{vmatrix} \pi_0(f_0) & \pi_0(f_1) & \cdots & \pi_0(f_d) \\ \pi_1(f_0) & \pi_1(f_1) & \cdots & \pi_1(f_d) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \pi_d(f_1) & \cdots & \pi_d(f_d) \end{vmatrix} \\ &= \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \begin{vmatrix} \sum_{i=1}^{n} c_{0,a_i}\pi_0(\mathbf{x}^{a_i}) & \sum_{i=1}^{n} c_{1,a_i}\pi_0(\mathbf{x}^{a_i}) & \cdots & \sum_{i=1}^{n} c_{d,a_i}\pi_0(\mathbf{x}^{a_i}) \\ \sum_{i=1}^{n} c_{0,a_i}\pi_1(\mathbf{x}^{a_i}) & \sum_{i=1}^{n} c_{1,a_i}\pi_1(\mathbf{x}^{a_i}) & \cdots & \sum_{i=1}^{n} c_{d,a_i}\pi_1(\mathbf{x}^{a_i}) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} c_{0,a_i}\pi_d(\mathbf{x}^{a_i}) & \sum_{i=1}^{n} c_{1,a_i}\pi_d(\mathbf{x}^{a_i}) & \cdots & \sum_{i=1}^{n} c_{d,a_i}\pi_d(\mathbf{x}^{a_i}) \end{vmatrix} \\ &= \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \det\left(M \times C\right), \end{aligned}$$

where

$$M = \begin{pmatrix} \pi_0(\mathbf{x}^{a_1}) & \pi_0(\mathbf{x}^{a_2}) & \cdots & \pi_0(\mathbf{x}^{a_{n-1}}) & \pi_0(\mathbf{x}^{a_n}) \\ \pi_1(\mathbf{x}^{a_1}) & \pi_1(\mathbf{x}^{a_2}) & \cdots & \pi_1(\mathbf{x}^{a_{n-1}}) & \pi_1(\mathbf{x}^{a_n}) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \pi_d(\mathbf{x}^{a_1}) & \pi_d(\mathbf{x}^{a_2}) & \cdots & \pi_d(\mathbf{x}^{a_{n-1}}) & \pi_d(\mathbf{x}^{a_n}) \end{pmatrix} \quad \text{and} \quad C = \begin{pmatrix} c_{0,a_1} & c_{1,a_1} & \cdots & c_{d,a_1} \\ c_{0,a_2} & c_{1,a_2} & \cdots & c_{d,a_2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{0,a_{n-1}} & c_{1,a_{n-1}} & \cdots & c_{d,a_{n-1}} \\ c_{0,a_n} & c_{1,a_n} & \cdots & c_{d,a_n} \end{pmatrix}.$$

Using the Cauchy-Binet expansion [AW92] for the determinant of a product of matrices, we can expand the above determinant into sum of product of determinants

$$\det\left(M \times C\right) = \sum_{1 \le i_0 < \cdots < i_d \le d+1} \det(M_{i_0,\ldots,i_d}) \det(C_{i_0,\ldots,i_d}),$$

where $M_{i_0,\ldots,i_d}$ is $(d+1) \times (d+1)$ submatrix of M containing columns $i_0, i_1, \ldots, i_d$ and $C_{i_0,\ldots,i_d}$ is a $(d+1) \times (d+1)$ submatrix of C containing rows $i_0, i_1, \ldots, i_d$. For multi-index $\{i_0, i_1, \ldots, i_d\}$, let $\sigma = \langle a_{i_0}, a_{i_1}, \ldots, a_{i_d} \rangle$, then,

$$\sigma(\mathbf{c}) = \det(C_{i_0,\ldots,i_d}) \qquad \text{and} \qquad \sigma(\mathbf{x}) = \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \det(M_{i_0,\ldots,i_d}).$$

Note that the order of σ_j's in $\sigma = \langle \sigma_0, \sigma_1, \ldots, \sigma_d \rangle$ does not change the product $\sigma(\mathbf{c})\sigma(\mathbf{x})$.

Below we show that it suffices to have the sum to be over all $\sigma \Subset \mathcal{A}$ where $\sigma_j = a_{i_j} \in \mathcal{A}_j$ for $j = 0, \ldots, d$.

Given an arbitrary σ, it is impossible to rearrange $\sigma = \langle \sigma_0, \sigma_1, \ldots, \sigma_d \rangle$ so that each $\sigma_j \in \mathcal{A}_j$ for $j = 0, \ldots, d$, then it is easy to see that $\sigma(\mathbf{c}) = 0$. Assume that for all rearrangements of σ_i's in σ, there is some $\sigma_i \notin \mathcal{A}_i$. That means that the entry in the column corresponding to σ_i and in the row i (on the diagonal) of the matrix of $\sigma(\mathbf{c})$ is zero. Rearranging σ_i's in σ amounts to permuting columns of $\sigma(\mathbf{c})$. Since it is impossible to rearrange σ so that $\sigma_i \in \mathcal{A}_i$, it is impossible to rearrange columns of $\sigma(\mathbf{c})$ so that diagonal does not have zero. Consequently, the determinant of $\sigma(\mathbf{c})$ is zero. Hence the expansion

$$\theta(f_0, \ldots, f_d) = \sum_{\sigma \Subset \mathcal{A}} \sigma(\mathbf{c})\ \sigma(\mathbf{x})$$

is a reduced version of the Cauchy-Binet expansion. □

The above identity shows that if generic coefficients are assumed in the polynomial system, then the support of the Dixon polynomial depends entirely on the support of the polynomial system, as $\sigma(\mathbf{c})$ would not vanish or cancel each other. To emphasize the dependence of θ on $\mathcal{A}$, the above identity can also be written as $\theta_{\mathcal{A}} = \sum_{\sigma \Subset \mathcal{A}} \theta_\sigma$.

We define the support of the Dixon polynomial as:

$$\Delta_{\mathcal{A}} = \{\, \alpha \mid \mathbf{x}^\alpha \in \theta(f_0, \ldots, f_d) \,\},^{2}$$

where $\mathcal{A} = \langle \mathcal{A}_0, \ldots, \mathcal{A}_d \rangle$ and $\mathcal{A}_i$ is the support of f_i. Let

$$\overline{\Delta}_{\mathcal{A}} = \left\{\, \beta \mid \overline{\mathbf{x}}^\beta \in \theta(f_0, \ldots, f_d) \,\right\}.$$

For the generic case, using the reduced Cauchy-Binet formula,

$$\Delta_{\mathcal{A}} = \bigcup_{\sigma \Subset \mathcal{A}} \Delta_\sigma, \qquad \text{where} \quad \Delta_\sigma = \{\, \alpha \mid \mathbf{x}^\alpha \in \theta_\sigma \,\},$$

because of genericity, θ_σ does not cancel any part of θ_τ for any $\sigma, \tau \Subset \mathcal{A}$ and $\sigma \neq \tau$.

One of the properties of $\sigma(\mathbf{x})$, we will use, is that

$$\Delta_\sigma = \{\, \alpha \mid \mathbf{x}^\alpha \in \theta_\sigma |_{\overline{x}_i = 1} \},$$

that is, substituting $\overline{x}_i = 1$ for $i = 1, \ldots, d$, does not change the support of the Dixon polynomial. This can be seen by noting that given a monomial in the expansion of the determinant of $\sigma(\mathbf{x})$ in terms of variables $x_1, \ldots, x_d$, its coefficient in terms of variables $\overline{x}_1, \ldots, \overline{x}_d$ can be uniquely identified. This is because each monomial in θ_σ is of the same degree in terms of variables $x_i, \overline{x}_i$. Hence, substituting $\overline{x}_i = 1$ will not cancel any monomials; if there was cancellation, it should happen without making any substitution.

4 Dixon Multiplier Matrix

We define a Dixon multiplier matrix which is related to the Dixon matrix in the same way as *Sylvester* matrix is related to *Bezout*'s. In fact the first relationship generalizes the second. This formulation also generalizes some of the earlier results which first appeared in [CK00b].

4.1 Construction

Let m be a monomial in variables $\{x_1, x_2, \ldots, x_d\}$. For abbreviation, let

$$\theta = \theta(f_0, f_1, \ldots, f_d), \quad \text{and also} \quad \theta_i(m) = \theta(f_0, \ldots, f_{i-1}, m, f_{i+1}, \ldots, f_d).$$

Recall that in [CK00b], $\theta_i = \theta_i(1) = \theta(f_0, \ldots, f_{i-1}, 1, f_{i+1}, \ldots, f_d)$.

[2] By abuse of notation, for some polynomial f, by $\mathbf{x}^\alpha \in f$ we mean that $\mathbf{x}^\alpha$ appears in (the simplified form of) f with a non-zero coefficient, i.e. α is in the support of f.

Theorem 4.1

$$m\ \theta(f_0,\ldots,f_d) = \sum_{i=0}^{d} f_i\ \theta_i(m).$$

Proof: Let $a_{i,j}$ and q_j for $i=0,\ldots,d$ and $j=1,\ldots d$ be arbitrary. In general, the following sum of determinants is zero, that is,

$$\begin{aligned}
&\sum_{i=0}^{d} f_i \begin{vmatrix} f_0 & \cdots & f_{i-1} & 0 & f_{i+1} & \cdots & f_d \\ a_{0,1} & \cdots & a_{i-1,1} & q_1 & a_{i+1,1} & \cdots & a_{d,1} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{0,d} & \cdots & a_{i-1,d} & q_d & a_{i+1,d} & \cdots & a_{d,d} \end{vmatrix} \\
&= \sum_{i=0}^{d} f_i \sum_{j=1}^{d} (-1)^{i+j} q_j \begin{vmatrix} f_0 & \cdots & f_{i-1} & f_{i+1} & \cdots & f_d \\ a_{0,1} & \cdots & a_{i-1,1} & a_{i+1,1} & \cdots & a_{d,1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{0,j-1} & \cdots & a_{i-1,j-1} & a_{i+1,j-1} & \cdots & a_{d,j-1} \\ a_{0,j+1} & \cdots & a_{i-1,j+1} & a_{i+1,j+1} & \cdots & a_{d,j+1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{0,d} & \cdots & a_{i-1,d} & a_{i+1,d} & \cdots & a_{d,d} \end{vmatrix} \\
&= \sum_{j=1}^{d} (-1)^{j} q_j \sum_{i=0}^{d} (-1)^{i} f_i \begin{vmatrix} f_0 & \cdots & f_{i-1} & f_{i+1} & \cdots & f_d \\ a_{0,1} & \cdots & a_{i-1,1} & a_{i+1,1} & \cdots & a_{d,1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{0,j-1} & \cdots & a_{i-1,j-1} & a_{i+1,j-1} & \cdots & a_{d,j-1} \\ a_{0,j+1} & \cdots & a_{i-1,j+1} & a_{i+1,j+1} & \cdots & a_{d,j+1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{0,d} & \cdots & a_{i-1,d} & a_{i+1,d} & \cdots & a_{d,d} \end{vmatrix} \qquad (1) \\
&= \sum_{j=1}^{d} (-1)^{j} q_j \begin{vmatrix} f_0 & \cdots & f_{i-1} & f_i & f_{i+1} & \cdots & f_d \\ f_0 & \cdots & f_{i-1} & f_i & f_{i+1} & \cdots & f_d \\ a_{0,1} & \cdots & a_{i-1,1} & a_{i,1} & a_{i+1,1} & \cdots & a_{d,1} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{0,j-1} & \cdots & a_{i-1,j-1} & a_{i,j-1} & a_{i+1,j-1} & \cdots & a_{d,j-1} \\ a_{0,j+1} & \cdots & a_{i-1,j+1} & a_{i,j+1} & a_{i+1,j+1} & \cdots & a_{d,j+1} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{0,d} & \cdots & a_{i-1,d} & a_{0,i} & a_{i+1,d} & \cdots & a_{d,d} \end{vmatrix} = 0,
\end{aligned}$$

as every determinant in the last sum has the same first two rows.

From the above relation, we can see that

$$\begin{aligned}
m\ \theta(f_0,\ldots,f_d) &= m \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \begin{vmatrix} \pi_0(f_0) & \pi_0(f_1) & \cdots & \pi_0(f_d) \\ \pi_1(f_0) & \pi_1(f_1) & \cdots & \pi_1(f_d) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \pi_d(f_1) & \cdots & \pi_d(f_d) \end{vmatrix} \\
&= \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \begin{vmatrix} m\pi_0(f_0) & m\pi_0(f_0) & \cdots & m\pi_0(f_d) \\ \pi_1(f_0) & \pi_1(f_1) & \cdots & \pi_1(f_d) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \pi_d(f_1) & \cdots & \pi_d(f_d) \end{vmatrix} \\
&= \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \sum_{i=0}^{d} f_i \begin{vmatrix} \pi_0(f_0) & \cdots & \pi_0(f_{i-1}) & m & \pi_0(f_{i+1}) & \cdots & \pi_0(f_d) \\ \pi_1(f_0) & \cdots & \pi_1(f_{i-1}) & 0 & \pi_1(f_{i+1}) & \cdots & \pi_1(f_d) \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \cdots & \pi_d(f_{i-1}) & 0 & \pi_d(f_{i+1}) & \cdots & \pi_d(f_d) \end{vmatrix}
\end{aligned}$$

$$= \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \sum_{i=0}^{d} f_i \left(\begin{vmatrix} \pi_0(f_0) & \dots & \pi_0(f_{i-1}) & m & \pi_0(f_{i+1}) & \dots & \pi_0(f_d) \\ \pi_1(f_0) & \dots & \pi_1(f_{i-1}) & 0 & \pi_1(f_{i+1}) & \dots & \pi_1(f_d) \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \dots & \pi_d(f_{i-1}) & 0 & \pi_d(f_{i+1}) & \dots & \pi_d(f_d) \end{vmatrix} \right.$$
$$\left. + \begin{vmatrix} \pi_0(f_0) & \dots & \pi_0(f_{i-1}) & 0 & \pi_0(f_{i+1}) & \dots & \pi_0(f_d) \\ \pi_1(f_0) & \dots & \pi_1(f_{i-1}) & q_1 & \pi_1(f_{i+1}) & \dots & \pi_1(f_d) \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \dots & \pi_d(f_{i-1}) & q_d & \pi_d(f_{i+1}) & \dots & \pi_d(f_d) \end{vmatrix} \right)$$
$$= \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \sum_{i=0}^{d} f_i \begin{vmatrix} \pi_0(f_0) & \dots & \pi_0(f_{i-1}) & m & \pi_0(f_{i+1}) & \dots & \pi_0(f_d) \\ \pi_1(f_0) & \dots & \pi_1(f_{i-1}) & q_1 & \pi_1(f_{i+1}) & \dots & \pi_1(f_d) \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \dots & \pi_d(f_{i-1}) & q_d & \pi_d(f_{i+1}) & \dots & \pi_d(f_d) \end{vmatrix},$$

where the last equality is obtained by adding in the sum of equation (1) using multilinearity of determinants. Since q_i can be anything, we choose $q_i = \pi_i(m)$, and since $m = \pi_0(m)$, we have

$$\begin{aligned} m\ \theta(f_0, \dots, f_d) &= \sum_{i=0}^{d} f_i \prod_{i=1}^{d} \frac{1}{\overline{x}_i - x_i} \begin{vmatrix} \pi_0(f_0) & \dots & \pi_0(f_{i-1}) & \pi_0(m) & \pi_0(f_{i+1}) & \dots & \pi_0(f_d) \\ \pi_1(f_0) & \dots & \pi_1(f_{i-1}) & \pi_1(m) & \pi_1(f_{i+1}) & \dots & \pi_1(f_d) \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \pi_d(f_0) & \dots & \pi_d(f_{i-1}) & \pi_d(m) & \pi_d(f_{i+1}) & \dots & \pi_d(f_d) \end{vmatrix} \\ &= \sum_{i=0}^{d} f_i\ \theta(f_0, \dots, f_{i-1}, m, f_{i+1}, \dots, f_d) = \sum_{i=0}^{d} f_i\ \theta_i(m). \end{aligned}$$

□

In the case where $m = 1$, the above identity was already used in [CK00b] as well [CM96] to show that the Dixon polynomial is in the ideal of original polynomial system. As we shall see later, using a general monomial m enables us to build smaller Dixon multiplier matrices as there is a choice in selecting the monomial m.

In bilinear form,

$$\theta_i(m) = \overline{X}_i\ \Theta_i(m)\ X_i,$$

where $\Theta_i(m)$ is the Dixon matrix of $\{ f_0, \dots, f_{i-1}, m, f_{i+1}, \dots, f_d \}$. Expressing $\theta_i(m)$ in term of $\Theta_i(m)$ matrix, we have

$$\theta_i(m)\ f_i = (\overline{X}_i\, \Theta_i(m)\, X_i)\ f_i = (\overline{X}_i \Theta_i(m))(X_i f_i).$$

Thus, we can construct a multiplier matrix by using monomial multipliers X_i for f_i.

$$M \times Y = \begin{pmatrix} X_0 f_0 \\ X_1 f_0 \\ \vdots \\ X_d f_d \end{pmatrix}.$$

Using the above notation, we can rewrite the formula for the Dixon polynomial,

$$\begin{aligned} m\ \theta(f_0, \dots, f_d) &= \overline{X}\ \Theta\ m\ X \\ &= \sum_{i=0}^{d} \theta_i(m)\ f_i = \sum_{i=0}^{d} \overline{X}_i\, \Theta_i(m)\, (X_i f_i) \\ &= \overline{Y}\, (\Theta_0(m) : \Theta_1(m) : \dots : \Theta_d(m)) \begin{pmatrix} X_0 f_0 \\ X_1 f_1 \\ \vdots \\ X_d f_d \end{pmatrix} \\ &= \overline{Y}\ (T \times M)\ Y = \overline{Y}\ \Theta'\ Y, \end{aligned}$$

where $\overline{Y} = \bigcup_{i=0}^{d} \overline{X}_i$ and $\Theta' = T \times M$. Therefore,

$$\overline{X}\ \Theta\ m\ X = \overline{Y}\ \Theta'\ Y.$$

Note that $m\,X \subseteq Y$, and $\overline{X} \subseteq \overline{Y}$; therefore, Θ and Θ' are the same matrices except for Θ' having some extra zero rows and columns.

Corollary 4.1.1 *Given a polynomial system $\mathcal{F} = \{f_0, \ldots, f_d\}$, its Dixon matrix can be factored as a product of two matrices one of which is a multiplier matrix. That is*

$$\Theta = T \times M,$$

where M is the Dixon multiplier matrix of the polynomial system $\mathcal{F}$.

4.2 Univariate case: Sylvester and Dixon Multiplier

Let

$$\begin{aligned} f_0 &= a_0 + a_1 x + a_2 x^2 + \cdots + a_{m-1} x^{m-1} + a_m x^m, \\ f_1 &= b_0 + b_1 x + b_2 x^2 + \cdots + b_{n-1} x^{n-1} + b_n x^n. \end{aligned}$$

For the polynomial system $\{f_0, f_1\} \subset \mathbb{C}[x]$, the Dixon formulation is better known as Bezoutian after Bezout who gave this construction for univariate polynomials. Here in the expression $\Theta = T \times M$, we will note that M is the well known Sylvester matrix and Θ is the Bezout Matrix. The Sylvester matrix for $\{f_0, f_1\}$ is given by

$$M = \begin{array}{c} n \left\{ \begin{array}{c} \\ \\ \\ \\ \end{array} \right. \\ m \left\{ \begin{array}{c} \\ \\ \\ \\ \end{array} \right. \end{array} \left| \begin{array}{cccccccc} a_0 & a_1 & \cdots & a_m & & & & \\ & a_0 & a_1 & \cdots & a_m & & & \\ & & & & \ddots & \ddots & & \\ & & & & & a_0 & a_1 & \cdots & a_m \\ b_0 & b_1 & \cdots & b_n & & & & \\ & b_0 & b_1 & \cdots & b_n & & & \\ & & & & \ddots & \ddots & & \\ & & & & & b_0 & b_1 & \cdots & b_n \end{array} \right|,$$

and $T = (\Theta_0 : \Theta_1)$ where $\theta_0 = \theta(1, f_1) = \overline{X}_0 \Theta_0 X_0$ and $\theta_1 = \theta(f_0, 1) = \overline{X}_1 \Theta_1 X_1$. For instance,

$$\theta(1, f_1) \quad = \quad \frac{1}{\overline{x} - x} \left| \begin{array}{cc} 1 & f_1(x) \\ 1 & f_1(\overline{x}) \end{array} \right| = \frac{f_1(x) - f_1(\overline{x})}{\overline{x} - x} = b_1 S_1 + b_2 S_2 + \cdots + b_n S_n,$$

where $S_k = \sum_{i=1}^{k} \overline{x}^{i-1} x^{k-i}$.

Note that $X_0 = \{1, x, \ldots, x^{n-1}\}$ and $X_1 = \{1, x, \ldots, x^{m-1}\}$ as in the Sylvester matrix construction. From this, it can be seen that Θ_0 matrix is $n \times n$, and hence, similarly Θ_1 will be $m \times m$, and the matrix T is of size $\max\{m, n\} \times (n + m)$. For the univariate case we can write down the matrix T as

$$T = \begin{pmatrix} & & & & & & & & a_m \\ & & & b_n & & & & a_m & a_{m-1} \\ & & b_n & b_{n-1} & & & & & \\ & & & \vdots & & & & & \vdots \\ & b_n & \cdots & b_3 & b_2 & & a_m & \cdots & a_3 & a_2 \\ b_n & b_{n-1} & \cdots & b_2 & b_1 & a_m & a_{m-1} & \cdots & a_2 & a_1 \end{pmatrix}.$$

It can be seen from the above construction that when $m > n$, $\det(\Theta)$, the determinant of the Dixon matrix, has the extra factor of a_m^{m-n}, which comes from the matrix T. This can be verified using the *Cauchy-Binet* formula for the determinant of a product of non-square matrices [AW92].

4.3 Example: Bivariate Case

Let

$$\begin{aligned} f_0 &= a_{00}+a_{01}y+a_{10}x+a_{11}xy+a_{02}y^2+a_{20}x^2, \\ f_1 &= b_{00}+b_{01}y+b_{10}x+b_{11}xy+b_{02}y^2+b_{20}x^2, \\ f_2 &= c_{00}+c_{01}y+c_{10}x+c_{11}xy+c_{02}y^2+c_{20}x^2. \end{aligned}$$

In the expression $\Theta = T \times M$, matrices T and M can be split up into 3 blocks

$$T = \left(\begin{array}{c|c|c} \Theta_0 & \Theta_1 & \Theta_2 \end{array} \right), \qquad M = \left(\begin{array}{c} X_0 f_0 \\ \hline X_1 f_1 \\ \hline X_2 f_2 \end{array} \right).$$

Since in the unmixed case, $X_0 = X_1 = X_2 = \{y^2, xy, x, y, 1\}$, the structure of M is simple to see: it has 15 rows and 14 columns. Also, the monomial structure of θ_i is the same for $i = 0, 1, 2$, and hence, matrices Θ_i have the same layout. We illustrate the matrix Θ_0; the other matrices are similar.

$$\Theta_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & |02.20| \\ 0 & 0 & 0 & 0 & |20.11| \\ 0 & 0 & |20.02| & |11.02| & |10.02| \\ 0 & 0 & |20.11| & |20.02| & |10.11|+|20.01| \\ |20.02| & |11.02| & |20.01| & |10.02|+|11.01| & |10.01| \end{pmatrix}, \quad \text{where} \quad |ij.kl| = \begin{vmatrix} b_{ij} & b_{kl} \\ c_{ij} & c_{kl} \end{vmatrix}.$$

The above relationship between the Dixon multiplier matrices and the Dixon matrices has also been studied in [Zha00] for the bivariate bidegree case. The construction discussed above is general in the sense that it works for any number of variables and any degree. For the bivariate bi-degree case, the above construction reduces to the construction given in [Zha00].

4.4 Maximal Minors

It was proved in [KSY94] that under certain conditions, any maximal minor of the Dixon matrix is a projection operator (i.e., the nontrivial multiple of the resultant). [BEM00] has derived that any maximal minor of the Dixon matrix is a projection operator of a certain variety which is the projective closure of the associated parameterized affine set. These results immediately apply to the Dixon multiplier matrix, establishing that it is a resultant matrix.

Consider the following "specialization" map ϕ such that given a polynomial $f = \sum_\alpha c_\alpha \mathbf{x}^\alpha$,

$$\phi(f) = \sum_\alpha \phi\left(c_\alpha\right) \mathbf{x}^\alpha \quad \text{and} \quad \phi(c_\alpha) \in \mathbb{C}, \quad \text{so that } f \in \mathbb{C}[\mathbf{x}].$$

Theorem 4.2 *[KS96],[BEM00] The determinant of a maximal minor of the Dixon matrix Θ of a polynomial system $\mathcal{F}$ is a projection operator, that is*

$$\det\left(\mathtt{minor}_{\max}\left(\Theta\right)\right) = e \ \mathrm{Res}_V(\mathcal{F}),$$

where $\mathrm{Res}_V(\mathcal{F})$ is the resultant of the polynomial system $\mathcal{F}$ over the associated variety V[3], that is, $\phi(\mathcal{F}) \equiv 0$ has a solution $v \in V$ if and only if $\phi(\mathrm{Res}_V) = 0$.

[3]The variety in question is not defined explicitly; one can use [KSY94] or [BEM00] to define such varieties. In general, the associated variety is some projective closure of an affine set.

In general, the choice of variety is not clear, but it usually "contains" an open subset of the affine space; hence the resultant is a necessary and sufficient condition for polynomial system to have a solution in that open subset.

The next theorem shows that the resultant is a factor in the projection operators computed from the Dixon multiplier matrix using maximal minor construction. Note that in the construction of a Dixon multiplier matrix, we must assume that for a map ϕ such that $\phi(\mathcal{F}) \equiv 0$ has a solution in the respective variety, the monomial m used to construct the Dixon multiplier matrix does not vanish for that solution. If the variety being considered is the projective closure of $(\mathbb{C}^*)^d$,[4] any monomial m chosen for the construction satisfies this condition.

Theorem 4.3 *Given the Dixon matrix Θ of a polynomial system $\mathcal{F} = \{f_0, \ldots, f_d\}$, if*

$$Q = \det\left(\mathtt{minor}_{\max}(\Theta)\right), \qquad \textit{where} \qquad Q = e_Q \operatorname{Res}_V,$$

then for the Dixon multiplier matrix M of $\mathcal{F}$,

$$L = \det\left(\mathtt{minor}_{\max}(M)\right), \qquad \textit{where} \qquad L = e_L \operatorname{Res}_V.$$

Proof: By Corollary 4.1.1, there exists a matrix T such that

$$\Theta = T \times M.$$

Let $k = \operatorname{rank}(\Theta)$ and let $C_1, \ldots, C_k$ be linearly independent columns of Θ. Consequently, the corresponding columns $L_{i_1}, \ldots, L_{i_k}$ of M, such that $C_j = T \times L_{i_j}$, are also linearly independent.

Let ϕ be a specialization of coefficients of the given polynomial system such that $\operatorname{rank}(\phi(\Theta))$ is deficient, then $\phi(Q) = 0$. Note that $\phi(\mathcal{F})$ has a solution in V if and only if $\phi(\operatorname{Res}_V) = 0$.

Since $\operatorname{rank}(\phi(\Theta)) < k$, it follows that $\phi(C_1), \ldots, \phi(C_k)$ are linearly dependent, which implies that

$$\operatorname{rank}(\phi(T)) < \operatorname{rank}(T) \qquad \text{or} \qquad \phi(L_{i_1}), \ldots, \phi(L_{i_k}) \quad \text{are linearly dependent.}$$

Since the above is true for any $k \times k$ maximal minor of Θ, it follows that either (i) $\phi(T)$ has deficient rank or (ii) every corresponding $k \times k$ minor of $\phi(M)$ has rank smaller than k implying that $\operatorname{rank}(\phi(M)) < \operatorname{rank}(M)$. Therefore,

$$\operatorname{rank}(\phi(\Theta)) < \operatorname{rank}(\Theta) \qquad \Longrightarrow \qquad \operatorname{rank}(\phi(T)) < \operatorname{rank}(T) \quad \text{or} \quad \operatorname{rank}(\phi(M)) < \operatorname{rank}(M).$$

If $G_\Theta = \gcd(\det\left(\mathtt{minor}_{\max}(\Theta)\right))$, $G_T = \gcd(\det\left(\mathtt{minor}_{\max}(T)\right))$ and $G_M = \gcd(\det\left(\mathtt{minor}_{\max}(M)\right))$, then

$$\operatorname{Res}_V \mid G_\Theta \qquad \text{and} \qquad G_\Theta \mid G_T\, G_M \qquad \Longrightarrow \qquad \operatorname{Res}_V \mid G_T\, G_M.$$

To see that Res_V does not divide G_T, note that generically, for an n-degree unmixed polynomial systems ([KS96, Sax97]),

$$\deg_{f_i} G_T < \deg_{f_i} G_\Theta = \deg_{f_i} \operatorname{Res}_V,$$

where $\deg_{f_i}$ stands for the degree (in terms of generic coefficients of polynomial f_i) of the original polynomial system. Therefore, it must be that case that Res_V divides G_M. $\square$

Another advantage of the Dixon multiplier matrix is that the choice of a monomial m used to construct it influences not only the variety in question, and hence the particular resultant being computed, but also the size of the matrix itself. In practical cases, one might choose m so as to generate the smallest possible matrix and still have the condition for the existence of a solution in the variety under consideration.

This suggests that the projection operator extracted from the corresponding Dixon matrix typically contains more extraneous "information". The variety over which the resultant is included in such a projection operator can be bigger. Further, the gcd of all maximal minors of T appears as a factor in the projection operator of the Dixon matrix.

[4] $\mathbb{C}^*$ is $\mathbb{C} - \{0\}$.

4.4.1 Example

Consider a mixed bivariate system,

$$\begin{aligned} f_0 &= a_{10}x + a_{20}x^2 + a_{12}xy^2, \\ f_1 &= b_{01}y + b_{02}y^2 + b_{21}x^2y, \\ f_2 &= u_1x + u_2y + u_0. \end{aligned}$$

The mixed volume of the above polynomial system is $\langle 2, 2, 4\rangle = 8$, (see section 2.1). Hence, generically the degree of the toric resultant (capturing toric solutions $(\mathbb{C}^*)^d$) is 8.

Moreover, there are other parts to the resultant over the affine space. When $x = 0$, f_0 becomes identically 0, and

$$\begin{aligned} f_1 &= b_{01}y + b_{02}y^2, \\ f_2 &= u_2y + u_0. \end{aligned}$$

The resultant in that case is $\mathrm{Res}_{x=0} = u_0b_{02} - u_2b_{01}$.

Similarly, when $y = 0$, f_1 becomes identically 0, and

$$\begin{aligned} f_0 &= a_{10}x + a_{20}x^2, \\ f_2 &= u_1x + u_0. \end{aligned}$$

The resultant in this case is $\mathrm{Res}_{y=0} = u_0a_{20} - u_1a_{10}$. Therefore, the affine bounds are $\langle 3, 3, 7\rangle = 13$. The degree of the resultant is $\mathrm{Res}_{\substack{x=0 \\ y=0}} \text{ or } = \langle 1, 1, 3\rangle$. This is added to the degree of the toric resultant, which is:

$$\begin{aligned} \mathrm{Res}_T &= u_2^4b_{01}a_{20}^2b_{21} + u_2^4b_{21}^2a_{10}^2 - u_2^3b_{21}b_{02}a_{20}^2u_0 + u_2^3b_{21}b_{02}a_{20}u_1a_{10} - 2u_2^2b_{01}b_{21}u_1^2a_{12}a_{10} \\ &+ 4u_2^2b_{01}b_{21}u_1a_{20}a_{12}u_0 + 2u_2^2b_{21}^2u_0^2a_{12}a_{10} + u_2b_{02}u_1^3a_{12}a_{20}b_{01} + 4u_2b_{21}u_1^2u_0a_{10}b_{02}a_{12} \\ &- 3u_2b_{21}u_1a_{20}b_{02}a_{12}u_0^2 + u_1^4a_{12}^2b_{01}^2 + 2b_{21}u_0^2a_{12}^2u_1^2b_{01} + b_{02}^2a_{12}u_1^4a_{10} + b_{21}^2u_0^4a_{12}^2 \\ &- b_{02}^2a_{12}u_1^3a_{20}u_0. \end{aligned}$$

The resultant over the affine space is

$$\mathrm{Res}_A = u_0 \ \mathrm{Res}_{x=0} \ \mathrm{Res}_{y=0} \ \mathrm{Res}_T.$$

In contrast, the projection operator obtained from the 7×7 Dixon matrix is

$$\det(\Theta) = b_{02}\, a_{20}\, b_{21}^3\, a_{12}^3\, u_0 \ \mathrm{Res}_{x=0} \ \mathrm{Res}_{y=0} \ \mathrm{Res}_T.$$

If a Dixon multiplier matrix is constructed using the term $m = 1$ in the construction, then

$$\det(M) = b_{21}\, u_0 \ \mathrm{Res}_{x=0} \ \mathrm{Res}_{y=0} \ \mathrm{Res}_T.$$

But if y, for instance, is used to construct the Dixon multiplier matrix is chosen, then

$$\det(M) = b_{21} \, \mathrm{Res}_{x=0} \ \mathrm{Res}_T.$$

It appears that in a generic mixed case, extraneous factors can be minimized by using the Dixon multiplier matrix.

For toric solutions in $(\mathbb{C}^*)^d$, it suffices to choose any m for constructing the Dixon multiplier matrix. Selection of m can be formulated as an optimization problem as discussed below so that the Dixon multiplier matrix is of the smallest size. This way, the degree of a projection operator and hence the degree of the extraneous factor in it can be minimized.

5 Minimizing the Degree of Extraneous Factors

For computing a resultant over a toric variety, the supports of a given polynomial system can be translated so as to construct smaller Dixon as well as Dixon multiplier matrices. This is evident from the following example for the bivariate case.

Example: Consider the following polynomial system:

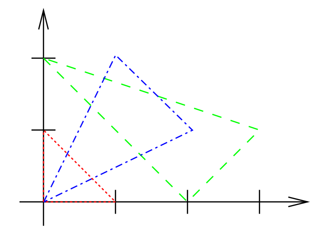

Figure 1: Mixed example.

$$\begin{aligned} f_0 &= a_{00} + a_{10}x + a_{01}y, \\ f_1 &= b_{02}y^2 + b_{20}x^2 + b_{31}x^3y, \\ f_2 &= c_{00} + c_{12}xy^2 + c_{21}x^2y. \end{aligned}$$

This generic polynomial system has the 2-fold mixed volume of $\langle 8, 3, 4\rangle = 15$; hence, the optimal multiplier matrix is 15×15, containing 8 rows from polynomial f_0, 3 rows from f_1 and 4 rows from f_2. Figure 1 shows the overlaid supports of these polynomials.

To construct the Dixon multiplier matrix, if we choose $m = x^{\alpha_x}y^{\alpha_y} = 1$, i.e. $\alpha = (0, 0)$ and consider the original polynomial system $\{f_0, f_1, f_2\}$, then

$$|X_0| = 9, \quad |X_1| = 4, \quad |X_2| = 5,$$

and the Dixon multiplier matrix has 18 rows. In fact, for the polynomial system $\{f_0, f_1, f_2\}$, the best choice for α is from $\{(0, 0), (0, 1), (1, 0)\}$, each one producing a 18×18 Dixon multiplier matrix. In other words, an extraneous factor of at least degree 3 is generated using the Dixon multiplier matrix no matter what multiplier monomial is used if supports are not translated.

On the other hand, if we consider $\{x^2yf_0, f_1, xf_2\}$ and let $\alpha \in \{(2, 1), (2, 2), (3, 1)\}$,

$$|X_0| = 8, \quad |X_1| = 3, \quad |X_2| = 4,$$

and the resulting Dixon multiplier matrix has 15 rows, i.e., the matrix is optimal. Figure 2 shows the translated supports. This example also illustrates that it is possible to get exact resultant matrices if supports are translated even when untranslated supports have a nonempty intersection.

The Dixon matrix for the above polynomial system is of size 9×9; its size is the same as though the system was unmixed with the support of the polynomial system being the union of individual supports of the polynomials. For the translated polynomial system, however, the Dixon matrix is of size 8×8. In both cases, there are extraneous factors of degree 12 and 9, respectively.

In fact, it will be shown (section 5.1.1) that for generic mixed bivariate polynomial systems, the size of the Dixon matrix is at least $\max(|X_0|, |X_1|, |X_2|)$ when a monomial is appropriately chosen to do its construction.

As illustrated by the above example, the Dixon multiplier matrix as well as the Dixon matrix are sensitive to the translation of the supports of the polynomials in the polynomial system. Since the mixed volume of supports is invariant under translation, most resultant methods in which matrices are constructed using supports are also invariant with respect to translation of supports. Methods based on mixed volume compute resultants over toric variety and hence are insensitive to such translations of the supports of polynomials.

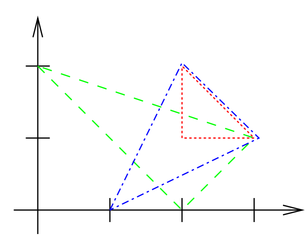

Figure 2: Translated example.

Since the Dixon multiplier matrix is sensitive to the choice of α (whereas the Dixon matrix is not), it is possible to further optimize the size of the Dixon multiplier matrix by properly selecting the multiplier monomial, along with an appropriate translation of the supports of the polynomial system.

To formalize the above discussion, let $\alpha \in \mathbb{N}^d$; consider a translation $t = \langle t_0, t_1, t_2, \ldots, t_d\rangle$ where $t_i \in \mathbb{N}^d$, to translate the support of a given polynomial system. The resulting translated support is $\mathcal{A} + t = \langle \mathcal{A}_0 + t_0, \mathcal{A}_1 + t_1, \cdots, \mathcal{A}_d + t_d\rangle$.

Choosing an appropriate m and t for mixed polynomial systems can be formulated as an optimization problem in which the size of the support of each $\theta_i(m)$ and hence, the size of the multiplier set for each f_i is minimized.

One can either optimize the size of the Dixon matrix or alternatively, the size of the Dixon multiplier matrix can be optimized. In other words, we can consider the following two problems:

(i) for the Dixon matrix construction, the size of the Dixon polynomial $|\Delta_{\mathcal{A}+t}|$ is minimized.

(ii) for the Dixon multiplier matrix, where $\Phi_i(\alpha, t) = |\Delta_{\langle \mathcal{A}_0+t_0,\ldots,\mathcal{A}_{i-1}+t_{i-1},\{\alpha\},\mathcal{A}_{i+1}+t_{i+1},\ldots,\mathcal{A}_d+t_d\rangle}|$, the sum

$$\Phi(\alpha, t) = \sum_{i=0}^{d} \Phi_i(\alpha, t),$$

i.e., the number of rows is minimized

In the case of (i), a heuristic for choosing such translation vectors t is described in [CK02a]. For minimizing the size of Dixon multiplier matrices, a slightly modified method is needed as the objective is to minimize the sizes of $\theta_i(m)$, i.e., $\Phi_i(\alpha, t)$ for each i (which also involves choosing m). Since $\Phi_i(\alpha, t)$ represents the number of rows corresponding to the polynomial $\mathbf{x}^{t_i} f_i$ in the Dixon multiplier matrix, the goal is to find α and $t = \langle t_0, t_1, \ldots, t_d\rangle$ such that $\Phi(\alpha, t)$ is minimized, that is, the size of the entire Dixon multiplier matrix is minimized so as to minimize the degree of the extraneous factor.

Below, we make some observations and prove properties which are helpful in selecting α and t.

5.1 Multiplier Sets using the Dixon Method

The multipliers used in the construction of a Dixon multiplier matrix are related to the monomials of the Dixon polynomial, which also label the columns of the Dixon matrix. For a given polynomial f_i, its multiplier set generated using a term α is obtained from $\theta_i(f_0, \cdots, f_{i-1}, \alpha, f_{i+1}, \cdots, f_d)$. The support of the polynomial system for which $\theta_i(f_0, \cdots, f_{i-1}, \alpha, f_{i+1}, \cdots, f_d)$ is the Dixon polynomial, is $\langle \mathcal{A}_0, \ldots, \mathcal{A}_{i-1}, \{\alpha\}, \mathcal{A}_{i+1}, \ldots, \mathcal{A}_d\rangle$. This support is denoted by $\mathcal{A}(i, \alpha)$.

Proposition 5.1 *Given a support $\mathcal{A} = \langle \mathcal{A}_0, \ldots, \mathcal{A}_d\rangle$ of a polynomial system $\mathcal{F}$, for any $\alpha \in \mathbb{N}^d$,*

$$\Delta_{\mathcal{A}} \subseteq \bigcup_{i=0}^{d} \Delta_{\mathcal{A}(i,\alpha)}.$$

Proof: Note that $\Delta_{\mathcal{A}(i,\alpha)}$ is the support of $\theta_i(\mathbf{x}^\alpha)$. Since $\mathbf{x}^\alpha\, \theta(f_0, \ldots, f_d) = \sum_{i=0}^{d} f_i\, \theta_i(\mathbf{x}^\alpha)$; we can conclude that

$$\overline{\Delta}_{\mathcal{A}} \subseteq \bigcup_{i=0}^{d} \overline{\Delta}_{\mathcal{A}(i,\alpha)},$$

since no f_i has terms in variables $\overline{x}_i$.

As stated earlier, the Dixon polynomial depends on the variable order used in its construction. Let $\Delta_{\mathcal{A}}^{\langle x_1,x_2,\ldots,x_d\rangle}$ stand for the support of the Dixon polynomial constructed using the variable order $\langle x_1, x_2, \ldots, x_d\rangle$, i.e., x_1 is first replaced by $\overline{x}_1$, followed by x_2 and so on. Therefore,

$$\overline{\Delta}_{\mathcal{A}}^{\langle x_d,\ldots,x_1\rangle} \subseteq \bigcup_{i=0}^{d} \overline{\Delta}_{\mathcal{A}(i,\alpha)}^{\langle x_d,\ldots,x_1\rangle}.$$

However, $\overline{\Delta}_{\mathcal{A}}^{\langle x_d,\ldots,x_1\rangle} = \Delta_{\mathcal{A}}^{\langle x_1,\ldots,x_d\rangle}$, as can be seen from Definition 3.1 of θ. Substituting this into the previous equation, the statement of the proposition is proved. □

The above proposition establishes that the support of the Dixon polynomial is contained in the union of the Dixon multiplier sets. It is shown below that the converse holds if the term chosen for the construction of the Dixon multipliers appears in all the polynomials of a given polynomial set.

Theorem 5.1 *Given $\mathcal{A} = \langle \mathcal{A}_0, \ldots, \mathcal{A}_d\rangle$ as defined above, consider an $\alpha \in \bigcap_{i=0}^{d} \mathcal{A}_i$. Then,*

$$\Delta_{\mathcal{A}} = \bigcup_{i=0}^{d} \Delta_{\mathcal{A}(i,\alpha)}.$$

Proof: Using Proposition 5.1, it suffices to show that

$$\bigcup_{i=0}^{d} \Delta_{\mathcal{A}(i,\alpha)} \subseteq \Delta_{\mathcal{A}}.$$

Note that

$$\Delta_{\mathcal{A}} = \bigcup_{\sigma \Subset \mathcal{A}} \Delta_\sigma \quad \text{and} \quad \Delta_{\mathcal{A}(i,\alpha)} = \bigcup_{\sigma \Subset \mathcal{A}(i,\alpha)} \Delta_\sigma,$$

and for every $\sigma \Subset \mathcal{A}(i,\alpha)$, $\sigma \Subset \mathcal{A}$. Hence the statement of the theorem follows. □

In particular, for an unmixed $\mathcal{A}$, where $\mathcal{A}_i = \mathcal{A}_j$ for all $i, j \in \{0, \ldots, d\}$,

$$\Delta_{\mathcal{A}} = \Delta_{\mathcal{A}(i,\alpha)} \qquad \text{for any } \alpha \in \mathcal{A}_i, \text{ and } i \in \{0, \ldots, d\}.$$

The above property shows that the columns of the Dixon matrix are exactly the monomial multipliers of the Dixon multiplier matrix. That is

$$X \subseteq \bigcup_{i=0}^{d} X_i$$

in the construction of a Dixon multiplier matrix, and equality holds whenever the chosen monomial m is in the support of all polynomials of an unmixed polynomial system. The above property is independent of choice of the $m = \mathbf{x}^\alpha$ in $\mathcal{A}$, indicating a tight relationship between the Dixon matrix and the associated Dixon multiplier matrix.

5.1.1 Size of the Dixon Multiplier Matrix

Using Theorem 5.1, we can prove an observation made earlier that the size of a Dixon matrix is at least as big as the size of the largest multiplier set of the corresponding Dixon multiplier matrices.

Theorem 5.2 *For a generic polynomial system $\mathcal{F}$, the size of the Dixon matrix is at least as big as the size of the largest multiplier set for the Dixon multiplier matrices, i.e.,*

$$\texttt{Size}(\Theta) \geq \max_{i=0}^{d} \Phi_i(\alpha) \qquad \textit{when} \quad \alpha \in \bigcap_{i=0}^{d} \mathcal{A}_i.$$

A direct consequence of the above theorem is:

Corollary 5.2.1 *Consider a polynomial system with support $\mathcal{A} = \langle \mathcal{A}_0, \ldots, \mathcal{A}_d \rangle$ and let $\alpha \in \cap_{i=0}^{d} \mathcal{A}_i$. Then,*

$$(d+1)\,\texttt{Size}(\Theta) \geq \texttt{Size}(M_\alpha),$$

where M_α is the Dixon multiplier matrix constructed using α, and the above relation becomes an equality in the unmixed case, that is when $\mathcal{A}_i = \mathcal{A}_j$ for all $1 \leq i \neq j \leq d$.

Proof: The number of multipliers for a polynomial f_i used in the construction of a Dixon multiplier matrix is $\Phi_i(\alpha)$, and in unmixed case $\Phi_i(\alpha) = \Phi_j(\alpha)$, for all i, j. The number of rows of M_α is the sum of sizes of the multiplier sets for each polynomial, i.e. $\Phi(\alpha) = (d+1)\Phi_i(\alpha)$ and also

$$\Delta_{\mathcal{A}} = \bigcup_{i=0}^{d} \Delta_{\mathcal{A}(i,\alpha)} \qquad \text{and therefore} \qquad |\Delta_{\mathcal{A}(i,\alpha)}| = \Phi_i(\alpha) \leq |\Delta_{\mathcal{A}}| = \texttt{Size}(\Theta).$$

By Theorem 5.1. It follows that M_α is at most $d+1$ times bigger than Θ, where equality happens in the unmixed case. □

Using the above propositions, we have one of the key results of this paper.

Theorem 5.3 *Given a generic, unmixed polynomial system $\mathcal{F}$ and a monomial m in $\mathcal{F}$, then if the Dixon matrix is exact, the Dixon multiplier matrix built using monomial m is also exact.*

Proof: Since the Dixon matrix is exact, its size $\texttt{Size}(\Theta)$ equals to the degree the coefficients of f_i appearing in the resultant, which by Theorem 5.1, is the size of multiplier set for each polynomial f_i in the polynomial system $\mathcal{F}$. Hence the coefficients of f_i will appear at most of the same degree $\texttt{Size}(\Theta)$, in the projection operator of the Dixon multiplier matrix. But this is precisely their degree in the resultant, therefore the projection operator extracted from the Dixon multiplier matrix is precisely the resultant, that is, M_α is exact. □

In all generic cases, the ratio between the sizes of two matrices is at most $d+1$; therefore, the Dixon multiplier matrices are as good as Dixon matrices in unmixed cases, in terms of extraneous factors and usually better in mixed cases as well.

5.2 Choosing Monomial for Constructing Multiplier Sets

From the last two subsections, we can make the following observations to design a heuristic to choose α and t to minimize $\Phi(\alpha, t)$.

1. Let $\mathcal{Q}_t = \bigcup_{j=0}^{d} (t_j + \mathcal{A}_j)$; consider the support $\mathcal{P}_t = \langle \mathcal{Q}_t, \ldots, \mathcal{Q}_t \rangle$ of an unmixed polynomial system. Then, by Theorem 5.1,

$$\bigcup_{i=0}^{d} \Delta_{[\mathcal{A}+t](i,\alpha)} \subseteq \Delta_{\mathcal{P}_t} \qquad \text{and therefore for all } i, \qquad \Phi_i(\alpha, t) \, \leq \, |\Delta_{\mathcal{P}_t}|.$$

   Since it is difficult to minimize the size of $\Delta_{[\mathcal{A}+t](i,\alpha)}$, which is $\Phi_i(\alpha, t)$, we will minimize $|\Delta_{\mathcal{P}_t}|$, that is, we will use a method to choose t for minimizing the size of the Dixon matrix.

   - Let $\mathcal{Q}_t^i = \bigcup_{j=0}^{i} (t_j + \mathcal{A}_j)$, and let $\mathcal{P}_t^i = \langle \mathcal{Q}_t^i, \ldots, \mathcal{Q}_t^i \rangle$; find t_{i+1} such that the $\Delta_{\mathcal{P}_t^{i+1}}$ is minimal.
   - The procedure for finding t_{i+1} is iterative and like a gradient ascent (hill climbing) method. Starting with $i = 0$, set the initial guess to be $t'_{i+1} = (0, \ldots, 0)$, and compute the size of $\Delta_{\mathcal{P}_t^{i+1}}$. Then, select a neighboring point of t'_{i+1} and compute the size of the resulting set $\Delta_{\mathcal{P}_{t'}^{i+1}}$. If the set is smaller, select the neighboring point. Stop when all neighboring points result in support hulls of bigger size.

2. Once t is fixed by using the above procedure to minimize $|\Delta_{\mathcal{P}_t}|$, search for a monomial with exponent α for constructing the Dixon multiplier matrix such that

$$\alpha \in \texttt{SupportHull} \bigcup_{\substack{j=0 \\ j \neq i}}^{d} (t_j + \mathcal{A}_j), \quad \text{for maximal number of indices} \quad i = 0, \ldots, d,$$

where `SupportHull` is defined below.

Definition 5.1 *Given $k \in \mathbb{Z}_2^d$ and points $p, q \in \mathbb{N}^d$ define*

$$p \underset{k}{\leq} q \quad \textit{if} \quad \begin{cases} p_j \leq q_j & \textit{if } k_j = 1, \\ p_j \geq q_j & \textit{if } k_j = 0. \end{cases}$$

The support hull can be defined to be the set of points which are "inside" the hull.

Definition 5.2 *Given a support $\mathcal{P} \subset \mathbb{N}^d$, definite its support hull to be*

$$\texttt{SupportHull}(\mathcal{P}) = \{ p \mid \forall\, k \in \mathbb{Z}_2^d, \quad \exists\, q \in \mathcal{P}, \quad \textit{such that} \quad p \underset{k}{\leq} q \}.$$

The support hull of a given support is similar to the associated convex hull; whereas the later is defined in terms of the shortest Euclidean distance, the support hull is defined using the Manhattan distance. (For a complete description, see [CK02a].)
Example: Consider a highly mixed polynomial system (see figure 3):

$$\begin{aligned} f_0 &= a_{20}x^2 + a_{40}x^4 + a_{56}x^5y^6 + a_{96}x^9y^6, \\ f_1 &= b_{30}x^3 + b_{27}x^2y^7 + b_{59}x^5y^9 + b_{69}x^6y^9, \\ f_2 &= c_{04}y^4 + c_{29}x^2y^9 + c_{88}x^8y^8. \end{aligned}$$

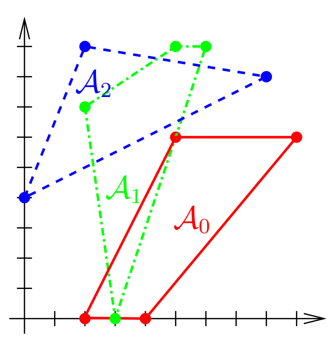

Figure 3: Mixed example.

The above procedure is illustrated in some detail on this example. The objective is to translate the supports so that they overlap the most. $\mathcal{A}_1$ is fixed; we try to adjust $\mathcal{A}_0$. At the start-point ($t_0 = (0, 0)$), $\Delta_{\mathcal{P}_t^1}$ is obtained which results in an unmixed system whose Dixon matrix is of size 80 columns. Using that as the starting point, we do local search in all four directions, and pick the one with the least matrix size: 74. We repeat the procedure as illustrated below, finally leading to the case when $t_0 = (-1, 3)$ and the Dixon matrix size is 65. At this point, all four directions lead to matrices of larger size, which is the stopping criterion.

$$\begin{array}{ccccccc}
 & & 72 & & & & \\
 & & \uparrow & & & & \\
67 & \leftarrow & \boxed{65} & \rightarrow & 71 & & \\
 & & \Uparrow & & & & \\
69 & \leftarrow & 68 & \rightarrow & 74 & & \\
 & & \Uparrow & & & & \\
71 & \leftarrow & 71 & & 77 & & \\
 & & \Uparrow & & \uparrow & & \\
74 & \leftarrow & 74 & \Leftarrow & \boxed{80} & \rightarrow & 88 \\
 & & \downarrow & & \downarrow & & \\
 & & 84 & & 90 & &
\end{array}$$

Now that t_0 and t_1 are fixed (in the above case they are $(-1,3)$ and $(0,0)$), t_2 is found in the same way. Starting with the initial value for which the Dixon matrix is of size 97, we eventually get the Dixon matrix of size 82 at $t_2 = (2,0)$ as shown below. Therefore we get $t_2 = (2,0)$. Figure 5 shows the final arrangement of the supports.

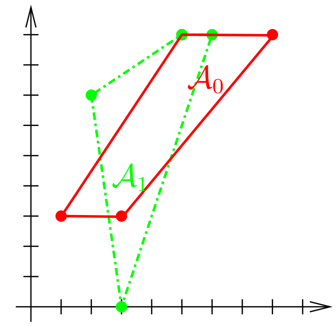

Figure 4: Adjusted $\mathcal{A}_0$.

$$\begin{array}{ccccccccc}
 & & 102 & & 88 & & 86 & & \\
 & & \uparrow & & \uparrow & & \uparrow & & \\
110 & \leftarrow & \boxed{97} & \Rightarrow & 84 & \Rightarrow & \boxed{82} & \rightarrow & 88 \\
 & & \downarrow & & \downarrow & & \downarrow & & \\
 & & 98 & & 86 & & 85 & &
\end{array}$$

For a proper choice of α (see figure 6, where $\Phi(\alpha,t) = \sum_{i=0}^{d} \Phi_i(\alpha,t)$ is shown), $\Phi_i(\alpha,t) \leq 82$. We should note that the optimal choice (obtained by exhaustive search) for the Dixon multiplier matrix is $t = \langle(-1,3),(0,0),(3,0)\rangle$. In the above example, originally the Dixon matrix (without translating supports) has 109 columns; after translation, it has only 82 columns. The Dixon multiplier matrix for the original supports is $91+95+89 = 275$ rows; after the above translation, the number of rows is $77+53+65 = 195$. The optimal translation leads to a Dixon multiplier matrix with $75+53+65 = 193$ rows.

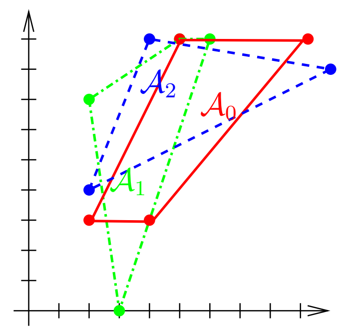

Figure 5: Adjusted $\mathcal{A}_2$.

Note that BKK bound of the above system is $\langle 75,51,63\rangle$, putting the lower bound of 189 on the size of any dialytic resultant matrix. By trying to optimize the size of the matrix, the degree (and as a consequence the size) of the extraneous factors has been brought down. In this example, it is $\langle 2,2,2\rangle$, where i^{th} entry in the tuple denotes the degree of extraneous factor in terms of coefficients of f_i. Note that in this example, α is chosen from the support hull intersection; in general, good choices for α are always from the support hull as the Dixon matrix construction is invariant under the presence of monomials whose exponent is support hull interior (see [CK02a]).

In general, to find a translation vector $t = \langle t_0,\ldots,t_d\rangle$ using the above procedure, one possibility is to search for each t_i in the range so that there is some overlap between $\mathcal{Q}_t^{i-1}$ and $t_i + \mathcal{A}_i$. If the maximum degree of the polynomial system is k, then the distance from optimal t_i and the initial guess will be in the order k. Hence the cost of finding translation vector t is

$$O(d\,k)\,\texttt{Cost}(|\Delta_{\mathcal{P}_t^i}|).$$

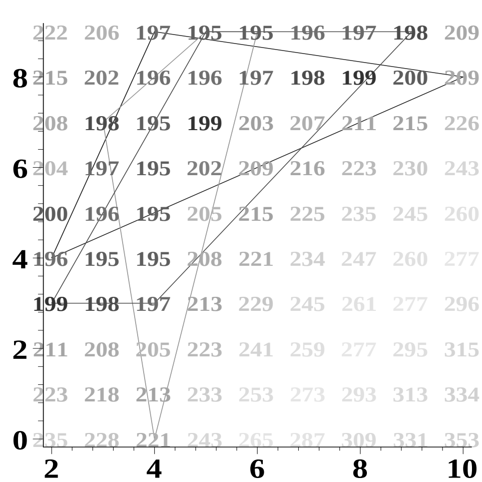

Figure 6: $\Phi(\alpha,t)$ for different choices of $\alpha \in \mathbb{Z}^2$ and fixed t.

In the next section we will derive the complexity of constructing a Dixon matrix, a Dixon multiplier matrix as well as $\texttt{Cost}(|\Delta_{\mathcal{P}_t^i}|)$.

6 Complexity & Empirical Results

Proposition 6.1 *Given a polynomial system $\mathcal{F} = \{f_0, \ldots, f_d\}$ with support $\langle \mathcal{A}_0, \ldots, \mathcal{A}_d \rangle$, let $n = |\bigcup_{i=0}^{d} \mathcal{A}_i|$, the complexity of constructing the Dixon multiplier matrix M_α for fixed $\alpha \in \mathbb{N}^d$ is*

$$T_{DM} = O\left(\frac{n!\, d^2(d+1)}{(n-d)!}\right) = O(d^3 n^d).$$

Proof: Assume (in the worst case) that each support has n points; therefore, each θ_i has the same structure, except for different permutation of columns. Hence, the total complexity is bounded from above by $(d+1)$ times the complexity of expanding single θ_i, w.l.o.g. assume θ_0.

$$\theta_0 = \theta(\mathbf{x}^\alpha, f_1, \ldots, f_d) = \prod_{i=0}^{d} \frac{1}{\overline{x}_i - x_i}$$

$$\sum_{\sigma \in \langle \{\alpha\}, \mathcal{A}_1, \ldots, \mathcal{A}_d \rangle} \begin{vmatrix} 1 & 0 & 0 & \cdots & 0 \\ c_{1,\sigma_0} & c_{1,\sigma_1} & c_{1,\sigma_2} & \cdots & c_{1,\sigma_d} \\ c_{2,\sigma_0} & c_{2,\sigma_1} & c_{2,\sigma_2} & \cdots & c_{2,\sigma_d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{d,\sigma_0} & c_{d,\sigma_1} & c_{d,\sigma_2} & \cdots & c_{d,\sigma_d} \end{vmatrix} \times \begin{vmatrix} \pi_0(\mathbf{x}^\alpha) & \pi_0(\mathbf{x})^{\sigma_1} & \pi_0(\mathbf{x})^{\sigma_2} & \cdots & \pi_0(\mathbf{x})^{\sigma_d} \\ \pi_1(\mathbf{x}^\alpha) & \pi_1(\mathbf{x})^{\sigma_1} & \pi_1(\mathbf{x})^{\sigma_2} & \cdots & \pi_1(\mathbf{x})^{\sigma_d} \\ \pi_2(\mathbf{x}^\alpha) & \pi_2(\mathbf{x})^{\sigma_1} & \pi_2(\mathbf{x})^{\sigma_2} & \cdots & \pi_2(\mathbf{x})^{\sigma_d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \pi_d(\mathbf{x}^\alpha) & \pi_d(\mathbf{x})^{\sigma_1} & \pi_d(\mathbf{x})^{\sigma_2} & \cdots & \pi_d(\mathbf{x})^{\sigma_d} \end{vmatrix},$$

where $\sigma_i \in \mathcal{A}_i$, for $i \in \{1, \ldots, d\}$ and $\sigma_i \neq \sigma_j$ for $i \neq j$. Hence there are $\binom{n}{d}$ terms in this sum, where we need to expand 2 determinants of size $(d+1)$. In the worst case, it will take $2(d+1)!$ time. It is not necessary to carry out division by $\overline{x}_i - x_i$, as the resulting monomials can be easily deduced. This is analogous to the Sylvester matrix construction, where given the degrees of the polynomials, any entry in the matrix can be deduced in constant time; in the second determinant, operations have to be done on exponent vectors, and hence, have the complexity of d. □

If we take into account the search for optimizing translation vector t, and constructing monomial with exponent α, then total complexity is

$$O(d^4 k n^d),$$

where k is the range of search for the translation vector t, which is bounded by maximum degree of all polynomials.

Note that the subdivision and incremental methods have, respectively, complexities as

$$T_S = O\left(Size(M) d^{9.5} n^{6.5} \log^2 k \log^2 \frac{1}{\epsilon_l \epsilon_\delta}\right) \quad \text{and} \quad T_I = O^*\left(e^{3d} n^{5.5} (\deg \mathrm{Res})^3\right) + O^*\left(d^{7.5} n^{2d+5.5}\right),$$

where k is the maximum degree of polynomials in the system, ϵ_l is the probability of failure to pick generic lifting vector, and ϵ_δ is the probability of perturbation failure [CE00]. In the above formula, $Size(M) = \Omega(\deg \mathrm{Res})$ and $\deg \mathrm{Res} = \Omega(n^d)$. With these crude lower bounds for the size of the resultant matrix and the degree of the resultant itself, we can compare the methods.

$$\frac{T_S}{T_{DM}} = O\left(d^{5.5} n^{6.5} \log \frac{1}{\epsilon_l \epsilon_\delta}\right) \quad \text{and} \quad \frac{T_I}{T_{DM}} = O\left(e^{3d} n^{5.5} \deg \mathrm{Res}^2\right) + O\left(d^{3.5} n^{d+5.5}\right).$$

The experimental results below confirm that the Dixon based dialytic method is an order of magnitude faster than the subdivision and incremental algorithms as well as more successful in constructing smaller matrices (the main bottleneck); further more, it can be optimized to construct even smaller matrices if one is willing to spend more time in searching for the appropriate α and t.

Column *Dixon Mult* shows the size and construction time when $t = \langle \mathbf{0}, \ldots, \mathbf{0} \rangle$ and $\alpha = \mathbf{0}$. Column *DM optimized* shows results when t and α are searched using the above heuristic. Entry containing *unmix* is not filled since the example is unmixed for which no optimization is needed.

The implementation of the *incremental method* [EC95] is in C, where as other algorithms are implemented in Maple. The implementation of the *subdivision* algorithm used is downloaded from I. Emiris's web site. The optimization algorithm is implemented very crudely, without taking advantage of appropriate data structures. All operations are done on lists; hence, timings deteriorate fast with the number of variables.

Nr.	Problem	Incremental		Subdivision		Dixon Mult		DM optimized	
		Size	Time	Size	Time	Size	Time	Size	Time
1.	Pappus theorem (6d)	-	-	20	7.00	17	0.44	-	-
2.	Max Volume of tetrahedron (4d)	107	5.75	137	186.83	60	0.36	60	1508.21
3.	Side Bisector (2d)	33	0.08	33	7.58	40	0.09	30	3.90
4.	Conformal anal. Cyclic molecules (3d)	112	4.14	119	140.96	84	3.39	unmix	-
5.	Kissing Circles Theorem (5d)	-	-	239	624.62	208	2.39	-	-
6.	Implicitization of strophoid (2d)	12	0.12	11	3.56	10	0.22	13	13.85
7.	Random unmixed, (2d)-21 Unmixed	395	63.77	384	881.26	308	5.76	unmix	-
8.	Random mixed, (3d) 4 simlexes	461	46.50	480	1950.58	494	21.08	452	251.88
9.	Random mixed, (2d)-mixed	301	19.06	296	371.70	306	4.45	261	46.44
10.	Random mixed, (4d)-simplexes ≤ 3	745	188.52	663	4738.70	579	43.36	419	46301.62

Table 1: Performance comparison of Dixon multiplier matrices with Subdivision and Incremental

The advantages of optimizing vectors t and α are greater when the input system is more and more mixed. Since the method for finding t and α is based on a heuristic, it is not guaranteed to produce optimal values. In example 6, for instance, the original values from the Dixon multiplier algorithm already give an optimal matrix; the heuristic on the other hand gives a worse value. This is mainly due to the fact that optimization is done assuming generic coefficients.

The above heuristic is expensive in time (see example 10). Table 2 shows the correlation between the time taken to interpolate a determinant and the size of a matrix. As can be seen, the size of the resultant matrix is a major bottleneck in the determinant computation.

Dialytic Matrix Size	Interpolation Time (sec)	Rate (Evaluations/sec)
15	1.37	3300
22	12.25	1100
29	52.30	600
36	173.40	290
44	1099.80	150
52	3530.91	87

Table 2: Interpolation timings with 9 parameters

7 Multi-graded Polynomial Systems: Exact Cases

In this section we show how the construction for Dixon Multiplier resultant matrices generates exact matrices in certain cases (including generic unmixed multi-graded systems), without any a priori knowledge about the structure of such polynomial systems.

Below we consider only unmixed polynomial systems with supports $\langle \mathcal{A}_0, \dots, \mathcal{A}_d \rangle$, where $\mathcal{A}_i = \mathcal{A}_j$. For clarity, we will drop the index and denote by $\mathcal{A}$, the support of each polynomial, i.e. $\mathcal{A} = \mathcal{A}_i$ for all $i \in \{0, \dots, d\}$.

One of the operations on supports which was considered in [CK00a], is the **direct sum** on supports.

Definition 7.1 *Given two supports $\mathcal{P} \subset \mathbb{N}^k$ and $\mathcal{Q} \subset \mathbb{N}^l$, define the* **direct sum of $\mathcal{P}$ and $\mathcal{Q}$:**

$$\mathcal{P} \oplus \mathcal{Q} = \{ (p_1, \dots, p_l, q_1, \dots, q_k) \mid p = (p_1, \dots, p_l) \in \mathcal{P} \quad \textbf{and} \quad q = (q_1, \dots, q_k) \in \mathcal{Q} \}.$$

A polynomial is called **homogeneous** if all monomials appearing in the polynomial have the same degree. In terms of the support $\mathcal{A}$, a polynomial is homogeneous of degree n if for $\alpha = (\alpha_1, \dots, \alpha_d) \in \mathcal{A}$, $\alpha_1 + \dots + \alpha_d = n$. Any polynomial can be homogenized by introducing an extra variable. In a sense, each polynomial has a homogeneous and non-homogeneous version.

Definition 7.2 *A polynomial with support $\mathcal{A}$ is called* ***multihomogeneous*** *of type $(l_1, l_2, \ldots, l_r; k_1, k_2, \ldots, k_r)$ for some integers l_i, k_j and r if $\mathcal{A} \subseteq \mathcal{Q}_1 \oplus \cdots \oplus \mathcal{Q}_r$, and $\mathcal{Q}_i \subset \mathbb{N}^{l_i}$ is the support of a homogeneous polynomial of degree k_i, for $i = 1, \ldots, r$.*

By abuse of notation, we will call a polynomial multihomogenous of certain type if it can be homogenized into one. Note that the same polynomial can be multihomogenized in number of different ways. For example, the polynomial

$$f = c_1 x + c_2 y + c_3$$

can be homogenized into

$$f = c_1 x + c_2 y + c_3 z \qquad \text{or} \qquad f = c_1 xt + c_2 ys + c_3 st$$

where the first is of type $(2; 1)$ in terms of variables x, y with the homogenizing variable z and the second is of type $(1, 1; 1, 1)$ in terms of variables x,y, and the homogenizing variables s and t, respectively. Note that in the first case, all monomials of degree 1 are present, whereas in the second case, the monomial xy could have been included with the resulting polynomial being still of type $(1, 1; 1, 1)$.

Proposition 7.1 *Given a unmixed generic polynomial system with polynomial support $\mathcal{A} \subseteq \mathcal{P} \oplus \mathcal{Q}$, where $\mathcal{P} \subset \mathbb{N}^k$ and $\mathcal{Q} \subset \mathbb{N}^l$, then*

$$\Delta_{\mathcal{A}} \subseteq \Delta_{\mathcal{P}} \oplus (\underbrace{\mathcal{Q} + \cdots + \mathcal{Q}}_{k} + \Delta_{\mathcal{Q}}).$$

Proof: A polynomial with support $\mathcal{A}$ has two blocks of variables– one corresponding to $\mathcal{P}$, $\{x_1, \ldots, x_k\}$ and other corresponding to $\mathcal{Q}$, $\{y_1, \ldots, y_l\}$. Obviously, the polynomial is in variables $\{x_1, \ldots, x_k, y_1, \ldots, y_l\}$. The matrix for computing the Dixon polynomial can be split according to $\mathcal{P}$ and $\mathcal{Q}$.

$$\theta(f_0, \ldots, f_d) = \left(\prod_{i=1}^{k} \frac{1}{\overline{x}_i - x_i} \prod_{i=1}^{l} \frac{1}{\overline{y}_i - y_i} \right) \left| \begin{array}{cccc} \pi_0(f_0) & \pi_0(f_1) & \cdots & \pi_0(f_d) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_{k-1}(f_0) & \pi_{k-1}(f_1) & \cdots & \pi_{k-1}(f_d) \\ \hline \pi_k(f_0) & \pi_k(f_1) & \cdots & \pi_k(f_d) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_{k+l-1}(f_0) & \pi_{k+l-1}(f_1) & \cdots & \pi_{k+l-1}(f_d) \\ \hline \pi_{k+l}(f_0) & \pi_{k+l}(f_1) & \cdots & \pi_{k+l}(f_d) \end{array} \right|.$$

Let $J = \{0, \ldots, k + l\}$; for a subset $a = \{a_1, \ldots, a_k\} \subset J$ of size k; let $\widehat{a}$ be such that $a \cup \widehat{a} = J$, i.e. $|\widehat{a}| = l$. Using the Laplace formula for the determinant, the above expression can be expanded in terms of of the determinants of minors coming from each block as

$$\begin{aligned} \theta(f_0, \ldots, f_d) = & \sum_{a \subset J} (\pm 1) \\ & \left(\prod_{i=1}^{k} \frac{1}{\overline{x}_i - x_i} \left| \begin{array}{cccc} \pi_0(f_{a_1}) & \pi_0(f_{a_2}) & \cdots & \pi_0(f_{a_k}) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_{k-1}(f_{a_1}) & \pi_{k-1}(f_{a_2}) & \cdots & \pi_{k-1}(f_{a_k}) \end{array} \right| \right) \times \\ & \left(\prod_{i=1}^{l} \frac{1}{\overline{y}_i - y_i} \left| \begin{array}{cccc} \pi_k(f_{\widehat{a}_1}) & \pi_k(f_{\widehat{a}_2}) & \cdots & \pi_k(f_{\widehat{a}_l}) \\ \vdots & \vdots & \ddots & \vdots \\ \pi_{k+l-1}(f_{\widehat{a}_1}) & \pi_{k+l-1}(f_{\widehat{a}_2}) & \cdots & \pi_{k+l-1}(f_{\widehat{a}_l}) \\ \pi_{k+l}(f_{\widehat{a}_1}) & \pi_{k+l}(f_{\widehat{a}_2}) & \cdots & \pi_{k+l}(f_{\widehat{a}_l}) \end{array} \right| \right). \end{aligned}$$

The support of the first determinant in the above expression in terms of variables $\{x_1, \ldots, x_d\}$ is contained in $\Delta_{\mathcal{P}}$. Also it is not hard to see that the support of the first determinant in terms of variables $\{y_1, \ldots, y_d\}$ is contained in $\underbrace{\mathcal{Q} + \cdots + \mathcal{Q}}_{k}$. Hence the support of the first determinant is contained in

$$\Delta_{\mathcal{P}} \oplus (\underbrace{\mathcal{Q} + \cdots + \mathcal{Q}}_{k}).$$

The support of the second determinant in terms of variables $\{y_1, \ldots, y_l\}$ is exactly $\Delta_{\mathcal{Q}}$. (Note that the second determinant does not contain any variables from $\{x_1, \ldots, x_d\}$.) Combining these, we get that

$$\Delta_{\mathcal{A}} \subseteq \Delta_{\mathcal{P}} \oplus \big(\underbrace{\mathcal{Q} + \cdots + \mathcal{Q}}_{k} + \Delta_{\mathcal{Q}}\big).$$

□

We consider a subclass of multihomogeneous polynomials, called **multigraded** polynomials.

Definition 7.3 *A multihomogeneous polynomial of type $(l_1, l_2, \ldots, l_r; k_1, k_2, \ldots, k_r)$ is called **multigraded** if for each $i = 1, \ldots, r$, either $l_i = 1$ or $k_i = 1$.*

A **multigraded** polynomial system $\mathcal{F}$ of type $(l_1, l_2, \ldots, l_r; k_1, k_2, \ldots, k_r)$ is an unmixed system of $d+1$ generic multigraded polynomials in d variables, where $\sum_{i=1}^{r} l_i = d$. We prove below that the Dixon multiplier matrix for a multigraded system is square, and its determinant is the resultant of $\mathcal{F}$.

Proposition 7.2 *Let $\mathcal{F}$ be a multigraded polynomial system of type $(l; k)$, then*

$$\Delta_{\mathcal{A}} = \{\, p = (p_1, \ldots, p_l) \mid p_1 + \cdots + p_l \le k - 1 \,\}.$$

Proof: If $l = 1$, then the support of the Dixon polynomial corresponds to the monomials $\{1, x, x^2, \ldots, x^{k-1}\}$ used in constructing the Bezout matrix. If $k = 1$, then $\mathcal{F}$ is a linear system of equations. The Dixon polynomial in that case contains a constant term, obtained by expanding the corresponding determinant and performing division by $\overline{x}_i - x_i$. □

Proposition 7.3 *Let $\mathcal{A}$ be the support of a multihomogeneous polynomial of type $(l; k)$. For an integer $n > 0$, $\mathcal{Q} = \underbrace{\mathcal{A} + \cdots + \mathcal{A}}_{n}$ is the support of a multihomogeneous polynomial of type $(l; nk)$, that is*

$$\mathcal{Q} = \{\, p = (p_1, \ldots, p_l) \mid p_1 + \cdots + p_l \le nk \,\} \quad \textit{and} \quad |\mathcal{Q}| = \binom{nk + l}{l}.$$

Proof: Since $\mathcal{A} = \{\, p = (p_1, \ldots, p_l) \mid p_1 + \cdots + p_l \le k \,\}$, the sum of $\mathcal{A}$, n times, contains all points up to nk, which is the support of a multihomogeneous polynomial of type $(l; nk)$. The number of points in the support is $|\mathcal{Q}| = \binom{nk+l}{l}$. □

From the last two propositions, we have:

Proposition 7.4 *Let $\mathcal{F}$ be a multigraded polynomial system of type $(l; k)$ with support $\mathcal{A}$. Then*

$$\underbrace{\mathcal{A} + \cdots + \mathcal{A}}_{n} + \Delta_{\mathcal{A}} = \{\, p = (p_1, \ldots, p_l) \mid p_1 + \cdots + p_l \le nk + k - 1 \,\}.$$

Definition 7.4 *Given a partition of variables $(l_1, \ldots, l_r)$, let*

$$(m_1, \ldots, m_r) = \left\{ (p_{1,1}, \ldots, p_{1,l_1}, p_{2,1}, \ldots, p_{2,l_2}, \ldots, p_{r,1}, \ldots, p_{r,l_r}) \mid \sum_{j=1}^{l_i} p_{i,j} \le m_i,\ i = 1, \ldots, r \right\}.$$

*The r-tuple $(m_1, \ldots, m_r)$ is called a **multi-index.***

It is easy to see that

$$|(m_1, \ldots, m_r)| = \prod_{i=1}^{r} \binom{m_i + l_i}{l_i}.$$

Proposition 7.5 *Given a multigraded polynomial system $\mathcal{F} = \{f_0, \ldots, f_d\}$ of type $(l_1, l_2, \ldots, l_r; k_1, k_2, \ldots, k_r)$,*

$$\Delta_{\mathcal{A}} \quad \subseteq \quad (m_1, m_2, \ldots, m_r), \qquad \textit{where} \quad m_i = k_i - 1 + k_i \sum_{j=1}^{i-1} l_j.$$

Proof: This directly follows from the previous two propositions. □

Let $M_\mathbf{0}$ be the Dixon multiplier matrix constructed from a multigraded polynomial system $\mathcal{F}$ using monomial $\mathbf{x^0} = \mathbf{1}$. Since for the unmixed case, $\Delta_{\mathcal{A}(i,\mathbf{0})} = \Delta_{\mathcal{A}}$,

$$\#_{\mathsf{rows}}(M_\mathbf{0}) \le \big(1 + \sum_{i=1}^{r} l_i\big)|(m_1, m_2, \ldots, m_r)| = \left(1 + \sum_{i=1}^{r} l_i\right) \prod_{i=1}^{r} \binom{m_i + l_i}{l_i} = (d+1) \prod_{i=1}^{r} \binom{m_i + l_i}{l_i},$$

given that $d = \sum_{i=1}^{r} l_i$.

Proposition 7.6 *Let $\mathcal{X} = (m_1, \ldots, m_r)$ and f be a multihomogenous polynomial of type $(l_1, l_2, \ldots, l_r; k_1, k_2, \ldots, k_r)$ with support $\mathcal{P}$. Then, $\mathcal{X} + \mathcal{P} = (m_1 + k_1, \ldots, m_r + k_r)$.*

Thus, the number of columns in $M_\mathbf{0}$ is at most

$$|(m_1 + k_1, m_2 + k_2, \ldots, m_r + k_r)| = \prod_{i=1}^{r} \binom{m_i + k_i + l_i}{l_i}.$$

The degree of the resultant Res of $\mathcal{F}$ is determined by the mixed volumes of the system (see [SZ94]), and is given by

$$\deg \mathrm{Res} = (d+1)\, \mathtt{MixVol}(S_\mathbf{x}(\mathcal{F})) = (d+1)\, d! \mathsf{Vol}(S_\mathbf{x}(\mathcal{F})) = (d+1)\, d! \prod_{i=1}^{r} \frac{k_i^{l_i}}{l_i!}.$$

Proposition 7.7 *For a multigraded system $\mathcal{F}$, let $m_i = k_i - 1 + k_i \sum_{j=1}^{i-1} l_j$.*

(i) The determinant of a maximal minor of $M_\mathbf{0}$ is a projection operator, which has total degree bigger or equal than of the resultant Res*, i.e.,*

$$(d+1) \prod_{i=1}^{r} \binom{m_i + l_i}{l_i} \ge \#_{\mathsf{rows}}(M_\mathbf{0}) \ge \deg \mathrm{Res}, \quad \textit{and} \quad \prod_{i=1}^{r} \binom{m_i + k_i + l_i}{l_i} \ge \#_{\mathsf{cols}}(M_\mathbf{0}) \ge \deg \mathrm{Res},$$

(ii) the number of rows in $M_\mathbf{0}$ is equal to the (total) degree of the resultant Res*, i.e.,*

$$(d+1) \prod_{i=1}^{r} \binom{m_i + l_i}{l_i} \quad = \quad (d+1)\, d! \prod_{i=1}^{r} \frac{k_i^{l_i}}{l_i!}, \quad \textit{and}$$

(iii) the number of columns in $M_\mathbf{0}$ is equal to the number of rows in $M_\mathbf{0}$, hence the matrix $M_\mathbf{0}$ is square, i.e.,

$$(d+1) \prod_{i=1}^{r} \binom{m_i + l_i}{l_i} \quad = \quad \prod_{i=1}^{r} \binom{m_i + k_i + l_i}{l_i}.$$

Proof: (i) This is a consequence of a property of the Dixon matrix [KSY94, Sax97, BEM00] that the determinant of its maximal minor contains a toric resultant. By Theorem 4.3, the Dixon multiplier matrix $M_\mathbf{0}$ has that property as well.

For (ii), we have from (i) that

$$(d+1) \prod_{i=1}^{r} \binom{m_i + l_i}{l_i} \ge \#_{\mathsf{rows}}(M_\mathcal{F}) \ge \deg \mathrm{Res} = (d+1)\, d! \prod_{i=1}^{r} \frac{k_i^{l_i}}{l_i!}.$$

Below we show that for multigraded systems (i.e., for each i, $k_i = 1$ or $l_i = 1$), the lower and upper bounds on the number of rows in $M_\mathbf{0}$ coincide. Consider

$$h_i = \frac{\binom{m_i + l_i}{l_i}}{k_i^{l_i}/l_i!} = \left.\frac{\binom{t_i k_i}{l_i}}{k_i^{l_i}/l_i!}\right|_{k_i = 1 \text{ or } l_i = 1} = \frac{t_i!}{t_{i-1}!},$$

where $t_i = \sum_{j=1}^{i} l_j$. Thus, $\prod_{i=1}^{r} h_i = t_r! = d!$, proving (ii).

For (iii), for multigraded systems (i.e., for each i, $k_i = 1$ or $l_i = 1$), consider

$$g_i = \frac{\binom{m_i+k_i+l_i}{l_i}}{\binom{m_i+l_i}{l_i}} = \left.\frac{\binom{(t_i+1)k_i}{l_i}}{\binom{t_ik_i}{l_i}}\right|_{k_i=1 \text{ or } l_i=1} = \frac{t_i+1}{t_{i-1}+1}.$$

The product $\prod_{i=1}^{r} g_i = t_r + 1 = l_1 + \cdots + l_r + 1 = d + 1$, and hence, (iii) follows. $\Box$

For a multigraded polynomial system $\mathcal{F}$, the Dixon multiplier matrix is thus square, and its size is exactly the degree of the resultant of $\mathcal{F}$. From the fact that the Dixon multiplier matrix of a polynomial system $\mathcal{F}$ contains a multiple of its resultant, it follows that the determinant of this matrix gives exactly the resultant. We thus have:

Theorem 7.1 *Given an unmixed generic multigraded system $\mathcal{F}$, the determinant of its Dixon multiplier matrix is exactly the resultant of $\mathcal{F}$. Moreover, depending on the order of variable blocks, there exist $r!$ such Sylvester-type matrices.*

In [SZ94] it has been shown that for multigraded systems of type $(l_1, l_2, \ldots, l_r; k_1, k_2, \ldots, k_r)$, the multi-index $(m_1, \ldots, m_r)$, where $m_i = (k_i - 1)l_i + k_i \sum_{j=1}^{i-1} l_j$, constitutes the multiplier sets for constructing a square Sylvester-type resultant matrix whose determinant is nonzero for generic coefficients. For generic multigraded systems, the multiplier set used for each polynomial in the Dixon multiplier matrix construction is precisely the same as the multi-index in [SZ94].

The Dixon multiplier construction not only results in exact Sylvester-type resultant matrices for generic unmixed multigraded systems but for a much wider class of polynomial systems, without any prior knowledge about the structure of the polynomial systems. This is in contrast to the construction proposed in [SZ94] which only applies for unmixed generic multigraded systems.

8 Conclusions

For multivariate polynomial systems, a new algorithm for constructing Sylvester-type matrices, called the Dixon multiplier matrices, is introduced based on the Dixon formulation, for simultaneously eliminating many variables. The resulting matrices are sparse i.e., their size is determined by the supports of the polynomials in a polynomial system. However, unlike other algorithms for constructing sparse resultant matrices which explicitly use the support structure of the polynomial system in the construction, the proposed algorithm exploits the sparse structure only implicitly, just like the generalized Dixon matrix construction.

The algorithm uses an arbitrary term to construct the multiplier sets for each polynomial in the polynomial system. In the unmixed case, it is shown that the Dixon multiplier matrices are of the smallest size if the term used in the construction is from the support. The size of a Dixon multiplier matrix puts an upper bound on the degree of the projection operator that can be extracted from it, thus also determining an upper bound on the degree of the extraneous factor, in case the projection operator is not exactly the resultant. Consequently, the degree of extraneous factors can be minimized by minimizing the size of the associated Dixon multiplier matrices. It is also shown that it does not matter what term is picked from the support for unmixed cases.

It is also shown that for generic multigraded polynomial systems, the Dixon multiplier matrices constructed using the proposed method are exact in the sense that their determinant is the resultant of the polynomial system.

In the mixed case, however, the choice of a term for construction becomes crucial. It is shown that if a term is selected from a nonempty intersection of the supports of the polynomials in a polynomial system, then the resulting Dixon multiplier matrices are smaller. Since translation of supports does not affect the size of the Dixon multiplier matrices, translating supports so as to maximize the overlap among translated supports and selecting a term from this overlap can be formulated as an optimization problem.

The new method is compared theoretically and empirically with other methods for generating Sylvester-type resultant matrices, including subdivision and incremental algorithms for constructing sparse resultants proposed in [CE00]. These results theoretically confirm the practical advantages of the Dixon resultant formulation, which have been observed in a number of applications.

References

[AW92] W.A. Adkins and S.H. Weintraub. *Algebra, An Approach via Module Theory.* Springer-Verlag, New York, 1992.

[BEM00] Laurent Buse, Mohamed Elkadi, and Bernar Mourrain. Generalized resultants over unirational algebraic varieties. *J. Symbolic Computation*, 29:515–526, 2000.

[CE00] J.F. Canny and I.Z. Emiris. A subdivision based algorithm for the sparse resultant. *J. ACM*, 2000.

[CK00a] A. D. Chtcherba and D. Kapur. Conditions for Exact Resultants using the Dixon formulation. *ISSAC00*, pages 62–70, 2000.

[CK00b] A. D. Chtcherba and D. Kapur. Extracting Sparse Resultant Matrices from the Dixon Resultant Formultation. *Proc. of 7th Rhine Workshop (RCWA'00)*, pages 167–182, 2000.

[CK02a] A. D. Chtcherba and D. Kapur. On Relationship between the Dixon-based Resultant Construction and the Supports of Polynomial Systems. Technical Report To appear, Computer Science Dept., Univ. of New Mexico, Albuquerque, New Mexico, USA, 2002.

[CK02b] A. D. Chtcherba and D. Kapur. Resultants for Unmixed Bivariate Polynomial Systems using the Dixon formulation. *Submited to JSC*, May 2002.

[CLO98] D. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry.* Springer-Verlag, New York, first edition, 1998.

[CM96] J.P. Cardinal and B. Mourrain. Algebraic approach of residues and applications. In J. Renegar, M. Shub, and S. Smale, editors, *AMS-SIAM Summer Seminar on Math. of Numerical Analysis, Lectures in Applied Mathematics*, volume 32, pages 189–210. Am. Math. Soc. Press, 1996.

[Dix08] A.L. Dixon. The eliminant of three quantics in two independent variables. *Proc. London Mathematical Society*, 6:468–478, 1908.

[EC95] I.Z. Emiris and J.F. Canny. Efficient incremental algorithms for the sparse resultant and the mixed volume. *J. Symbolic Computation*, 20(2):117–149, August 1995.

[EM99] Ioannis Z. Emiris and Bernard Mourrain. Matrices in elimination theory. *Journal of Symbolic Computation*, 28(1–2):3–43, 1999.

[GKZ94] I.M. Gelfand, M.M. Kapranov, and A.V. Zelevinsky. *Discriminants, Resultants and Multidimensional Determinants.* Birkhauser, Boston, first edition, 1994.

[KS95] D. Kapur and T. Saxena. Comparison of various multivariate resultants. In *ACM ISSAC 95*, Montreal, Canada, July 1995.

[KS96] D. Kapur and T. Saxena. Sparsity considerations in the Dixon resultant formulation. In *Proc. ACM Symposium on Theory of Computing*, Philadelphia, May 1996.

[KSY94] D. Kapur, T. Saxena, and L. Yang. Algebraic and geometric reasoning using the Dixon resultants. In *ACM ISSAC 94*, pages 99–107, Oxford, England, July 1994.

[Mac16] F.S. Macaulay. The algebric theory of modular systems. *Cambridge Tracts in Math. and Math. Phys.*, 19, 1916.

[PS93] P.Pedersen and B. Sturmfels. Product formulas for resultants and chow forms. *Math. Zeitschrift*, 214:377–396, 1993.

[Sax97] T. Saxena. *Efficient variable elimination using resultants.* PhD thesis, Department of Computer Science, State Univeristy of New York, Albany, NY, 1997.

[Sha94] I.R. Shafarevich. *Basic Algebraic Geometry.* Spring-Verlag, New-York, second edition, 1994.

[Syl53] J.J Sylvester. On a theory of syzygetic relations of two rational integral functions, comprising an application to the theory of sturm's functions, and that of the greatest algebraic common measure. *Philosophical Trans.*, 143:407–548, 1853.

[SZ94] B. Sturmfels and A. Zelevinski. Multigraded resultants of sylvester type. *Journal of Algebra*, 163:115–127, 1994.

[Zha00] M. Zhang. *Topics in Resultants and Implicitization.* PhD thesis, Rice University, Dept. of Computer Science, May 2000.