Indian Journal of Science and Technology, *Vol 9(46), DOI: 10.17485/ijst/2016/v9i46/107082, December 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Comparison of Some Multivariable Hybrid Resultant Matrix Formulations

Surajo Sulaiman[1,2] and Nor'aini Aris[1]*

[1]Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia 81310, UTM Johor Bahru, Johor, Malaysia; ssurajo2@live.utm.my , noraini@utm.my
[2]Department of Mathematics, Faculty of Science, Northwest University Kano 700221, Kano-Nigeria; surajsuleiman@yahoo.com

Abstract

Objective: To evaluate and compare different hybrid resultant formulations in relation to computational complexity, performance and optimality condition. **Methods/Statistical Analysis:** Hybrid matrices are evaluated using computer algebra system. **Findings:** we have shown that, none of the hybrid formulation works well with the exception of HDP_4. However, after deleting the zero rows and columns the resulting matrix may not be a square matrix, on the other hand, we studied and established that none of the hybrid formulation produces a square matrix in general and likewise predicted that, the density or sparseness of the polynomial equations does not influence the performance of these hybrid matrix formulations. **Applications/Improvements:** This comparison reveals that, the existing hybrid methods for computing resultant are not efficient and therefore there is need for another formulations with will focus on the current limitations described in this paper.

Keywords: Hybrid Matrix, Hybrid Resultant, Resultant, Resultant Matrix, System of Polynomials

1. Introduction

Resultant is polynomials in the coefficients of $f_1, f_2, \ldots, f_n$ which vanishes if and only if $f_1, f_2, \ldots, f_n$ have a common solution. The notion is also called eliminant, being one of the efficient tools of eliminating a set of parameters in a given system of polynomials[1]. Currently, there are two types of constructions which depend on the nature of the resultant matrix. The Sylvester-type construction uses the coefficients of the system of polynomials, while the Cayley or Dixon-type is in the form of Bezout matrix. Construction methods such as Macaulay, Newton sparse and incremental are considered to be of Sylvester-type, while the Dixon matrix as the name implies is regarded as Dixon type. Sometime a resultant matrix combines the two structures of the constructions and is referred to the hybrid resultant matrix. For details on the Sylvester-type construction, refer to[2-4]. Details on the Dixon type construction can be found in the work of [5-10].The foundation work for hybrid resultant was first introduced in[11], derived for certain class of the multivariate polynomials. Independently, in 1999[12] had proposed another hybrid construction which possibly the first construction that can be applied to more general class of the system of polynomials. Apart from the classical hybrid resultant matrix, the sparse hybrid formulation was constructed; this is due to the frequent appearance of such systems in many engineering applications[13].

However, it is not clear whether or not the constructions can generate exact resultant. Another construction was given by[14] and unlike the work of[13], Khetan gives an example which he does not solve but gives the dimensions of the resultant matrix. His construction only considers systems of polynomials with unmixed support and the size of the matrix is very large[14]. A complete implementation of the Sylvester-Bezout construction is given by[15] in her PhD thesis and finds out that, one of the shortcomings of the construction is the inability of the method to

**Author for correspondence*

produce an optimal matrix when the polynomials have mixed support.

2. Related Work

The foundation work on polynomial resultant was laid in[2,5] for univariate system of polynomials and subsequently generalized to the multivariate case by[10,16], matrix based method of computing resultant was popular due to its less computational complexity and low memory requirement compared to Groebner basis and Ritt-wu method[17], however matrix based method comprises three routine during the computation of the resultant; the computation of symbolic determinant, identifying the superfluous factor and the extraction of the projection operator. The performance of the existing matrix based methods are compared in[7] and reported various limitations with the existing method. In ability of all matrix based methods to generate the exact resultant lead to the construction of hybrid resultant formulations, on the other hand, it is not clear whether or not these hybrid formulations could generates the exact resultants

Example 1: Consider the three bivariate systems of polynomial equations.

$$F=\begin{cases} f_1 = xy - y^2 + (a+2)x - 3y + a^2 + a - 2 \\ f_2 = ax^2 + 3a^2xy + xy + 3ay^2 \\ f_3 = x^2 - 2y^2 - xy + (a-1)x - 2(a-1)y \end{cases}$$

Eliminating x and y in the system of example 1 using the Groebner basis generate the following:-

$$R = 6a^{10} + 25a^9 - a^8 - 94a^7 - 26a^6 + 131a^5 + 17a^4$$

$$-62a^3 - 4a^2 + 8a$$

Which completely contain the common zero of the example 1, while the matrix based method produced the following:

$(3a+2)(2a+1)(3a^2+6a+1)(a-1)^2 R$ .Note that $(3a+2),(2a+1)$ and $(a-1)^2$ are already contain in R

and the zeros of $(3a^2+6a+1)$ did not provide any information on the common zero of system of polynomials which make them superfluous, refer to [7] for the details of the existing matrix method. Our work compared the existing hybrid matrix method of computing resultant of multivariate polynomial systems and proposes a new method based on the limitations of these hybrid formulations.

3. Multivariate Resultant Formulation

Consider the following bivariate polynomials:

$$\begin{aligned} f(x,y) &= \sum_{i=0}^{s}\sum_{j=0}^{t} a_{i,j}x^i y^j \\ g(x,y) &= \sum_{k=0}^{s}\sum_{l=0}^{t} a_{k,l}x^k y^l \qquad (1) \\ h(x,y) &= \sum_{p=0}^{s}\sum_{q=0}^{t} a_{p,q}x^p y^q \end{aligned}$$

We shall adopt the format used in[18] to construct various types of formulations.

3.1 The $P_{s,t}^{x>y}$ Sylvester Matrix

Let $J=[f,g,h]$ and $P_{s,t}^{x>y}$ be the matrix of the coefficients of $x^v y^u J$ with $0 \le v \le 2s-1$ and $0 \le t \le t-1$, which is of Sylvester- type. Fixing a lexicographical order $x>y$, we have the following:

$$\begin{bmatrix} 1 \\ \vdots \\ \vdots \\ x^u y^v \\ x^u y^{v+1} \\ \vdots \\ \vdots \\ x^{3s-1} y^{t-1} \end{bmatrix}^T P_{s,t}^{x>y}$$

According to[19], the matrix $P_{s,t}^{x>y}$ has dimension $6st \times 6st$, such that different variable orderings produce different matrices which are square matrices in each case. The block structure of the matrix $P_{s,t}^{x>y}$ given by[20] is the matrix:

$$P_{s,t}^{x>y} = \begin{bmatrix} M_0 & & & & & \\ \vdots & \ddots & & & & \\ \vdots & & \ddots & & & \\ M_{s-1} & \cdots & \cdots & M_0 & & \\ M_S & \cdots & \cdots & M_1 & M_0 & \\ & \ddots & & \vdots & \vdots & \ddots \\ & & \ddots & \vdots & \vdots & & \ddots \\ & & & M_S & M_{S-1} & \cdots & \cdots & M_0 \\ & & & & M_S & \cdots & \cdots & M_1 \\ & & & & & \ddots & & \vdots \\ & & & & & & \ddots & \vdots \\ & & & & & & & M_S \end{bmatrix}$$

Therefore the resultant is $|P_{s,t}^{x>y}|$ where each block $M_i, 0 \le i \le s$ has the following structure:

$$M_i = \begin{bmatrix} J_{i,0} & & & \\ \vdots & \ddots & & \\ \vdots & & \ddots & \\ J_{i,t-1} & \cdots & \cdots & J_{i,0} \\ J_{i,t} & \cdots & \cdots & J_{i,1} \\ & \ddots & & \vdots \\ & & \ddots & \vdots \\ & & & J_{i,t} \end{bmatrix}, \text{ with}$$

$$J_{i,j} = \begin{bmatrix} a_{i,j} & b_{i,j} & c_{i,j} \end{bmatrix}$$

3.2 The $D_{s,t}^{x>y}$ Resultant Matrix

Consider the system of polynomials equations (1), the Dixon polynomial of the three equations is

$$\Delta_{s,t}(x,y,\alpha,\beta) = \frac{1}{(\alpha - x)(\beta - y)} \begin{vmatrix} f(x,y) & g(x,y) & h(x,y) \\ f(\alpha,y) & g(\alpha,y) & h(\alpha,y) \\ f(\alpha,\beta) & g(\alpha,\beta) & h(\alpha,\beta) \end{vmatrix}$$

Fixing a lexicographical order $x > y$, $\alpha > \beta$ and due to[19], we have the following:

Therefore, $|D_{s,t}^{x>y}|$ is the Cayley's resultant of the polynomials f, g and h. The matrix $D_{(s,t)}^{(x>y)}$ can be represented by the following[18]:

$$D_{s,t}^{x>y} = \begin{bmatrix} B_{0,0} & \cdots & \cdots & B_{0,0} \\ \vdots & \ddots & . & \vdots \\ \vdots & . & \ddots & \vdots \\ B_{s-1,0} & \cdots & \cdots & B_{0,0} \end{bmatrix}$$

with each block $B_{i,j} = \sum_{k=0}^{\min(i,2s-1-j)} M_{i-k} F_{j+k}$,

$0 \le i \le s-1$ and $0 \le j \le 2s-1$. The entries of the matrix F_j are computed in. Another way to compute the entries of $D_{s,t}^{x>y}$ directly is using the following formula.

$$\Delta_{s,t}(x,y,\alpha,\beta) = \begin{bmatrix} 1 \\ \vdots \\ \vdots \\ x^u y^v \\ x^u y^{v+1} \\ \vdots \\ \vdots \\ x^{n-1} y^{2m-1} \end{bmatrix}^T D^{x>y} \begin{bmatrix} 1 \\ \vdots \\ \vdots \\ \alpha^u \beta^v \\ \alpha^u \beta^{v+1} \\ \vdots \\ \vdots \\ \alpha^{2n-1} \beta^{m-1} \end{bmatrix}$$

3.3 Hybrid of $D_{s,t}^{x>y}$ and $P_{s,t}^{x>y}$

In[20] different types of the hybrid resultant are formulated one of the construction uses the matrices $D_{s,t}^{x>y}$ and $P_{s,t}^{x>y}$, which is briefly described as follows:

$$HDP_j = \begin{bmatrix} M_0 & & & & B_{0,0} & \cdots & \cdots & B_{0,2s-1} \\ \vdots & \ddots & & & \vdots & \ddots & & \vdots \\ \vdots & & \ddots & & \vdots & & \ddots & \vdots \\ M_{s-1} & \cdots & \cdots & M_0 & B_{s-1,0} & \cdots & \cdots & B_{s-1,2s-1} \\ M_s & \cdots & \cdots & M_1 & & & & \\ & \ddots & & \vdots & & & & \\ & & \ddots & \vdots & & & & \\ & & & M_s & & & & \end{bmatrix} \quad (2)$$

Here, HDP_j is the j^{th} hybrid of $D_{s,t}^{x>y}$ and $P_{s,t}^{x>y}$, and consists of the first $2s - j$ columns of $P_{s,t}^{x>y}$ on the left combined with the first j^{th} columns of $D_{s,t}^{x>y}$ on the right. For a bivariate system of degree two, the following are some of the hybrid settings:

$$HDP_0 = \begin{bmatrix} M_0 & 0 & 0 & 0 \\ M_1 & M_0 & 0 & 0 \\ M_2 & M_1 & M_0 & 0 \\ 0 & M_2 & M_1 & M_0 \\ 0 & 0 & M_2 & M_1 \\ 0 & 0 & 0 & M_2 \end{bmatrix}$$

$$HDP_1 = \begin{bmatrix} M_0 & 0 & 0 & B_{0,0} \\ M_1 & M_0 & 0 & B_{1,0} \\ M_2 & M_1 & M_0 & 0 \\ 0 & M_2 & M_1 & 0 \\ 0 & 0 & M_2 & 0 \end{bmatrix}$$

$$HDP_2 = \begin{bmatrix} M_0 & 0 & B_{0,0} & B_{0,1} \\ M_1 & M_0 & B_{1,0} & B_{1,1} \\ M_2 & M_1 & 0 & 0 \\ 0 & M_2 & 0 & 0 \end{bmatrix}$$

$$HDP_3 = \begin{bmatrix} M_0 & B_{0,0} & B_{0,1} & B_{0,2} \\ M_1 & B_{1,0} & B_{1,1} & B_{1,2} \\ M_2 & 0 & 0 & 0 \end{bmatrix}$$

and

$$HDP_4 = D_{s,t}^{x>y} \begin{bmatrix} B_{0,0} & B_{0,1} & B_{0,2} & B_{0,3} \\ B_{1,0} & B_{1,1} & B_{1,2} & B_{1,3} \end{bmatrix}$$

where each block M_i has size $2t \times 3t = 4 \times 6$ and each $B_{i,j} = 2t \times t = 4 \times 2$, for example

$$B_{0,0} = \begin{bmatrix} |10,01,00| & |10,02,00| \\ |10,01,00| + |11,01,00| & |10,02,01| + |11,02,00| \\ |11,02,00| + |12,01,00| & |11,02,01| + |12,02,00| \\ |12,02,00| & |12,02,01| \end{bmatrix}$$

where each block is given by a bracket define in (3)

$$|ij,kl,pq| = \begin{vmatrix} a_{i,j} & b_{i,j} & c_{i,j} \\ a_{k,l} & b_{k,l} & c_{k,l} \\ a_{p,q} & b_{p,q} & c_{p,q} \end{vmatrix} \quad (3)$$

3.4 Khetan's Hybrid Matrix

Consider the systems of polynomials $f_1, f_2, f_3 \in \mathbb{C}[x_1, x_2, x_1^{-1}, x_2^{-1}]$ that is f_i are Laurent polynomials in n variables. The set of exponents of the given system is called the support and the Newton polytope of the system is the convex hull of the support of such polynomials. The main result of

Theorem 1[14] The resultant of a system $f_1, f_2, f_3 \in \mathbb{C}[x_1, x_2, x_1^{-1}, x_2^{-1}]$ with a common Newton polytope $\mathbf{Q}$ is the determinant of the block matrix

$$\begin{bmatrix} B & L \\ \overline{L} & 0 \end{bmatrix},$$

where L and $\overline{L}$ are the coefficients of the polynomials and the entries of B are of the form

$$[abc] = \begin{vmatrix} C_{1a} & C_{1b} & C_{1c} \\ C_{2a} & C_{2b} & C_{2c} \\ C_{3a} & C_{3b} & C_{3c} \end{vmatrix}$$

According to Theorem 1, the hybrid matrix for the system $f_i = C_{i1} + C_{i2}x + C_{i3}y + C_{i4}xy + C_{i5}x^2y + C_{i6}xy^2$ such that $i = 1,2,3$, is give by

$$\begin{bmatrix} 0 & [124] & 0 & [126]-[234] & -[234] & -[236] & C_{11} & C_{21} & C_{31} \\ 0 & 0 & 0 & 0 & 0 & 0 & C_{12} & C_{22} & C_{32} \\ 0 & [126]-[135] & 0 & [146]-[236] & [156]+[345] & [346] & C_{13} & C_{23} & C_{33} \\ 0 & -[145] & 0 & [156]-[345] & [256] & [356] & C_{14} & C_{24} & C_{34} \\ 0 & 0 & 0 & 0 & 0 & 0 & C_{15} & C_{25} & C_{35} \\ 0 & [156] & 0 & [356] & [456] & 0 & C_{16} & C_{26} & C_{36} \\ C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} & 0 & 0 & 0 \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} & 0 & 0 & 0 \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} & 0 & 0 & 0 \end{bmatrix}$$

4. Result and Discussions

In this section, we shall evaluate different types of hybrid formulations and present the advantages as well as limitations of each setting.

4.1 Fifth Hybrid Formulation

Consider the hybrid setting (2) from Section 3.3, the fifth hybrid resultant matrix consists of eight blocks each having $2t \times t$ dimensions with entries of the block given as the sum of the brackets is given by (3). We will illustrate the performance of this setting using the following example.

Example 2 Consider the three bivariate systems of polynomial equations of degree two

$$F = \begin{cases} f_1 = (u^2 + u^2w)x - u^2w + w + u^2 + 1 \\ f_2 = \left(u^2 + u^2w\right)y - u^2w - u + w \\ f_3 = \left(u^2 + u^2w\right)z - 2u^2 + 2w + 2 \end{cases} \quad (4)$$

Eliminating u, w using HDP_4 gives the following matrix in terms of which is of dimension 8×8, but the determinant of this matrix fails to give the required information since the determinant is zero. However deleting the zero rows and columns produces a maximal minor whose determinant contains the resultant. Therefore the resultant R is:-

$$\begin{aligned} R = &-16x^4 - 16x^3z + 12x^2z^2 + 8xz^3 - 4z^4 \\ &-144yz^2 + 4z^3 - 240x^2 + 1152xy - 192xz \\ &-176x^3 + 288x^2y - 96x^2z + 144xyz + 84xz^2. \\ &-1296y^2 + 288yz + 12z^2 - 656x + 1152y \\ &-128z - 208 \end{aligned}$$

This resultant agrees with the result of Dixon formulation obtained in (1).

Limitations: The resultant matrix after deleting the zero rows and columns may not be a square, refer to the

$$\begin{bmatrix} 0 & 0 & 2x-z+4 & 0 & 2x-z-2 & 0 & -2x+6y+z-4 & 0 \\ 0 & 0 & 2x-z-2 & 0 & 2x-z-2 & 0 & -4x+6y-z-2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2x-z+4 & 0 & 2x-z+4 & 0 & -2x+6y+z-4 & 0 & 2x-2z-2 & 0 \\ 2x-z-2 & 0 & 2x-z-2 & 0 & -4x+6y-z-2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

table 1 system C4 for example. The size of the matrix is large when compared with the existing method such as Dixon matrix and the complexity of computing this hybrid matrix is high.

4.2 Fourth Hybrid Formulation

Adopting the system (4) from example 1, this formulation produces a 12×12 HDP_3 resultant matrixes of which four of the rows and three of the columns have zero entries. However, deleting these zero rows and columns generate a matrix with 8×9 dimensions,

$$\begin{bmatrix} 1 & 0 & 2 & 0 & 0 & 0 & 2x-z+4 & 2x-z-2 & -2x+6y+z-4 \\ 1 & 1 & 2 & 1 & 0 & 2 & 2x-z-2 & 2x-z-2 & -4x+6y-z-2 \\ 0 & 0 & 0 & 1 & 1 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 2x-z+4 & -2x+6y+z-4 & 2x-2+2z \\ 0 & 0 & 0 & 0 & -1 & 0 & 2x-z-2 & -4x+6y-z-2 & 0 \\ x+1 & y & z-2 & 0 & 0 & 0 & 0 & 0 & 0 \\ x-1 & y-1 & z & x+1 & y & z-2 & 0 & 0 & 0 \\ 0 & 0 & 0 & x-1 & y-1 & z & 0 & 0 & 0 \end{bmatrix}$$

which implies that the determinant cannot be computed.

Limitations: The size of the resultant matrix is very large and the setting fails to provide any information on the common roots of the polynomials when the systems are sparse. The complexity during the computation is high which include the removal of the zero rows and columns from the hybrid matrix.

4.3 Third Hybrid Formulation

For the same system in example 1, the third hybrid formulation produces a 16×16 HDP_2 matrix of which four of the rows and two of the columns have zero entries, however, deleting the zero rows and columns generates a matrix of dimensions 12×14, refer to figure 1 of the next page, since the matrix is not square, the determinant cannot be found.

4.4 Second Hybrid Formulation

For the same system in example 1, the third hybrid formulation produces a 20×20 HDP_1 matrix of which five rows and one of the column are zeros, however, deleting the zero rows and column generates a matrix of dimensions 15×19 (the matrix is too large to be included here) which implies that the resultant cannot be extracted from the projection operator, since the determinant cannot be computed.

Limitations: The size of this resultant matrix is very large and could hardly produce a square matrix which makes the computation of the projection operator very difficult and increase the computational complexity of the hybrid matrix.

4.5 First Hybrid Formulation

Referring to the system of example 1, the first hybrid formulation produces a 24×24 HDP_0 matrix of which six of the rows have zero entries, on the other hand, deleting the zero rows generates a matrix of dimensions 18×24 (the matrix is too large to be included here), which also implies that the determinant cannot be computed.

Limitations: The size of this hybrid matrix is extra large and sparse in nature which easily leads this matrix to be singular. Even though, we can apply the method proposed in[8]to extract the projection operator, however the complexity involves in that process is very high.

4.6 Khetan's Hybrid Formulation

We shall use another example to evaluate this resultant formulation and compare the result with the previous formulations.

Example 3: Consider the three systems of polynomial equations

$$F=\begin{cases} f_1 = x^2y - xy^2 + 2yz^2 + xz + 1 \\ f_2 = 2x^2y + 3xy^2 + 2xyz^4 - xz^3 + y \\ f_3 = x^2y + xy^2 + xy - xz - z - 1 \end{cases}$$

Computing the resultant using Khetan's formulation yields a hybrid resultant matrix, whose determinant produces the following resultant

$$\begin{aligned} R &= 1024z^{26} + 1024z^{25} + 512z^{24} + 3136z^{22} + 4672z^{21} \\ &+ 960z^{20} - 4000z^{19} - 1936z^{18} + 2384z^{17} - 760z^{16} \\ &- 1888z^{15} - 1748z^{14} + 5192z^{13} + 11294z^{12} + 16606z^{11} \\ &+ 3555z^{10} - 23208z^{9} - 28154z^{8} - 17260z^{7} - 912z^{6} \\ &+ 13475z^{5} + 12371z^{4} + 4768z^{3} - 100z^{2} + 144z + 36. \end{aligned}$$

This multivariate resultant agrees with the result obtained by the hybrid HDP_4 and Groebner basis, which in both cases produced an irreducible polynomial of degree 26.

Limitations: If any of the system has a Newton polytope parallel to either the $x-$ axis or $y-$ axis, they fail to generate the Bezout matrix which makes the computation impossible.

4.7 Analysis of the result

We measure the sparseness of the matrix using the relation proposed in[21,22] as

$$\text{density} = \frac{\text{number of non} - \text{zero elements}}{\text{number of all elements}}.$$

Table 1 compares the performance of the two formulations using some selected examples (refer to an appendix), it is clear that HDP_4 performs better compared to the remaining hybrid, even though for the system C3, the determinant of the hybrid matrix after removing the zero rows and columns gives exactly zero which gives no information at all while for the same system a projection operator can be obtain using some method other than the hybrid formulation, on the other hand, the Khetan's formulation solved only one out of the ten selected examples, failing to provide the required result for the rest of the nine questions. The hybrid formulations such as HDP_0, HDP_1, HDP_2 and HDP_3 fails to provide any information on the resultant for the system (4) where the settings are purely sparse in nature. However these formulations produce some extraneous factors when the polynomials are dense. Although HDP_4 performs perfectly after deleting the zero rows and columns giving a square matrix with a non-zero determinant, it is actually another way of expressing the Dixon formulation. For any system of polynomials, HDP_4 also produces extraneous factors or fail to give the square matrix whenever Dixon formulation does (see C4 from table 1), as such all the limitations of the Dixon are inherited by HDP_4. The Khetan's formulation produces exact resultant for an unmixed system of polynomials. However, if any of the system has a Newton polytope parallel to either the $x-$ axis or $y-$ axis, the polynomials fail to generate the

Appendix Some selected system of polynomials

Cases	System of polynomial equations
C1	$_{11}x^2 + a_{12}xy + a_{13}xz + a_{22}y^2 + a_{23}yz + a_{33}z^3, f_2 = b_{11}x^2 + b_{12}xy + b_{13}xz + \quad b_{22}y^2 +$ and $f_3 = c_1x + c_2y + c_3z$
C2	$f_1 = a_1x^2y^2 + a_2x^2, f_2 = b_1x^2y^2 + b_2y^2$ and $f_3 = u_1x + u_2y + u_3$
C3	$f_1 = a_1x^2 - a_2y^2, f_2 = b_1x^2 - b_2y^2 + b_3xy$ and $f_3 = y - x + z$
C4	$f_1 = ax^2 + bxy + (b + c - a)x + ay + 3(c-1), f_2 = 2a^2x^2 + 2abxy + aby + b^3$ and $f_3 = 4(a-b)x + c(a+b)y + 4ab$
C5	$f_1 = x^2 + b_1x - y^2 + a_0, f_2 = y^2 + b_2y - z^2 + b_0$ and $f_3 = z^2 + b_3z - x^2 + c_0$ $f_3 = z^2 + b_3z - x^2 + c_0$
C6	$f_1 = c_{11} + c_{12}x + c_{13}y + c_{14}xy + c_{15}x^2y + c_{16}xy^2$ $f_2 = c_{21} + c_{22}x + c_{23}y + c_{24}xy + c_{25}x^2y + c_{26}xy^2$ $f_3 = c_{31} + c_{32}x + c_{33}y + c_{34}xy + c_{35}x^2y + c_{36}xy^2$
C7	$f_1 = c_{11} + c_{12}x + c_{13}y + c_{14}xy, f_2 = c_{21} + c_{22}x + c_{23}y + c_{24}x^2$ and $f_3 = c_{31} + c_{32}x + c_{33}y + c_{34}y^2$
C8	$f_1 = a_0x - a_1y + a_2xy, f_2 = b_0 + b_1y + b_2y^2$ and $f_3 = c_0 + c_1xy + c_2x^2$
C9	$f_1 = (s^2 + t^2 + 1)x - 2st$, $f_2 = (s^2 + t^2 + 1)y - 2t$ and $f_3 = (s^2 + t^2 + 1)z - 2s$
C10	$f_1 = 4x^2 + 5x + 6y^2 + 3yz + 5y + 1, f_2 = 5x^2 + xy + 2xz + 6z^2 + 3z + 3$ and $f_3 = 6xz + 5y^2 + 2y + 4z^2 + 6z + 5$

Table 1. Comparison of two hybrid matrix

	No. Poly. System	HDP_4				Khetan' Resultant		
		Size	Reduced Size	Density	C.P.U time	Size	Density	C.P.U Time
C1	3	8×8	5×5	0.8800	0.0	9×9	nil	Nil
C2	3	8×8	8×8	0.4375	0.062	9×9	nil	Nil
C3	3	8×8	5×5	0.6400	0.0	9×9	nil	Nil
C4	3	8×8	3×2	1.0000	nil	9×9	nil	Nil
C5	3	8×8	4×4	0.8750	0.0	9×9	nil	Nil
C6	3	8×8	5×5	0.8800	0.015	9×9	0.6296	0.062
C7	3	8×8	3×3	1.0000	0.0	9×9	nil	Nil
C8	3	8×8	4×4	0.625	0.0	9×9	nil	Nil
C9	3	8×8	5×5	0.6371	0.0	9×9	nil	Nil
C10	3	8×8	5×5	0.9200	0.0	9×9	nil	Nil

Bezout matrix[24]. On the other hand, these systems are not often encountered in the real life application[25].

$$\begin{bmatrix}
1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2x-z+4 \\
1 & 1 & 2 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2x-z-2 \\
0 & 0 & 0 & 1 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 0 & 2x-z+4 & 2x-z+4 \\
0 & 0 & 0 & 0 & -1 & 0 & 1 & 1 & 2 & 1 & 0 & 2 & 2x-z-2 & 2x-z-2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 & 0 & 0 \\
x+1 & y & z-2 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
x-1 & y-1 & z & x+1 & y & z-2 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & x-1 & y-1 & z & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & x+1 & y & z-2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & x-1 & y-1 & z & x+1 & y & z-2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x-1 & y-1 & z & 0 & 0
\end{bmatrix}$$

Figure 1. Resultant matrix of HDP_2

5. Future work

Dixon method is known for considerable size of its matrix, even though the entries are more complicated compared to other formulations, however, Jouanolou matrix is also known to produce the small matrix and the entries are not as complicated as that of Dixon matrix, combing the two matrices will also produce another matrix with considerable size, the determinant of this matrix will likely produce a projection operator with little or no extraneous factors.

6. Conclusion

In this paper we have shown that, hybrid formulations considered for this comparison do not work well with the exception of HDP_4 which in general produce gives a square matrix after reducing it to a maximal minor. However, the determinant of HDP_4 resultant matrix can still be zero, it is known from[19] that these hybrid setting have less computational complexity compared to the standard existing method, perhaps this is the only advantage of this hybrid formulation. On the other hand Khetan's hybrid resultant produces exact resultant for unmixed polynomials. This is not surprising as it is shown in[23] that the Dixon resultant formulation can also produce exact resultant for the same generic unmixed polynomials. However, comparing the two hybrids HDP_4 and Khetan's formulation reveals that HDP_4 performs better than Khetan's formulation (HDP_4 works for both mixed and unmixed).

7. References

1. Wang W, Lian X. Computations of multi-resultant with mechanization. Applied mathematics and computation. 2005; 170(1):237–57.
2. Sylvester JJ. On a theory of the syzygetic relations of two rational integral functions, comprising an application to the theory of Sturm's functions, and that of the greatest algebraical common measure. Philosophical transactions of the Royal Society of London. 1853,143. p.407–548.

3. Zippel R. Effective polynomial computation. Kluwer Academic Publishers, Boston; 1993.
4. Kapur D, Lakshman YN. Elimination Methods: an Introduction. Symbolic and Numerical Computation for Artificial Intelligence B. Donald et. al. Academic Press; 1992.
5. Cayley A. On the theory of elimination. Cambridge and Dublin Mathematical Journal. 1848, 3. p.116–20.
6. Chtcherba AD, Kapur D. Resultants for unmixed bivariate polynomial systems produced using the Dixon formulation. Journal of Symbolic Computation. 2004; 38(2):915–58.
7. Kapur D, Saxena T. Comparison of various multivariate resultant formulations. Proceedings of the 1995 international symposium on Symbolic and algebraic computation; ACM. 1995.
8. Kapur D, Saxena T, Yang L, editors. Algebraic and geometric reasoning using Dixon resultants. Proceedings of the international symposium on Symbolic and algebraic computation; ACM. 1994.
9. Kapur D, Saxena T, editors. Extraneous factors in the Dixon resultant formulation. Proceedings of the 1997 international symposium on Symbolic and algebraic computation; ACM. 1997.
10. Dixon AL. The eliminant of three quantics in two independent variables. Proceedings of the London Mathematical Society. 1909; 2(1):49–69.
11. Weyman J, Zelevinsky A. Determinantal formulas for multigraded resultants. Journal of Algebraic Geometry. 1994; 3(4):569–98.
12. Chionh E-W, Zhang M, Goldman R. Transformation and Transitions from the Sylvester to the Bézout Resultant. Citeseer, 1999.
13. D'Andrea C, Emiris IZ. Hybrid sparse resultant matrices for bivariate polynomials. Journal of Symbolic Computation. 2002; 33(5):587–608.
14. Khetan A. The resultant of an unmixed bivariate system. Journal of Symbolic Computation. 2003; 36(3):425–42.
15. Ahmad SN. Construction and implementation of a hybrid resultant matrix algorithm based on the sylvester-bezout formulation [PhD Thesis]. University of Technology Malaysia 2016.
16. Macaulay F. Some formulae in elimination. Proceedings of the London Mathematical Society. 1902; 1(1):3–27.
17. Karimisangdehi S. New algorithms for optimizing the sizes of dixon and Dixon dialytic matrices: Universiti Teknologi Malaysia, Faculty of Science. 2012.
18. Chionh E-W, Zhang M, Goldman RN, editors. The block structure of three Dixon resultants and their accompanying transformation matrices. Journal of Symbolic Computation – Elsevier. 1999.
19. Chionh E-W, Zhang M, Goldman RN. Fast computation of the Bezout and Dixon resultant matrices. Journal of Symbolic Computation. 2002; 33(1):13–29.
20. Zhang M, Chionh E, Goldman R. Hybrid dixon resultants. The Mathematics of Surfaces. 1998, 8, 193–212.
21. Awange JL, Grafarend EW, Paláncz B, Zaletnyik P. Algebraic geodesy and geoinformatics: Springer Science & Business Media; 2010.
22. Paláncz B, Zaletnyik P, Awange JL, Grafarend EW. Dixon resultant's solution of systems of geodetic polynomial equations. Journal of Geodesy. 2008; 82(8):505–11.
23. Chtcherba AD, Kapur D. Conditions for exact resultants using the Dixon formulation. Proceedings of the 2000 international symposium on Symbolic and algebraic computation. ACM. 2000.
24. Emiris IZ, Kalinka T, Konaxis C. Geometric operations using sparse interpolation matrices. Graphical Models. 2015, 82:99–109.
25. Li W, Yuan C-M, Gao X-S. Sparse difference resultant. Journal of Symbolic Computation. 2015; 68:169–203.