

Efficiently Evaluating U-Resultants

Manfred Minimair*

`manfred@minimair.org`

`http://minimair.org`

Department of Mathematics and Computer Science
Seton Hall University, 400 South Orange Avenue
South Orange, New Jersey 07079, USA

December 1, 2005

Abstract

The objective is to efficiently evaluate u-resultants for numerical u-values (such as over a finite field). The u-resultant of n homogeneous polynomials in $n + 1$ variables is defined to be the multi-variable resultant of these n polynomials and a general linear form in the same variables whose coefficients are represented by the symbols u_0, \dots, u_n . It is shown that the u-resultant can be extracted from a matrix that is smaller than the standard Macaulay matrix obtained from the definition of the u-resultant. The ratio of the sizes of the standard Macaulay matrix and of the matrix introduced by the current paper approximately equals the average of the total degrees of the homogeneous polynomials. As expected, experimental timings show a substantial speed-up when using the smaller matrix.

*Supported by the NSF grant CCF 0430741

1 Introduction

Resultants are fundamental in solving systems of polynomial equations and therefore have been extensively studied [13, 26, 6, 4, 7, 17, 19, 27, 8, 20, 37, 25, 12, 1, 22, 38, 15, 14, 11, 5, 39, 3, 34, 22, 28, 19, 16, 29, 8, 21, 18, 31, 30, 32, 35, 10]. The current paper is focused on u-resultants [4, 2, 39, 27, 11, 36, 9, 23, 24, 40] which have been part of resultant theory since its classical development. Computation of u-resultants is being studied because u-resultants allow to extract information about the common roots of systems of polynomials and to compute the roots. We assume that the reader is familiar with the notion of projective (dense, Macaulay) resultant [26, 11] because the u-resultants of the current paper are defined in terms of projective resultants.

The goal of the current paper is to efficiently evaluate u-resultants for numerical u-values (such as over a finite field). The u-resultant of n homogeneous polynomials in $n + 1$ variables is defined to be the projective resultant of these n polynomials and a general linear form in the same variables whose coefficients are represented by the symbols u_0, \dots, u_n . Evaluation of u-resultants is an important operation needed in u-resultant-based algorithms ([27, 11]). In Section 2 the paper shows that the u-resultant can be extracted from a matrix that is smaller than the standard Macaulay matrix obtained from the definition of the u-resultant.

In order to motivate the results of this paper, let us consider the case of the u-resultant of one polynomial h_1 in the variables x_0 and x_1 . The u-resultant of this polynomial is the resultant of h_1 and the linear form $h_0 = u_0 x_0 + u_1 x_1$. It is easy to see that this u-resultant is $h_1(-u_1, u_0)$ which equals $u_0^{d_1}$ times the resultant of $h_1(-\frac{u_1}{u_0} x_1, x_1)$, where d_1 is the total degree of h_1 . The current paper generalizes this observation to the multi-variate case in Theorem 1. This theorem seems to be quite fundamental. But the author did not find it in the literature.

We explain the structure of the paper. Section 2 states the main theorem, Theorem 1. Section 3 shows how the main theorem is used to efficiently evaluate u-resultants. Section 4 proves the main theorem.

2 Main theorem

We start with some required notation. Let h_0, \dots, h_n be homogeneous polynomials of total degrees d_0, \dots, d_n in the variables x_0, \dots, x_n . Then, as usual, $\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n)$ stands for the projective (dense, Macaulay) resultant of the polynomials h_0, \dots, h_n . Moreover, we restrict the polynomial h_0 to be of total degree 1 with independent symbolic coefficients u_i , that is, $h_0 = u_0 x_1 + \dots + u_n x_n$ and $d_0 = 1$. Hence we define the u-resultant of h_1, \dots, h_n to be the resultant $\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n)$.

Now we are ready to state the main theorem.

Theorem 1 (Main Theorem) *Let*

$$\tilde{h}_i = h_i\left(-\frac{u_1}{u_0} x_1 - \dots - \frac{u_n}{u_0} x_n, x_1, \dots, x_n\right)$$

for $i = 1, \dots, n$. Then

$$\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n) = u_0^{d_1 \cdots d_n} \text{Res}_{d_1, \dots, d_n}(\tilde{h}_1, \dots, \tilde{h}_n). \quad (1)$$

Note that \tilde{h}_i in Theorem 1 is obtained from h_i by replacing the variable x_0 with the polynomial $-\frac{u_1}{u_0} x_1 - \dots - \frac{u_n}{u_0} x_n$.

Example 2 Let $n = 2$, $d_1 = 3$ and $d_2 = 4$. Then h_1 and h_2 are homogeneous polynomials in the variables x_0, x_1 and x_2 . Furthermore, $\tilde{h}_i = h_i\left(-\frac{u_1}{u_0} x_1 - \frac{u_2}{u_0} x_2, x_1, x_2\right)$ and, by Theorem 1,

$$\text{Res}_{1, 3, 4}(u_1 x_1 + u_2 x_2 + u_3 x_3, h_1, h_2) = u_0^{12} \text{Res}_{3, 4}(\tilde{h}_1, \tilde{h}_2).$$

Remark 3 The u-resultant of (1) also equals

$$\frac{\text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n)}{u_0^{(n-1)d_1 \cdots d_n}}$$

(see Lemma 9), where

$$\bar{h}_i = h_i(-u_1 x_1 - \dots - u_n x_n, u_0 x_1, \dots, u_0 x_n). \quad (2)$$

This formula avoids fractions inside the resultant operator.

3 Computational use of the main theorem

This section discusses the computational use of Theorem 1 for efficiently evaluating u-resultants.

The next definition introduces some notions from the Macaulay-Canny resultant algorithm which are used subsequently.

Definition 4 (Macaulay matrix) The reader is reminded that the Macaulay-Canny resultant algorithm computes the projective resultant [11] as the quotient of two determinants of two matrices (or their generic perturbations in degenerate cases [2]). The larger matrix is contained in the dividend of this quotient. We call it the *Macaulay matrix*.

Obviously, u-resultants can be evaluated without the use of Theorem 1 simply by evaluating the definition [27].

Alternatively, Theorem 1 allows to evaluate the u-resultant via the right-hand side of (1) if $u_0 \neq 0$. The formula of (1) is designed for cases when the u_i 's come from a field, such as a prime field occurring in computations in homomorphic images, because it requires the evaluation of the inverse u_0^{-1} . (For computations over rings, not-necessarily fields, the equivalent formula from Remark 3 is better suited because it does not introduce fractions.)

Next we consider some practical aspects of Theorem 1. First we study the sizes of the Macaulay matrices derived from (1). Then we give speed-ups for experimental timings.

Sizes of Macaulay matrices

The following corollary compares the sizes of the Macaulay matrices derived from the left-hand and from the right-hand side of (1).

Corollary 5 *The ratio of the size of the Macaulay matrix of the left-hand side of (1) over the size of the Macaulay matrix of the right-hand side of (1) is*

$$\frac{1 + \sum_{i=1}^n d_i}{n}.$$

Proof: The size [26, 11] of the Macaulay matrix corresponding to the resultant $\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n)$ of the left-hand side of (1) is

$$\binom{\sum_{i=1}^{n+1} d_i}{n} = \binom{1 + \sum_{i=1}^n d_i}{n}.$$

Similarly, the size of the Macaulay matrix corresponding to the right-hand side of (1) is

$$\binom{\sum_{i=1}^n d_i}{n-1}.$$

Thus the ratio of the size of the left-hand side and the right-hand side is

$$\frac{(1 + \sum_{i=1}^n d_i)!}{(1 + \sum_{i=1}^n d_i - n)! n!} \bigg/ \frac{(\sum_{i=1}^n d_i)!}{(\sum_{i=1}^n d_i - n + 1)! (n-1)!} = \frac{1 + \sum_{i=1}^n d_i}{n}.$$

□

Speed-ups using Theorem 1

The speed-ups when comparing the running times measured for evaluating u-resultants without and with the use of Theorem 1 are listed in Figure 1. The computations were carried out with Maple using the Macaulay resultant package MR [33]. Random u-resultants over a 32-bit prime field with the u_i 's replaced by random field elements were computed in the unmixed case for $n = 2$. Figure 1 shows substantial speed-ups for polynomials of degrees up to 14. (Only for $d = 1$ there is a speed-up less than 1.) As expected (Corollary 5), the speed-up grows as the degree d grows.

| | | | | | | | |
|------------|------|------|-------|-------|-------|-------|-------|
| Degree d | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Speed-up | 0.5 | 1.5 | 4.0 | 7.0 | 13.3 | 23.4 | 33.0 |
| Degree d | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| Speed-up | 53.8 | 70.2 | 107.1 | 126.2 | 162.2 | 191.3 | 232.9 |

Figure 1: Speed-ups achieved by Theorem 1 in the unmixed case for $n = 2$

Thus we observe that Theorem 1 can be used to evaluate u-resultants with much improved efficiency. The theorem seems particularly promising for computations in homomorphic images, when many u-resultant evaluations for many different prime fields are required, because the speed-ups will naturally add up.

4 Proof of the main theorem

Before we start proving, we fix some necessary notations.

Notation 6 For the proofs of the lemmas we assume that all the polynomials h_i have *independent symbolic coefficients*, not only h_0 . This is an essential assumption to be used subsequently.

Furthermore, let

$$\bar{h}_i = h_i(-u_1 x_1 - \cdots - u_n x_n, u_0 x_1, \dots, u_0 x_n),$$

for $i = 1, \dots, n$, as in (2). The polynomial \bar{h}_i is the polynomial version of the rational expression h_i of Theorem 1.

Now we are ready to state lemmas.

Lemma 7 *For all (specialized) polynomials h_i , the vanishing of the resultant $\text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n)$ implies that either the coefficient u_0 or the resultant $\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n)$ vanishes.*

Proof: Assume that $\text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n) = 0$ and that $u_0 \neq 0$. Therefore we can fix an $(x_1, \dots, x_n) \neq 0$ such that $\bar{h}_1 = \cdots = \bar{h}_n = 0$. Now let $x_0 := \frac{-u_1 x_1 - \cdots - u_n x_n}{u_0}$. Thus, by the homogeneity of the h_i 's, $h_1 = \cdots = h_n = 0$ and $h_0 = u_0 x_0 + \cdots + u_n x_n = 0$. Therefore $\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n) = 0$. \square

Next we derive a skeleton equation, (3), for $\text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n)$.

Lemma 8 *We have*

$$\text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n) = \lambda u_0^\delta (\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n))^\epsilon, \quad (3)$$

for a constant λ and some integers δ and ϵ .

Proof: By Lemma 7, we have that the vanishing of the polynomial $\text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n)$ implies that the polynomial product

$$u_0 \text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n)$$

vanishes. Thus, by Hilbert's Nullstellensatz,

$$(u_0 \text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n))^\gamma = p \text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n), \quad (4)$$

for some positive integer exponent γ and some polynomial p . Notice that the left-hand side of (4) is the irreducible factorization of the right-hand side of (4) because both u_0 and $\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n)$ are irreducible polynomials. Thus the only irreducible factors of $\text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n)$ can be u_0 and $\text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n)$. \square

The next lemma specifies the indeterminate coefficient and exponents of the skeleton equation (3).

Lemma 9 *We have*

$$\text{Res}_{d_1, \dots, d_n}(\bar{h}_1, \dots, \bar{h}_n) = u_0^{(n-1)d_1 \cdots d_n} \text{Res}_{d_0, \dots, d_n}(h_0, \dots, h_n).$$

Proof: First we determine λ in (3) by specializing h_i to $x_i^{k_i}$. For this specialization, (3) becomes $1 = \lambda$.

Next we determine ϵ in (3) by comparing the total degrees in the coefficients of h_1 in the left-hand and right-hand side of (3). Notice that the total degree of the left-hand side is $k_2 \cdots k_n$ and of the right-hand side is $k_1 k_2 \cdots k_n = 1 \cdot k_2 \cdots k_n \cdot \epsilon$. Thus $\epsilon = 1$.

Finally, we determine δ in (3) by comparing the total degrees in the coefficients of h_0 . Notice that by the homogeneity of the h_i 's the total degree of the left-hand side is $n d_1 \cdots d_n$ and of the right-hand side is $\delta + d_1 \cdots d_n$. Thus $\delta = (n - 1) d_1 \cdots d_n$. \square

Now we are ready to prove Theorem 1.

Proof of Theorem 1: Theorem 1 follows by specializing the polynomial coefficients of the h_i 's in Lemma 9 and by taking into account the multi-homogeneity of the resultant and the homogeneity degree d_i of the h_i 's. \square

5 Conclusion and future directions

We have shown that one can evaluate the u-resultant efficiently by using a matrix that is smaller than the standard Macaulay matrix. Future directions may include generalizing the techniques of the current paper to the toric version [39] of the u-resultant.

References

- [1] L. Busé, M. Elkadi, and B. Mourrain. Generalized resultants over unirational algebraic varieties. *J. Symbolic Computation*, 29(4-5):515–526, 2000.
- [2] J. Canny. Generalised characteristic polynomials. *J. Symbolic Computation*, 9:241–250, 1990.
- [3] J. Canny and I. Emiris. A subdivision-based algorithm for the sparse resultant. *J. ACM*, 47(3):417–451, May 2000.
- [4] J. Canny, E. Kaltofen, and Lakshman Y. Solving systems of non-linear polynomial equations faster. In *Proc. ACM-SIGSAM 1989 Internat. Symp. Symbolic Algebraic Comput. ISSAC’89*, pages 121–128. ACM, 1989.
- [5] E. Cattani, A. Dickenstein, and B. Sturmfels. Residues and resultants. *J. Math. Sci. Univ. Tokyo*, 5(1):119–148, 1998.
- [6] A. Cayley. On the theory of elimination. *Cambridge and Dublin Math. J.*, 3:116–120, 1848.
- [7] M. Chardin. *Contributions à l’algèbre commutative effective et à la théorie de l’élimination*. PhD thesis, Université Paris VI, 1990.
- [8] C. C. Cheng, J. H. McKay, and S. S. Wang. A chain rule for multi-variable resultants. *Proceedings of the American Mathematical Society*, 123(4):1037–1047, April 1995.
- [9] E.-M. Chionh and R. N. Goldman. Degree, multiplicity, and inversion formulas for rational surfaces using u -resultants. *Comput. Aided Geom. Design*, 9(2):93–108, 1992.
- [10] A. Chtcherba, D. Kapur, and M. Minimair. Cayley-Dixon resultant matrices of multi-univariate composed polynomials. In V. Ganzha and E. Mayr, editors, *Computer Algebra in Scientific Computing*, volume 3718 of *Lecture Notes in Computer Science*, pages 125–137, Berlin Heidelberg, 2005. Springer Verlag. 8th International Workshop, CASC 2005, Kalamata, Greece, September 2005, Proceedings.

- [11] D. Cox, J. Little, and D. O’Shea. *Using Algebraic Geometry*. Springer Verlag, New York, Berlin, Heidelberg, 2nd edition, 2004.
- [12] C. D’Andrea and A. Dickenstein. Explicit formulas for the multivariate resultant. *Pure Appl. Algebra*, 164(1-2):59–86, 2001.
- [13] A.-L. Dixon. The eliminant of three quantics in two independent variables. *Proc. London Math. Soc.*, 7:49–69, 473–492, November 1908.
- [14] I. Z. Emiris and V. Pan. The structure of sparse resultant matrices. In *Proc. Int. Symp. on Symbolic and Algebraic Computation (ISSAC)*. ACM Press, 1997.
- [15] I. M. Gelfand, M. M. Kapranov, and A. V. Zelevinsky. *Discriminants, Resultants and Multidimensional Determinants*. Birkhäuser, Boston, 1994.
- [16] L. González-Vega. Determinantal formulae for the solution set of zero-dimensional ideals. *J. Pure Appl. Algebra*, 76(1):57–80, 1991.
- [17] L. González-Vega. Une théorie des sous-résultants pour les polynômes en plusieurs variables. *C. R. Acad. Sci. Paris Sér. I Math.*, 313/13:905–908, 1991.
- [18] H. Hong and M. Minimair. Sparse resultant of composed polynomials I. *J. Symbolic Computation*, 33:447–465, 2002.
- [19] J. P. Jouanolou. Le formalisme du résultant. *Adv. Math.*, 90(2):117–263, 1991.
- [20] D. Kapur and T. Saxena. Sparsity considerations in Dixon resultants. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing (Philadelphia, PA, 1996)*, pages 184–191, New York, 1996. ACM.
- [21] D. Kapur and T. Saxena. Extraneous factors in the Dixon resultant formulation. In *Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation (Kihei, HI)*, pages 141–148, New York, 1997. ACM.
- [22] A. Khetan. The resultant of an unmixed bivariate system. *J. Symbolic Comput.*, 36(3-4):425–442, 2003.

- [23] H. Kobayashi, T. Fujise, and A. Furukawa. Solving systems of algebraic equations by a general elimination method. *J. Symbolic Comput.*, 5(3):303–320, 1988.
- [24] D. Lazard. Résolution des systèmes d'équations algébriques. *Theoret. Comput. Sci.*, 15(1):77–110, 1981.
- [25] R. Lewis and P. Stiller. Solving the recognition problem for six lines using the Dixon resultant. *Math. Comput. Simulation*, 49(3):205–219, 1999.
- [26] F. S. Macaulay. *The algebraic theory of modular systems*. Cambridge Mathematical Library, 1916.
- [27] D. Manocha and J. Canny. Multipolynomial resultant algorithms. *J. Symbolic Computation*, 15(2):99–122, 1993.
- [28] J. H. McKay and S. S. Wang. A chain rule for the resultant of two polynomials. *Arch. Math.*, 53(4):347–351, 1989.
- [29] J. H. McKay and S. S. Wang. A chain rule for the resultant of two homogeneous polynomials. *Arch. Math.*, 56(4):352–361, 1991.
- [30] M. Minimair. Sparse resultant of composed polynomials II. *J. Symbolic Computation*, 33:467–478, 2002.
- [31] M. Minimair. Dense resultant of composed polynomials. *J. Symbolic Computation*, 36(6):825–834, December 2003.
- [32] M. Minimair. Factoring resultants of linearly combined polynomials. In J. R. Sendra, editor, *Proceedings of the 2003 International Symposium on Symbolic and Algebraic Computation*, pages 207–214, New York, NY, 2003. ACM. ISSAC 2003, Philadelphia, PA, USA, August 3-6, 2003.
- [33] M. Minimair. MR. <http://minimair.org/MR.mpl>, April 2003. Macaulay resultant package for Maple.
- [34] M. Minimair. Sparse resultant under vanishing coefficients. *J. Algebraic Combinatorics*, 18(1):53–73, July 2003.
- [35] M. Minimair. Resultants of partially composed polynomials. *J. Symbolic Computation*, 2005.

- [36] H. Murao, H. Kobayashi, and T. Fujise. On factorizing the symbolic U -resultant—application of the ddet operator. *J. Symbolic Comput.*, 15(2):123–142, 1993.
- [37] G. Nakos and R. M. Williams. Elimination with the Dixon resultant. *Mathematica for Education and Research*, 6/3:11–21, 1997.
- [38] P. Pedersen and B. Sturmfels. Product formulas for resultants and Chow forms. *Mathematische Zeitschrift*, 214:377–396, 1993.
- [39] J. M. Rojas. Solving degenerate sparse polynomial systems faster. *J. Symbolic Computation*, 28(1 and 2):155–186, July/August 1999. Special Issue Polynomial Elimination – Algorithms and Applications.
- [40] B. van der Waerden. *Modern Algebra*, volume II. Frederick Ungar Publishing Co., New York, 2nd edition, 1953.