Elimination ideal and bivariate resultant over finite fields

Gilles Villard

Elimination ideal and bivariate resultant over finite fields

Gilles Villard

CNRS, U. Lyon, Inria, ENS de Lyon, UCBL, Laboratoire LIP UMR5668, France

Abstract. A new algorithm is presented for computing the largest degree invariant factor of the Sylvester matrix (with respect either to x or y) associated to two polynomials a and b in $\mathbb{F}_q[x, y]$ which have no non-trivial common divisors. The algorithm is randomized of the Monte Carlo type and requires $O((de)^{1+\epsilon}\log(q)^{1+o(1)})$ bit operations, where d an e respectively bound the input degrees in x and in y. It follows that the same complexity estimate is valid for computing: a generator of the elimination ideal $\langle a, b\rangle \cap \mathbb{F}_q[x]$ (or $\mathbb{F}_q[y]$), as soon as the polynomial system $a = b = 0$ has not roots at infinity; the resultant of a and b when they are sufficiently generic, especially so that the Sylvester matrix has a unique non-trivial invariant factor. Our approach is to use the reduction of the problem to a problem of minimal polynomial in the quotient algebra $\mathbb{F}_q[x, y]/\langle a, b\rangle$. By proposing a new method based on structured polynomial matrix division for computing with the elements in the quotient, we manage to improve the best known complexity bounds.

1 Introduction

Given two polynomials $a, b \in \mathbb{K}[x, y]$, where $\mathbb{K}$ is a commutative field, their resultant $\mathrm{Res}_y(a, b)$ with respect to y is the determinant of the associated Sylvester matrix S_y over $\mathbb{K}[x]$ [22, Ch. 6]. Computing this determinant in quasi-linear time with respect to the input/output size is still beyond our reach in the general case.

In this paper we consider the relaxed problem which is to compute the last (of largest degree) invariant factor of S_y, in the case of a finite field $\mathbb{K} = \mathbb{F}_q$ with q elements. We consider a and b of x-degree at most d and y-degree at most e in $\mathbb{F}_q[x, y]$, having no non-trivial common divisors. For any $\epsilon > 0$, there exist a randomized Monte Carlo algorithm which solves the problem using a quasi-linear number of $O((de)^{1+\epsilon}\log(q)^{1+o(1)})$ bit operations.

The last invariant factor of S_y is a specific divisor of the resultant. If the polynomial system $a = b = 0$ has no roots at infinity with respect to y (the y-leading coefficients of a and b are coprime), then it gives central informations on the affine solutions. It is indeed a generator of the elimination ideal $\langle a, b\rangle \cap \mathbb{K}[x]$ [16, Ch. 2]. We also have, in particular, the fact that this invariant factor gives the resultant when a and b are sufficiently generic (Section 7). (Genericity is considered in the Zariski sense: a property is generic if it holds except on a hypersurface of the parameter space.)

Our approach over finite fields is inspired by and goes further than the major steps taken with: the change of order algorithm of Poteaux and Schost for triangular sets and radical ideals [53]; the algorithm of van der Hoeven and Lecerf, which computes the resultant of generic polynomials with respect to the total degree [28]. In the bivariate case, both these works provides solutions in quasi-linear expected time in the input/output size for the first time ([53] treats general multivariate cases). They are part of the same long line of research which reduces elimination problems to linear algebra [42; 16, Sec. 2.4 & 3.6], and especially to the computation of minimal polynomials in quotient algebras [44, 59]. It is this path that we are pursuing.

The role of minimal polynomials. Let $I = \langle a, b\rangle$ be the (zero-dimensional) ideal generated by a and b in $\mathbb{K}[x, y]$, and $\mathbb{A} = \mathbb{K}[x, y]/I$ be the associated quotient algebra. We remind in Section 2 that the last invariant invariant factor of the Sylvester matrix S_y can be computed as the minimal polynomial μ of the multiplication by x in $\mathbb{A}$, under the condition of absence of roots at infinity [43]. This is how we proceed. The condition on the behaviour at infinity is met for slightly modified polynomials not preventing us from computing the target invariant factor (Section 5).

For efficiency, the minimal polynomial problem is itself reduced to a power projection problem [35, Sec. 6] (a more complete list of references is given later in this introduction). Given a linear form ℓ in the dual of $\mathbb{A}$ over $\mathbb{K}$, the minimal polynomial in $\mathbb{A}$ is computed as the one of the linearly generated sequence $\{\ell(x^i \bmod I)\}_{i\geq 0}$ over $\mathbb{K}$. The application of a random linear form preserves the recursion which is sought in $\mathbb{A}$ [64] (Section 6). As observed by Shoup [57], the power projection problem is dual to the modular composition problem [11]. We finaly rely on Kedlaya

and Umans' approach to address those two latter issues [40] in quasi-linear time over finite fields. As we will now see, this is made possible by a new algorithm we propose for arithmetic operations modulo the ideal.

First result. One of the bottlenecks in above strategy is to perform arithmetic operations in $\mathbb{A}$ [25], even if only to compute the multiplication of two polynomials or the powers of x that need to be projected modulo the ideal I. This is where a main aspect of our contribution lies. In [53], the special case of triangular sets is considered. That is, in our context, when either a or b is univariate. On the other hand, the generic resultant algorithm of [53] relies on Gröbner bases techniques, and the normal form algorithm modulo I of [26].

We instead use polynomial matrix division [32, Sec. 6.3]. Viewing a polynomial f in $\mathbb{K}[x, y]$ as a vector with entries in $\mathbb{K}[x]$, we reduce its x-degree using division by the polynomial Sylvester matrix S_y; let us also specify that we may need to construct a Sylvester matrix from multiples of a and b if the dimensions do not match (Section 3). By definition of the Sylvester matrix, the remainder of this division gives a new polynomial in the coset $f + I$. By means of a similar division after the swich of the roles of x and y, this leads to a normal form algorithm modulo I, up to a regularity assumption related to roots at infinity (Lemma 2.3 and Proposition 3.1). This algorithm is algebraic and deterministic for arbitrary fields. If f has x-degree at most δ and y-degree at most η, then it uses $\tilde{O}((d + \delta)(e + \eta))$ arithmetic operations (Proposition 3.1). A Sylvester matrix is a Toeplitz-like matrix [6]. Our cost bound is based on fast structured matrix arithmetic which is discussed in Section 3.1. In particular, the normal form algorithm allows multiplication in $\mathbb{K}[x, y]/\langle a, b\rangle$ using $\tilde{O}(de)$ operations when the leading coefficients of a and b are sufficiently generic (Lemma 2.3). In the case of the total degree, for generic polynomials a, b with $\deg a \geq \deg b$, the algorithm of [26] costs $\tilde{O}((\deg a)(\deg b))$, after the precomputation of a concise Gröbner basis representation of the ideal using $\tilde{O}((\deg a)^2)$ operations. So in terms of their assumptions the two algorithms are complementary (Section 3.3).

Extension of Kedlaya and Umans' techniques for the power projections. As soon as the normal form algorithm is available, hence the arithmetic operations in $\mathbb{A}$, it is possible to develop the general strategy of Shoup [57, 59] for the computation of modular power projections, coupled by duality with the algorithm of Kedlaya and Umans for modular composition [40] (in this latter reference, the case of a univariate ideal I in $\mathbb{F}_q[x]$ is treated). This is what has been generalized in both [53] and [28], with respective shapes of the ideal I that we have seen above. We proceed in the same way, and integrate the new division algorithm into this overall process: (i) reduction of $f \in \mathbb{F}_q[x]$ modulo I, which is considered as modular composition according to $f(g(x)) \bmod I$ with $g = x$; modular composition relies on multivariate multipoint evaluation following [40, Thm. 3.1]; (ii) using the transposition principle [12, Thm. 13.20], the power projections are obtained (Section 3.4). Since the degree of the resultant of a and b with respect to y is at most $2de$, it is sufficient to be able to compute (i) f modulo I for $\deg f < 4de$ and (ii) $\{\ell(x^i \bmod I)\}_{0 \leq i < 4de}$, in order to deduce the minimal polynomial of the sequence (which is a divisor of the resultant). We establish in Section 4 that (i) and (ii) can be perfomed within our target cost bound over a finite field using bit operations.

Last invariant factor and elimination ideal $\langle a, b\rangle \cap \mathbb{F}_q[x]$. In the presence of roots at infinity, we use a random transformation of a and b into two other polynomials which meet the condition of regularity for computing normal forms, and still make it possible to obtain the initial last invariant factor. This is presented in Section 5. The general complexity bound for the computation of the last invariant factor is given in Section 6 from that of modular power projection. As a consequence of Lazard's structure theorems for bivariate ideals [43], the latter polynomial is a multiple of the minimal polynomial μ of the multiplication by x in $\mathbb{A}$, and both polynomials coincide if the system $a = b = 0$ has no roots at infinity (Lemma 2.1). Under this condition, what we have done so far allows in Section 7 to compute μ, that is a generator of the elimination ideal $I \cap \mathbb{F}_q[x]$.

Comparison to previous work. Given an arbitrary field, the bivariate resultant can be computed using $O(de^2)$ arithmetic operations [22, Chap. 11].

Over a finite field, the approach of [28] allows quasi-linear bit cost; for generic polynomials with respect to the total degree, and any $\epsilon > 0$, this leads to the complexity bound $O(((\deg a)(\deg b)\log q)^{1+\epsilon}) + \tilde{O}((\deg a)^2 \log q)$ when $\deg a \geq \deg b$. (The soft-O notation $\tilde{O}(c)$ captures an additional logarithmic factor $O(c \log^k c)$ for a positive k.) Our algorithm covers this case, in particular. Genericity ensures that there are no roots at infinity and a unique invariant factor (see Section 7), and we obtain a comparable asymptotic bound. Considering degree conditions on the variables individually we treat a larger class of problems and with weaker assumptions. For polynomials of x-degree d and y-degree e, we compute the resultant in quasi-linear time when the Sylvester matrix S_y has a unique non-trivial invariant factor.

Now, in a general way, in all cases as soon as there are no roots at infinity, our approach allows to compute a generator of the elimination ideal $I \cap \mathbb{F}_q[x]$. This is treated in Section 7. We are not aware of any previous method whose cost would be quasi-linear over finite fields under the same assumptions. The complexity of this problem is indeed related to that of the resultant and bivariate lexicographic Gröbner bases [17]. In particular, for a and b of total degree at most d, we arrive at the bound $O(d^{2+\epsilon}\log(q)^{1+o(1)})$, while previous estimates are $\tilde{O}(d^3 \log q)$ [45].

We use bit complexity. The bivariate resultant problem using an algebraic model of computation is a harder problem. To our knowledge, a quasi-linear complexity bound is not achievable at this time. We may refer to [46; 63, 52] and to the pointers found there. It it also important to note that quasi-linear time algorithms are given for bivariate polynomial systems with integer coefficients in [47].

Minimal polynomials and power projections. We give here some additional references from which the results we use largely inherit. The adaptation of numerical matrix methods to the finite field setting has started with the solution of sparse linear systems in mind [13, 64]. These methods result in projections of the powers of the involved matrix for computing its minimal polynomial, as evidenced by Wiedemann's approach [64]. The link is to be made with the use of power projections for computing minimal polynomials in quotient algebras, using the trace map in [61, 54] and general projections in [57, 39, 59]. (We have indeed a multiplication endomorphism in the quotient.)

The duality between the power projection problem and the modular composition one is observed in [57].

In the context of polynomial system solving, for which the literature is vast, we may refer to the use of the trace map in [1, 55, 24], or of arbitrary linear forms in [10]. Structured matrices and duality are applied to multivariate polynomial problems in [48]. Multivariate powers projections are considered in [59, 35], especially for minimal polynomials, and are exploited for the computation of special resultants in [8], and to the change of order of variables for triangular sets in [51]. The link between the change of ordering and linear algebra is also beneficial using power projections of a multiplication matrix in [19, 18], and particularly in order to take advantage of sparsity [20], which brings us back to Wiedemann's algorithm.

Following [43], the Sylvester matrix S_y (or S_x) is a polynomial matrix that we manipulate as such. This may be seen as working in a $\mathbb{K}[y]$-module rather than in a $\mathbb{K}$-vector space in order to represent the quotient algebra $\mathbb{A}$ [31, Sec. 3.10], and implement the operations on its elements. A similar direction has been taken in [4] for a change or ordering of Gröbner bases algorithm.

In linear algebra with implicitly represented matrices, an open problem is to compute the characteristic polynomial in essentially the same time as for the minimal polynomial [Sec. 3][35]. This applies in particular to sparse or structured matrices. The question of computing the bivariate resultant in essentially the same time as for the last invariant factor of the Sylvester matrix appears to be similar to Kaltofen's open problem.

Model of computation. The normal form algorithm for polynomials in $\mathbb{K}[x, y]$ modulo $\langle a, b\rangle$ and its transpose are presented using an algebraic model (Section 3), and work e.g. with computation trees [12, Sec. 4.4]. Complexity bounds correspond to numbers of arithmetic operations performed in $\mathbb{K}$.

The application of Kedlaya and Uman's techniques in Section 4 and threfore the last invariant factor computation in Section 6 rely on a RAM bit complexity model. We consider that arithmetic operations in $\mathbb{F}_q$ can be done in time $\tilde{O}(\log q)$, and that the RAM can produce a random element uniformly distributed in $\mathbb{F}_q$ with the same cost.

Notations. Throughout the paper we consider two polynomials $a, b \in \mathbb{K}[x, y]$, of degrees d_a and d_b in x, and e_a and e_b in y, respectively. We will use the notations $d = \max\{d_a, d_b\}$ and $e = \max\{e_a, e_b\}$. The associated Sylvester matrices with respect to x and y are $S_x \in \mathbb{K}[y]^{n_x \times n_x}$ and $S_y \in \mathbb{K}[x]^{n_y \times n_y}$, with dimensions $n_x = d_a + d_b$ and $n_y = e_a + e_b$. The resultants $\mathrm{Res}_x(a, b) \in \mathbb{K}[y]$ and $\mathrm{Res}_y(a, b) \in \mathbb{K}[x]$, of a and b with respect to x and y, are the respective determinants of S_x and S_y [22, Chap. 6]. We assume that a and b have no non-trivial common divisors, hence both S_x and S_y are non-singular. We focus on computations in relation to $\mathrm{Res}_y(a, b) = \det S_y$ (the conclusions would be unchanged in relation to $\mathrm{Res}_x(a, b)$).

We use expressions such as "x-degree" or "y-leading coefficient" to indicate the variable which is concerned, and use $\deg_x$ and $\deg_y$ in formulas when bivariate polynomials are involved. Subscripts for example in $\mathbb{K}[x, y]_{<(d,n_y)}$ indicate degree bounds in x and y, and $\mathbb{K}[x]_d^n$ is the set of polynomials of degree d.

We are often led to manipulate reversals of polynomials. For $k \geq 0$, we define the reversal of a polynomial $f \in \mathbb{K}[x]$ with respect to k as $\mathrm{rev}_k(f) = x^k f(1/x)$; by default, if k is not specified, the reversal is taken with respect to the degree of the polynomial. This is generalized to polynomial matrices viewed as matrix polynomials, we mean with matrix coefficients.

The polynomials in $\mathbb{K}[x, y]$ are identified with the (column) vectors of their coefficients, using dimensions which will be clear from the context. For example, given $f = f_0(x) + f_1(x)y + \ldots f_d(x)y^d$ and $n \geq d+1$, $v_y(f) \in \mathbb{K}[x]^n$ denotes the vector $[0 \ \ldots \ 0 \ f_d \ \ldots \ f_0]^\mathsf{T}$.

2 Polynomial matrices, resultant and bivariate ideals

We give the basic notions and results we need in the rest of the text concerning the relations between the resultant of two polynomials and the ideal they generate. As univariate polynomial matrix, the Sylvester matrix S_y is unimodularly equivalent to a matrix $\mathrm{diag}(s_1, \ldots, s_n) \in \mathbb{K}[x]^{n_y \times n_y}$ in Smith normal form, where s_n is the invariant factor of largest degree. We are not able to always compute the resultant within the cost target. We are, however, able to compute the last invariant factor (Corollary 6.1).

Using the structure theory of finitely generated modules, this last invariant factor can be seen as the minimal polynomial of a linear transformation in a finite dimensional $\mathbb{K}$-vector space [31, Sec. 3.10]. Such a formalism has been exploited occasionally for the efficient computation of general matrix normal forms [62, 60]. Concerning Sylvester matrices and in the broader context of polynomial system solution, this is related to the use of a multiplication map on a quotient algebra [42].

Let $I = \langle a, b \rangle$ be the (zero-dimensional) ideal generated by a and b in $\mathbb{K}[x, y]$, and $\mathbb{A} = \mathbb{K}[x, y]/I$ be the associated quotient algebra. We especially rely on the following results, which are immediate consequences of Lazard's theorem [43].

Lemma 2.1 ([43, Thm. 4]). *The last invariant factor of S_y is a multiple of the minimal polynomial of the multiplication by x in $\mathbb{A}$, both polynomials coincide if the y-leading coefficients of a and b are coprime in $\mathbb{K}[x]$.*

Proof. The last invariant factor is a multiple of the last diagonal entry $h \in \mathbb{K}[x]$ of the Hermite form, where the latter is lower triangular and obtained by unimodular column transformations. The polynomial h is in $I \cap \mathbb{K}[x]$ (combinations of columns of S_y are seen as combinations of a and b), which gives the first assertion. When the leading coefficients are coprime, the divisibility property (i) in [43, Thm. 4] shows that the Hermite form of S_y can be brought to Smith form using unimodular (row) transformations, without modifying the diagonal. From (ii) in [43, Thm. 4], the last invariant factor is therefore an element of a reduced Gröbner basis of I, and as polynomial in $\mathbb{K}[x]$ it generates the elimination ideal $I \cap \mathbb{K}[x]$. □

The condition on the leading coefficients of a and b in Lemma 2.1 is the fact that the system $a = b = 0$ has no roots at infinity with respect to y. In general, the resultant and the last invariant factor may have terms coming from both the affine variety and the behaviour at infinity [16, Chap. 3]. To still be able to reduce the invariant factor computation to a minimal polynomial problem the assumption of Lemma 2.1 will hold after a random modification of the input polynomials (see Section 5).

The resultant can be deduced from Lemma 2.1 in particular when the Smith form of S_y has a unique non-trivial invariant factor and there are no roots at infinity. This corresponds to certain situations in which the ideal I has a shape basis [23, 2].

Lemma 2.2 ([43, Thm. 4]). *The y-leading coefficients of a and b are coprime in $\mathbb{K}[x]$ and there exist two polynomials $\mu, \lambda \in \mathbb{K}[x]$ such that $I = \langle \mu(x), y - \lambda(x) \rangle$ if and only if, up to a non-zero element in $\mathbb{K}$, the resultant $res_y(a, b)$ is the minimal polynomial μ of the multiplication by x in $\mathbb{A}$.*

Proof. From [43, Thm. 4], under the hypothesis $I = \langle \mu(x), y - \lambda(x) \rangle$ and using the coprimeness, we know that the Hermite form of S_y has a unique non-trivial diagonal entry, which is μ. Therefore, the latter is also the determinant of S_y, up to the normalization to a monic polynomial in the Hermite form.

Conversely, the last element $h \in \mathbb{K}[x]$ of the diagonal of the Hermite form of S_y is in I . Hence h must be a multiple of μ, and of the resultant by assumption. It follows that $h = \mathrm{Res}_y(a, b) = \mu$, and all the other diagonal entries of the Hermite form are equal to 1. This proves that the y-leading coefficients of a and b are coprime since otherwise the first diagonal entry of the Hermite form would be a non-constant polynomial in $\mathbb{K}[x]$. Item (ii) [43, Thm. 4] allows to conclude. □

Concerning the links bewteen the resultant and the associated ideal, the reader may especially refer to [15], where a general multivariate version of Lemma 2.2 is given.

Example 2.1. The Sylvester matrix may have a unique non-trivial invariant factor (that our algorithm will compute) even though there are roots at infinity. With $\mathbb{K} = \mathbb{F}_2$, take $a = (x+1)\,y + x^2$ and $(x+1)\,y^2 + y$. We have $I = \langle x^2, y\rangle$, and the Hermite normal form of S_y is

$$S_y U = \begin{bmatrix} x+1 & 0 & 0 \\ 1 & x+1 & 0 \\ 0 & x^2 & x^2\,(x+1) \end{bmatrix},$$

with U unimodular. None of the arguments used for Lemmas 2.1 and 2.2 apply: the Hermite form cannot be brought to Smith form using unimodular row operations without modifying the diagonal (as used in the proof of Lemma 2.1), and the form is not either trivial (proof of Lemma 2.2). The last invariant factor of S_y is $\mathrm{Res}_y(a,b) = x^2\,(x+1)^3$. □

We now characterize the existence of roots at infinity using column reducedness of polynomial matrices [32, Sec. 6.3, p.384], which is used in next sections. Let S be a matrix in $\mathbb{K}[x]^{n\times n}$ whose column j has degree d_j. We call (column) leading (matrix) coefficient of S the matrix in $\mathbb{K}^{n\times n}$ whose entry (i,j) is the coefficient of degree d_j of the entry (i,j) of S. We manipulate non-singular univariate polynomial matrices, and say that such a matrix is column reduced if its leading coefficient is invertible.

Lemma 2.3. *S_x is column reduced if and only if, the y-leading coefficients of a and b are relatively prime and at least one of latter polynomials in $\mathbb{K}[x]$ has maximal degree d_a or d_b, respectively.*

Proof. Let $s, t \in \mathbb{K}[x]$ be the y-leading coefficients of a, b, with respective degrees d_s and d_t. The columns of the leading coefficient of S_x are given by the vectors in $\mathbb{K}^{d_a+d_b}$ associated to

$$x^{d_b-1}s, x^{d_b-2}s, \ldots, s, x^{d_a-1}t, x^{d_a-2}t, \ldots, t.$$

If S_x is column reduced then the first row of its leading matrix is non-zero and either $d_s = d_a$ or $d_t = d_b$. Let's say that $d_s = d_a$ (up to a column permutation). The leading coefficient of S_x is therefore given by

$$x^{d_b-1}s, x^{d_b-2}s, \ldots, x^{d_t}s, x^{d_t-1}s, x^{d_t-2}s, \ldots, s, x^{d_s-1}t, x^{d_s-2}t, \ldots, t, \tag{1}$$

and we see that its rank is that of the Sylvester matrix associated to s and t since the latter is given by

$$x^{d_t-1}s, x^{d_t-2}s, \ldots, s, x^{d_s-1}t, x^{d_s-2}t, \ldots, t.$$

Conversely, from the independence of the vectors in Eq. (1) we obtain the column reducedness of S_x. □

3 Bivariate polynomial division

In this section we propose a normal form algorithm for bivariate polynomials modulo the ideal $I = \langle a, b\rangle$. The algorithm relies on matrix polynomial division. Bivariate polynomials in $\mathbb{K}[x,y]$ are viewed as univariate polynomial vectors alternately over $\mathbb{K}[x]$ and $\mathbb{K}[y]$, dividing such a vector by S_y or S_x, is indeed quivalent to reducing the associated polynomial modulo the ideal. Sylvester matrices are Toeplitz-like matrices, we first recall in Section 3.1 how operations on matrices in this class can be performed taking into account their structure [6, 50]. We then study the division with remainder of a polynomial vector by S_y or S_x in Section 3.2. In order to be able to define a normal form and perform the division efficiently, we rely on a regularity assumption on leading coefficient matrices: we suppose that S_x and S_y are column reduced. This assumption is ultimately harmless for computing the last invariant factor (Section 5).

In Section 3.3 we present the normal form algorithm. We keep the same notations as before for the degrees of a and b, and the dimensions of the matrices; especially, d is the maximum degree in x and S_y is $n_y \times n_y$. Given a polynomial $f \in \mathbb{K}[x,y]$, we show how to compute a unique polynomial $\hat{f} \in \mathbb{K}[x,y]_{<(d,n_y)}$, that we denote by $\hat{f} = f \text{ rem } I$, such that $f - \hat{f} \in I$ (Proposition 3.1). Uniqueness is ensured using a properness property provided by the polynomial matrix division. The construction is a $\mathbb{K}$-linear map that sends f to $\hat{f}$ whose y-coefficients are given by

the entries of a vector $v_y(\hat{f}) \in \mathbb{K}[x]^{n_y}_{<d}$ such that $S_y^{-1} v_y(\hat{f})$ is strictly proper (tends to zero when x tends to infinity), see Eq. (2). This allows us to represent the elements in $\mathbb{A}$ by normal forms. The transpose algorithm, which computes corresponding power projections, is derived in Section 3.4

With $\deg_x a = d_a$, $\deg_x b = d_b$, $\deg_y a = e_a$, and $\deg_y b = e_b$, the quotient algebra $\mathbb{A}$ has dimension at most $d_a e_b + d_b e_a$. In order to represent its elements, the quotient is embedded in the space $\mathbb{K}[x]^{n_y}_{<d}$ of dimension

$$dn_y = \max\{d_a, d_b\}(e_a + e_b)$$

which can therefore be slightly larger (Example 3.1).

3.1 Structured matrix arithmetic

The normal form algorithm exploits the fact that Sylvester matrices are structured. The class of structure that we are facing is the one of Toeplitz-like polynomial matrices which are commonly handled using the notion of displacement rank [33]. The notion allows to have a concise matrix representation through which matrix arithmetic can be implemented efficiently [6, 50].

Given by the polynomials a and b, S_x and S_y are represented using $O(de)$ elements of $\mathbb{K}$. The division algorithm requires to solve associated linear systems and uses matrix inversion with truncated power series entries. We consider that polynomial Sylvester matrices and their inverses are represented using their concise Toeplitz-like representations [6]. This is obtained for example by extension of the ΣLU form defined over fields [34], to polynomials or truncated power series [52, Sec. 3].

Multiplying an $n \times n$ polynomial Sylvester matrix of degree d by a polynomial vector of degree at most l over $\mathbb{K}[x]$, can be done using $\tilde{O}(n(d+l))$ arithmetic operations in $\mathbb{K}$ [6]. This cost bound is valid for the same type of multiplication using instead the inverse of the matrix modulo x^l when it exists. If $T \in \mathbb{K}^{n \times n}$ is a non-singular Sylvester matrix and $v \in \mathbb{K}^n$, then the linear system $T^{-1}v$ can be solved using $\tilde{O}(n)$ arithmetic operations. This is obtained by combining an inversion formula for the Sylvester matrix [41], and matrix Padé approximation [3] (see also [6, Chap. 2, Sec. 9] and [63, Sec. 5]). The declination of this is applied in Section 3.2 over truncated power series modulo x^l. Let $S \in \mathbb{K}[x]^{n \times n}_d$ be a polynomial Sylvester matrix such that $\det S(0) \neq 0$, and consider a vector $v \in \mathbb{K}[x]^n$ of degree at most l. The system $S^{-1}v$ can be solved modulo x^l using $\tilde{O}(n(d+l))$ arithmetic operations. From [63, Prop. 5.1], the matrix inverse modulo x^l can itself be computed (with concise representation) within the same cost bound.

3.2 Matrix and bivariate polynomial division

Consider S in $\mathbb{K}[x]^{n \times n}$, non-singular of degree d. For any vector $v \in \mathbb{K}[x]^n$, we know from [32, Thm. 6.3-15, p. 389] that there exist unique $w, \hat{v} \in \mathbb{K}[x]^n$ such that

$$v = Sw + \hat{v}, \tag{2}$$

and $S^{-1}\hat{v}$ is strictly proper. From [32, Thm. 6.3-10, p. 383] we further have that the polynomial remainder vector $\hat{v}$ has degree less d; note however that uniqueness is ensured by properness and not by the latter degree property (Example 3.1).

The following will be applied to both S_x and S_y, hence we take a general notation S for the statement. We propose a structured matrix polynomial adaptation of the Cook-Sieveking-Kung algorithm for (scalar) polynomial division with remainder, about which the reader may refer to [22, Sec. 9.1].

Lemma 3.1. *Let $S \in \mathbb{K}[x]^{n \times n}$ be a Sylvester matrix of degree d, and assume that S is column reduced. Consider a vector $v \in \mathbb{K}[x]^n$ of degree at most l. The unique remainder $\hat{v}$ of the division of v par S as in Eq. (2) can be computed using $\tilde{O}(n(d+l))$ arithmetic operations in $\mathbb{K}$.*

Proof. Consider that S is associated to two polynomials $a, b \in K[x, y]$ as previously, such that $S = S_y$ and we have e_1 columns of degree d_1 and e_2 columns of degree d_2. Up to row and column permutations we assume that $d = \max\{d_1, d_2\} = d_1$.

We first treat the case $d = d_1 = d_2$. All the columns of S have the same degree, hence since S is column reduced it is also row reduced (use the definition given before Lemma 2.3, on the rows).

If $l < d$, then we take $\hat{v} = v$. From [32, Thm. 6.3-11, p. 385], by row reducedness, we know that $S^{-1}\hat{v}$ is strictly proper. If $l \geq d$, the polynomial division can be perfomed by reformulating [22, Sec. 9.1, Eq. (2)] on matrices. Since S has non-singular leading matrix, by the predictable degree property [32, Thm. 6.3-13, p. 387] we know that the quotient vector w has degree $\deg v - d$, hence at most $l - d$. Using reversals of matrix polynomials, Eq. (2) can be rewritten as

$$\mathrm{rev}_l(v) = \mathrm{rev}_d(S) \cdot \mathrm{rev}_{l-d}(w) + x^{l-d+1}\mathrm{rev}_{d-1}(\hat{v}),$$

hence we have

$$\mathrm{rev}_{l-d}(w) \equiv \mathrm{rev}_d(S)^{-1}\mathrm{rev}_l(v) \bmod x^{l-d+1}. \tag{3}$$

Remark that by reducedness assumption the coefficient matrix of degree 0 of $\mathrm{rev}_d(S)$ is non-singular, thus the latter matrix in invertible modulo x^{l-d+1}. As soon as $w' = \mathrm{rev}_{l-d}(w)$ hence $w = \mathrm{rev}_{l-d}(w')$ are known, then $\hat{v}$ can be deduced using $\hat{v} = v - Sw$. We know that $S^{-1}\hat{v}$ is strictly proper using reducedness, as done previously. Using fast structured matrix arithmetic (Section 3.1), $\mathrm{rev}_{l-d}(w)$ is computed from Eq. (3) and $\hat{v}$ is obtained within the claim cost bound.

When $d = d_1 > d_2$, first we balance the columns degrees. With $\delta = d_1 - d_2 > 0$, take $D = \mathrm{diag}(x^\delta, \ldots, x^\delta, 1, \ldots, 1)$, with e_1 entries x^δ. The matrix $T = SD^{-1}$ has all its column degrees equal to d_2. Here and below the degree of a rational function is the difference between the degrees of the numerator and the denominator. Column and row reducedness are extended accordingly.

If $l < d_2$ we let $v' = v$, otherwise we can compute a polynomial vector w' of degree at most $l - d_2$ and $v' = v - Tw'$ of degree less than d_2 such that $T^{-1}v'$ is strictly proper. This is done using Eq. (2) after having multiplied everything by x^δ so as to be reduced to a division with polynomial matrices, in time $\tilde{O}(n(d+l))$. This is similar to the $d_1 = d_2$ case above since T is column reduced.

Then, taking the quotient of the first e_1 entries of w' by x^δ, we write $w' = Dw + z$, where z is of degree less than δ and such that only its first e_1 entries may be non-zero. The vector w remains of degree at most $l - d_2$, and we obtain $\hat{v}$ in time $\tilde{O}(n(d+l))$ as

$$\hat{v} = v - Sw = v - SD^{-1}(w' - z) = v' + SD^{-1}z.$$

In order to complete the proof we check that $S^{-1}\hat{v}$ is strictly proper. This vector is $S^{-1}\hat{v} = S^{-1}v' + D^{-1}z = D^{-1}T^{-1}v' + D^{-1}z$. By construction, $T^{-1}v'$ is strictly proper, it is thus the same for $D^{-1}T^{-1}v'$; z has degree at most $\delta - 1$ for its first e_1 entries (the other ones are zero), hence $D^{-1}z$ is strictly proper. □

3.3 Normal form modulo the bivariate ideal

Given a polynomial $f \in \mathbb{K}[x, y]$ whose y-degree is less than the dimension n_y of S_y, we can apply Lemma 3.1 to the vector $v_y(f) \in \mathbb{K}[x]^{n_y}$ of the coefficients of f. Equation (2) becomes

$$v_y(f) = S_y w + v_y(\hat{f})$$

on vectors, and by definition of the Sylvester matrix we have

$$\hat{f} = f - ua - vb \in f + I$$

for some $u, v \in \mathbb{K}[x, y]$, with $\hat{f}$ of x-degree less than d. We show with Proposition 3.1 that, thanks to the uniqueness of the remainder, this allows us to define a normal form modulo $\langle a, b\rangle$. The general y-degree case for f is treated using a preparatory division by S_x (whose entries are in $\mathbb{K}[y]$) in order to reduce the degree in y. The overall construction gives a $\mathbb{K}$-linear map

$$\begin{aligned} \varphi : \mathbb{K}[x, y] &\to \mathbb{K}[x, y]_{<(d,n_y)} \\ f &\mapsto \hat{f} = f \bmod I \end{aligned} \tag{4}$$

such that $f - \varphi(f) \in I$, and $\varphi(g) = 0$ if $g \in I$. The map φ is thus appropriate in order to represent the elements in $\mathbb{A}$ by normal forms.

Example 3.1. With $\mathbb{K} = \mathbb{Q}$, consider $a = x^2y + y$ and $b = xy^2 + x$; we have $d = d_a = 2$ and $n_y = e_a + e_b = 1 + 2 = 3$. If $f = x$ then both f and $f - b = -xy^2$ are in $\mathbb{K}[x, y]_{<(2,3)}$, hence the map φ might not be surjective. The division as in Eq. (2) leads to

$$v_y(f) = \begin{bmatrix} 0 \\ 0 \\ x \end{bmatrix} = S_y w + v_y(\hat{f}) = \begin{bmatrix} x^2+1 & 0 & x \\ 0 & x^2+1 & 0 \\ 0 & 0 & x \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} -x \\ 0 \\ 0 \end{bmatrix},$$

and $\hat{f} = f - b$ since we can check that $S^{-1}v_y(\hat{f})$ is strictly proper, whereas $S^{-1}v_y(f)$ is not. It may be noted that the quotient algebra $\mathbb{K}[x, y]/\langle a, b\rangle$ has dimension 5, which is smaller than the dimension of $\mathbb{K}[x, y]_{<(2,3)}$. □

If $\eta = \deg_y f \geq n_y$ and $\delta = \deg_x f$ is less than the dimension n_x of S_x, then we can directly proceed to the division using S_x. Otherwise, as we now see with Lemma 3.2, we first extend S_x to a bigger appropriate Sylvester matrix T_x of dimension $\delta + 1$. By linearization, we associate to f a vector $v_x(f) \in \mathbb{K}[y]^{\delta+1}$ of y-degree η. Then using division by T_x, whose y-degree is the degree e of S_x, we can compute f' of y-degree less than $e < n_y$, such that $f - f' \in I$.

Lemma 3.2. *Assume that the Sylvester matrix S_x associated to a and b with respect to x is column reduced. Consider $f \in \mathbb{K}[x, y]$ of x-degree at most δ and y-degree at most $\eta \geq n_y$. Using $\tilde{O}((n_x + \delta)\eta)$ arithmetic operations in $\mathbb{K}$ we can compute a polynomial $f' \in \mathbb{K}[x, y]$, of x-degree at most $\max\{n_x - 1, \delta\}$ and y-degree less than $e < n_y$, such that $f - f' \in I$.*

Proof. If δ is less than n_x, we simply take $T_x = S_x$. Otherwise, let $m = \delta - n_x + 1$, and denote the y-leading coefficients of a, b by $s, t \in \mathbb{K}[x]$. Since S_x is column reduced, from Lemma 2.3 we know that $\gcd(s, t) = 1$. Either s or t is not divisible by x, let us assume that it is s, and take for T_x over $\mathbb{K}[y]$, the Sylvester matrix associated to a and $x^m b$ with respect to x. The Sylvester matrix associated to s and $x^m t$ is non-singular, hence T_x is column reduced by Lemma 2.3 again: if either $\deg s = d_a$ or $\deg t = d_b$, then either $\deg s = d_a$ or $\deg t + m = d_b + m$.

This matrix T_x has dimension $\max\{n_x, \delta+1\}$, and degree $e = \max\{e_a, e_b\}$ in y. The remainder of the division of $v_x(f)$ by T_x gives f' such that $f - f' \in I$, its y-degree is less than the one of T_x, and its x-degree is less than the dimension of T_x. The cost bound is from Lemma 3.1, with a matrix of dimension $n = \max\{n_x, \delta + 1\}$ and degree e, and a vector of degree $l = \eta \geq e$. □

Lemma 3.2 allows to first reduce the y-degree, the reduction of the degree in x now also ensures the normal form.

Proposition 3.1. *Assume that the Sylvester matrices S_x and S_y associated to a and b are column reduced, and consider $f \in \mathbb{K}[x, y]$. The $\mathbb{K}$-linear map in Eq. (4) is well defined by choosing for $\hat{f}$ the unique polynomial in $\mathbb{K}[x, y]_{<(d,n_y)}$ such that $f - \hat{f} \in I$, and $S_y^{-1}v_y(\hat{f})$ is strictly proper. If f has x-degree at most δ and y-degree at most η, then this normal form for $f + I$ in $\mathbb{A}$ can be computed using $\tilde{O}((d + \delta)(e + \eta))$ arithmetic operations in $\mathbb{K}$.*

Proof. We show the existence of such an $\hat{f}$ for every f, then show that $\hat{g} = 0$ if $g \in I$. After division by S_x using Lemma 3.2 we have $f' \in \mathbb{K}[x, y]$ of y-degree less than $e < n_y = e_a + e_b$ such that $f - f' \in I$. Then by Lemma 3.1, that is by division by S_y, we obtain $\hat{f} \in \mathbb{K}[x, y]_{<(d,n_y)}$ such that $f' - \hat{f} \in I$, hence $f - \hat{f} \in I$. By construction, $S_y^{-1}v_y(\hat{f})$ is strictly proper. For $g \in I$, this first leads to some g' of y-degree less than $e < n_y$. Since S_y is column reduced, we know from Lemma 2.3 that the y-leading coefficients of a and b are relatively prime, hence using [43, Lem. 7] there exist polynomials $r, s \in \mathbb{K}[x, y]$ such that

$$g' - ra - sb = 0, \ \deg_y r < e_b, \ \text{and} \ \deg_y s < e_a.$$

By uniqueness it follows that we must have $\hat{g} = 0$ because this value is appropriate using above identity.

The map $\varphi(f) = \hat{f}$ is well defined and provides a normal form. For f_1, f_2 in the coset $f + I$ we indeed have $\varphi(f_1 - f_2) = 0$ hence $\varphi(f_1) = \varphi(f_2)$ by $\mathbb{K}$-linearity of the divisions.

From Lemma 3.2, the first division by S_x costs $\tilde{O}((d + \delta)(e + \eta))$, where we use that S_x has dimension $n_x \leq 2d$ and degree e. This leads to the next division of a vector of degree at most $\max\{n_x - 1, \delta\}$ by S_y, whose dimension is $n_y < 2e$ and degree d. Using Lemma 3.1 this adds $\tilde{O}(e(d + \delta))$ operations. □

Example 3.2. We continue with $a = x^2y + y$ and $b = xy^2 + x$ as in Example 3.1; S_x and S_y have dimension $n_x = n_y = 3$. For $f = y^3 + x^3y^2 + 1$, we first reduce the y-degree using S_x. Since $\delta = \deg_x f \geq n_x$, we cannot directly use S_x which is 3×3. Following the proof of Lemma 3.2 we increase the dimension and consider $T_x \in \mathbb{K}[y]^{4\times4}$, the Sylvester matrix with respect to x associated to a and xb. The first division is therefore:

$$v_x(f) = \begin{bmatrix} y^2 \\ 0 \\ 0 \\ y^3+1 \end{bmatrix} = T_x w_1 + v_x(f') = \begin{bmatrix} y & 0 & y^2+1 & 0 \\ 0 & y & 0 & y^2+1 \\ y & 0 & 0 & 0 \\ 0 & y & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ y^2 \\ 1 \\ -y \end{bmatrix} + \begin{bmatrix} -1 \\ y \\ 0 \\ 1 \end{bmatrix}.$$

The new polynomial is $f' = -x^3 + yx^2 + 1$, its y-degree is $1 < n_y$, so the division by $S_y \in \mathbb{K}[x]^{3\times3}$ in order to reduce the x-degree is now possible:

$$v_y(f') = \begin{bmatrix} 0 \\ x^2 \\ -x^3+1 \end{bmatrix} = S_y w_2 + v_y(\hat{f}) = \begin{bmatrix} x^2+1 & 0 & x \\ 0 & x^2+1 & 0 \\ 0 & 0 & x \end{bmatrix} \begin{bmatrix} x \\ 1 \\ -x^2 \end{bmatrix} + \begin{bmatrix} -x \\ -1 \\ 1 \end{bmatrix}$$

and we obtain the normal form $\hat{f} = -xy^2 - y + 1 \in \mathbb{K}[x, y]_{<(2,3)}$. □

The assumptions of Proposition 3.1 are central to be able to reduce the degree in x and also ensure the normal form. The following example describes a situation with the existence of roots at infinity with respect to y.

Example 3.3. With $\mathbb{K} = \mathbb{F}_7$, take $a = (x + 3)y + x^2 + 5x + 5$ and $b = (x + 3)(x + 4)y + x^2 + 4x + 2$. Then, the minimal polynomial of x in the quotient algebra is $x + 2$ but cannot be obtained by combinations of a and b of y-degree less than $e_a = e_b = 1$, that is using combinations of the columns of S_y. The vector $[0 \;\; x + 2]^\mathsf{T}$ is its own remainder of the division by $S_y \in \mathbb{K}[x]^{2\times2}$, hence $x + 2$ is not reduced to zero while being in the ideal. In this case however, thanks to the random conditioning of Section 5, we correctly compute the resultant (see Section 7).

Since the multiplication in $\mathbb{K}[x, y]$ can be computed in quasi-linear time [22, Sec. 8.4], Proposition 3.1 allows multiplication in $\mathbb{K}[x, y]/\langle a, b\rangle$ using $\tilde{O}(de)$ arithmetic operations. This is valid as soon as both Sylvester matrices are column reduced. From Lemma 2.3 this means that the x-leading (resp. y-) coefficients of a and b are coprime and one of them has maximal degree d_a or d_b (resp. e_a or e_b). In a complementary situation, that is with a sufficiently generic ideal $\langle a, b\rangle$ for the graded lexicographic order and using the total degree, a quasi-linear complexity was already achieved in [26] for the multiplication in such a quotient. Even though it retains specific assumptions on the ideal, let us also mention the multiplication bound $\tilde{O}((de)^{1.5})$ of [30, Sec. 4.5].

3.4 Power projections via transposed normal form

Using Shoup's general approach for the computation of minimal polynomials in a quotient algebra, we especially rely on the fact that the power projection problem is the transpose of the modular composition problem [57; 35, Sec. 6]. The normal form algorithm of Proposition 3.1 treats a special case of modular composition since $f \bmod I$ can be seen as $f(g(x)) \bmod I$ for $g = x$. Certain power projections can therefore already be derived by transposition from what we have done so far, as we explain in this section. This is used at the core of the general algorithm in Section 4 for the computation of a larger number of $O(de)$ projections efficiently for $\mathbb{K} = \mathbb{F}_q$.

Consider the restriction $\varphi_{\delta,\eta}$ of φ to the $\mathbb{K}$-vector space $\mathcal{U} = \mathbb{K}[x, y]_{\leq(\delta,\eta)}$, and denote $\mathbb{K}[x, y]_{<(d,n_y)}$ as a $\mathbb{K}$-vector space by $\mathcal{V}$. We also introduce the dual spaces $\widehat{\mathcal{U}}$ and $\widehat{\mathcal{V}}$ of the $\mathbb{K}$-linear forms on $\mathcal{U}$ and $\mathcal{V}$, respectively. The transpose of $\varphi_{\delta,\eta}$ is the $\mathbb{K}$-linear map

$$\begin{aligned} \varphi_{\delta,\eta}^\mathsf{T} : \widehat{\mathcal{V}} &\to \widehat{\mathcal{U}} \\ \ell &\mapsto \ell \circ \varphi_{\delta,\eta}. \end{aligned} \tag{5}$$

We view the polynomials in $\mathcal{U}$ as vectors on the monomial basis $\mathcal{B} = \{1, x, \ldots, x^\delta, y, xy, \ldots, x^\delta y^\eta\}$. The linear forms in $\widehat{\mathcal{U}}$ on the dual basis of $\mathcal{B}$ are represented by vectors in $\mathbb{K}^{(\delta+1)(\eta+1)}$. The elements in $\mathcal{V}$ and $\widehat{\mathcal{V}}$ are viewed in $\mathbb{K}^{dn_y}$ on

the basis $\{y^{n_y-1}, y^{n_y-1}x, \ldots, y^{n_y-1}x^{d-1}, y^{n_y-2}, y^{n_y-2}x, \ldots, x^{d-1}\}$ of $\mathcal{V}$ (in accordance with the definition of the Sylvester matrix S_y). From Eq. (5), the entries of $\varphi^{\mathsf{T}}_{\delta,\eta}(\ell)$ are the bivariate power projections

$$(\ell \circ \varphi_{\delta,\eta})(x^i y^j) \text{ for } 0 \le i \le \delta \text{ and } 0 \le j \le \eta. \tag{6}$$

We compute these projections by applying the transposition principle [7, 21; 57]. The principle asserts that if a $\mathbb{K}$-linear map $\phi : \mathcal{E}_1 \to \mathcal{E}_2$ can be computed by a linear straight-line program of length l, then the transpose map can be computed by a program of length $l + \dim \mathcal{E}_2$ (l if ϕ is an isomorphism) [12, Thm. 13.20]. We use a commonly applied strategy to implement the principle [9]. Proposition 3.2 follows directly from Proposition 3.1 e.g. by mimicking the results of [8, Sec. 4] in $\mathbb{K}[x,y]/\langle f(x), g(y)\rangle$ or [51, 53] modulo triangular sets. The change concerns only the way in which the ideal is represented.

Proposition 3.2. *Assume that the Sylvester matrices S_x and S_y associated to a and b are column reduced. Given two integers $\delta, \eta \ge 0$ and $\ell \in \widehat{\mathcal{V}}$ one can compute $\varphi^{\mathsf{T}}_{\delta,\eta}(\ell)$ using $\tilde{O}((d+\delta)(e+\eta))$ arithmetic operations in $\mathbb{K}$.*

Proof. The claim on $\varphi^{\mathsf{T}}_{\delta,\eta}$ is going to follow from the application of the transposition principle to the algorithm of Proposition 3.1 for $\varphi_{\delta,\eta}$, with portions written as a $\mathbb{K}$-linear straight-line program.

The computation of $\varphi_{\delta,\eta}$ reduces to two applications of Lemma 3.1, hence it suffices to study the transposition of the matrix division with remainder algorithm. Regarding this algorithm, observe that if the matrix inverses such as in Eq. (3) are pre-computed, then afterwards only $\mathbb{K}$-linear forms in the entries of the input vector are involved. Furthermore, these linear forms can be computed by $\mathbb{K}$-linear straight-line computations. The division algorithm of Lemma 3.1 can therefore be viewed as follows. The inverse of the reversed Sylvester matrix in Eq. (3) is pre-computed over truncated power series in time as stated in Lemma 3.1, using the complexity bounds in Section 3.1. Then the linear operations involving the input vector, including structured matrix times vector products [6], are performed within the same cost bound. The transposed division algorithm follows: it uses the pre-computed inverse as a parameter, and it is obtained from the transposition principle applied to the linear straight-line remaining portions. For the transposed division, this leads to the same complexity bound as stated in Lemma 3.1, and for the transpose $\varphi^{\mathsf{T}}_{\delta,\eta}$, to the bound as in Proposition 3.1. □

We directly use the transposition principle. The transpose algorithm could nevertheless be stated more explicitly as done for the univariate case in [9] — using duality with linear recurrence sequence extension [56], and for multivariate triangular sets in [51, 53].

4 Application of Kedlaya and Umans' techniques

The minimal polynomial of the multiplication by x in $\mathbb{A}$ requires the computation of projections of $O(de)$ power of x (Section 6). Proposition 3.2 therefore is not sufficient in order to achieve quasi-linear complexity. This now leads us to apply Kedlaya and Umans' techniques [40], and their extensions in [53, 28], for efficient modular composition over a finite field (Corollary 4.1) and power projection by transposition (Corollary 4.2).

Given three polynomials $f, g, h \in \mathbb{K}[x]$ with $\deg(f) < n$ and $\deg(g) < n$ where $n = \deg(h)$, the problem of modular composition is to compute $f(g) \bmod h$ [11]. (The problem is more fundamentally stated over a ring.) At the very beginning in this case, we benefit from the fact that for such polynomials the division with remainder can be computed using $\tilde{O}(n)$ arithmetic operations [22, Sec. 9.1]. One of the difficulties in the bivariate case is to be able to start from an analogous point, we mean from an efficient division with remainder modulo I. Once this is achieved, the approach of [40] can be followed for both modular composition and power projection. This is what has been accomplished in [53] (multivariate case) and [28] (special case $g = x$), with respective shapes of the ideal I that we have already mentioned. We proceed in the same way, and integrate the new division (normal form) algorithm into the overall process. We therefore do not repeat all the details for the proof of Theorem 4.1 and its corollaries, and refer the reader to the stem papers. As for Proposition 3.2 our change is the way in which the ideal is represented, which leads to a new modular bivariate projection algorithm in Corollary 4.2.

The first main ingredient is to reduce the problem of division (of modular composition), to divisions with smaller input degrees and to a problem of multipoint evaluation [40, Pb. 2.1]. More precisely, Theorem 4.1 shows that the

problem of computing the normal form of $f \in \mathbb{K}[x]_{<\delta}$ modulo I can be reduced, for $2 \le d_\epsilon < \delta$, to normal forms of polynomials of x- and y-degrees less than $d_\epsilon d \log \delta$ and $d_\epsilon e \log \delta$, respectively, and to multipoint evaluation. Here, remember the notations $d = \max\{deg_x a, deg_x b\}$ and $e = \max\{deg_y a, deg_y b\}$. The Sylvester matrix S_y is $n_y \times n_y$ over $\mathbb{K}$.

Theorem 4.1 ([40, Thm. 3.1], generalized in [53, 28]). *Consider $f \in \mathbb{K}[x]$ of degree less than δ, and an arbitrary integer $2 \le d_\epsilon < \delta$. Assume that the Sylvester matrices S_x and S_y associated to a and b are column reduced, and $|\mathbb{K}| > l(d_\epsilon - 1)\max\{d-1, n_y - 1\}$ where $l = \lceil \log_{d_\epsilon}(\delta) \rceil$. If $\delta = O(de)$ then $f(x)$ rem I can be computed using $\tilde{O}(d_\epsilon^2 de)$ arithmetic operations in $\mathbb{K}$, plus one multivariate multipoint evaluation of a polynomial with l variables over $\mathbb{K}$ and individual degrees less than d_ϵ, at $O(l^2 d_\epsilon^2 de)$ points in $\mathbb{K}^l$.*

Proof. The following six steps are those of the proof of [40, Thm. 3.1].

1. We first appeal to the inverse Kronecker substitution [40, Dfn. 2.3], in order to map f to a polynomial with l variables and degree less than d_ϵ in each variable. This $\mathbb{K}$-linear map from $\mathbb{K}[x]_{<\delta}$ to $\mathbb{K}[z_0, \ldots, z_{l-1}]_{<(d_\epsilon, \ldots, d_\epsilon)}$ is defined as follow. For $0 \le k < \delta$, the monomial x^k is sent to $z_0^{k_0} z_1^{k_1} \ldots z_{l-1}^{k_{l-1}}$, where $k_0, k_1, \ldots, k_{l-1}$ are the coefficients of the expansion of k in base d_ϵ. This is extended linearly to $\mathbb{K}[x]_{<\delta}$, and f is mapped in this way to a polynomial $\phi \in \mathbb{K}[z_0, \ldots, z_{l-1}]_{<(d_\epsilon, \ldots, d_\epsilon)}$. The map is injective on $\mathbb{K}[x]_{<\delta}$ and is computed in linear time using monomial bases.
2. Then we compute the polynomials $\chi_i = x^{d_\epsilon^i}$ rem I in $\mathbb{K}[x, y]_{<(d, n_y)}$, for $i = 0, \ldots, l-1$. This corresponds to l exponentiations by d_ϵ modulo I. By successive bivariate multiplications [22, Sec. 8.4], each followed by a reduction modulo the ideal, this can be done in time $\tilde{O}(de)$ from Proposition 3.1.
   A key property is that $f(x)$ rem $I = \phi(\chi_0, \ldots, \chi_{l-1})$ rem I. This leads to the idea of first computing $\phi(\chi_0, \ldots, \chi_{l-1})$ by evaluation-interpolation, and to perform only afterwards the reduction modulo the ideal. We have that the degree of $\phi(\chi_0, \ldots, \chi_{l-1})$ is at most $\delta' = l(d_\epsilon - 1)(d-1)$ in x, and $\eta' = l(d_\epsilon - 1)(n_y - 1)$ in y.
3. We choose subsets K_1 and K_2 of $\mathbb{K}$ or cardinalities $\delta' + 1$ and $\eta' + 1$, respectively. By multipoint bivariate evaluation, we compute all values $\mu_{i,j,k} = \chi_i(\lambda_j, \lambda_k) \in \mathbb{K}$ for $i = 0, \ldots, l-1$ and $(\lambda_j, \lambda_k) \in K_1 \times K_2$. Using univariate evaluation [22, Sec. 10.1], variable by variable, this can be done using $\tilde{O}(l\delta'\eta')$ hence $\tilde{O}(d_\epsilon^2 de)$ operations ($n_y \le 2e$).
4. This is followed by all the evaluations $\phi(\mu_{1,j,k}, \ldots, \mu_{l,j,k})$, which is multipoint evaluation of a polynomial with l variables, with individual degrees less than d_ϵ, at $(\delta'+1)(\eta'+1)$ points in $\mathbb{K}^l$.
5. From there, $\phi(\chi_0, \ldots, \chi_{l-1})$ is recovered using bivariate interpolation from its values just obtained at $K_1 \times K_2$, this uses $\tilde{O}(d_\epsilon^2 de)$ operations in a way similar to multipoint bivariate evaluation above.
6. Finally, $f(x)$ rem $I = \phi(\chi_0, \ldots, \chi_{l-1})$ rem I. We know from Proposition 3.1 that this costs $\tilde{O}(\delta'\eta')$ operations, which is $\tilde{O}(d_\epsilon^2 de)$. □

In line with [40, Thm 7.1] and [53, 28], thanks to fast multipoint evaluation [40, Cor. 4.5], we now can bound the cost of the reduction of a univariate polynomial modulo the ideal.

Corollary 4.1. *Let $\mathbb{K}$ be a finite field $\mathbb{F}_q$. Assume that the Sylvester matrices S_x and S_y associated to a and b are column reduced, and consider $f \in \mathbb{F}_q[x]$ of degree less than $\delta = 4de$. For every constant $\epsilon > 0$, if $q \ge \delta^{1+\epsilon}$, then f rem I can be computed using $O((de)^{1+\epsilon}\log(q)^{1+o(1)})$ bit operations.*

Proof. Depending on ϵ, we choose a large enough constant integer c for $d_\epsilon = \lceil \delta^{1/c} \rceil$ to be sufficiently small compared to de. We have in particular, $d_\epsilon < \delta^\epsilon$, and $l = \lceil \log_{d_\epsilon}(\delta) \rceil \le c$. For δ large enough this leads to $l(d_\epsilon - 1)\max\{d - 1, n_y - 1\} \le q$ and therefore we can apply Theorem 4.1. We know that f rem I can be computed using $\tilde{O}(d_\epsilon^2 de)$ operations in $\mathbb{F}_q$, which is $O(de)^{1+\epsilon}$, plus the cost of the associated multipoint evaluation. Then we use the fact that for every constant $\gamma > 0$, there is an algorithm for evaluating a polynomial in $\mathbb{F}_q[z_0, \ldots, z_{l-1}]_{<(d_\epsilon, \ldots, d_\epsilon)}$ at n points in $\mathbb{F}_q^l$ using $(d_\epsilon^l + n)^{1+\gamma}\log(q)^{1+o(1)}$ bit operations, when the individual degrees d_ϵ are sufficiently large, and the number of variables l is at most $d_\epsilon^{o(1)}$ [40, Cor. 4.5]. Considering the evaluation parameters in Theorem 4.1 and $l \le c$, for every $\gamma > 0$, the cost of the multipoint evaluation then is $O((\delta + d_\epsilon^2 de)^{1+\gamma}\log(q)^{1+o(1)})$, which allows to obtain the claimed complexity bound. □

As it has been said before, our presentation is simplified compared to the one of [40]. The dependence in q, in complexity bounds analogous to the one in Corollary 4.1, is made explicit and written using polylogarithmic functions in [53]. The study of [28] uses an explicit function of slow increase for the number of variables of the multipoint evaluation problem. Sharper bounds and improved algorithms can be found in [27, 5] for multipoint evaluation, and [29]

for multivariate modular composition over finite fields. Since we rely also on a solution for the dual problem, and that it is not treated in the latter references, we remain essentially based on [40].

In a similar way to what we did in Section 3.4 we now transpose the algorithm of Corollary 4.1. Our reasoning is that of [40, Thm 7.7], [53, Thm. 3.3] and [28, Prop. 1] for modular power projection. We use the notation φ of Eq. (4) for the normal form map.

Corollary 4.2. *Let $\mathbb{K}$ be a finite field $\mathbb{F}_q$. Assume that the Sylvester matrices S_x and S_y associated to a and b are column reduced. Let ℓ be a linear form in the dual of $\mathbb{F}_q[x,y]_{<(d,n_y)}$. For every constant $\epsilon > 0$, if $q \geq \delta^{1+\epsilon}$ with $\delta = 4de$, then the projections $(\ell \circ \varphi)(x^i)$ for $0 \leq i < \delta$ can be computed using $O((de)^{1+\epsilon}\log(q)^{1+o(1)})$ bit operations.*

Proof. From Eq. (6), we have to compute $\varphi^{\mathsf{T}}_{\delta-1,0}(\ell)$. The claim follows from the transposition principle (Section 3.4) applied to the successive algebraic steps of the normal form algorithm of Corollary 4.1, in reverse order. The non-algebraic portions of the algorithm involved in multipoint evaluation are treated by means of [40, Thm. 7.6]. The steps of the algorithm are given in the proof of Theorem 4.1 [40, Thm. 3.1]. The four of them that depend on the input f have to be considered, these are Steps 1, 4, 5, and 6, that we see as $\mathbb{F}_q$-linear maps. The last step 6 is reduction modulo I, the transpose is obtained from Proposition 3.2. Step 5 is bivariate interpolation, computed by interpolating in x then in y. This is transposed using two transposed univariate interpolation [36, 9]. Step 4 is multivariate evaluation using [40, Cor. 4.5]. The transpose is given by [40, Thm. 7.6] when the ambient dimension is equal to the number of evaluation points, i.e. the linear map can be represented by a square matrix. The general case in which we are, with a larger number of evaluation points, is treated as in the proof of [40, Thm. 7.7] using several instances of the square case with a cost that fits the claimed bound. Finally, the transpose of the inverse Kronecker substitution at Step 1 is a projection that takes linear time. In view of the transposition principle and of [40, Thm. 7.6], the algorithm obtained from those transpositions computes the power projections using $O((de)^{1+\epsilon}\log(q)^{1+o(1)})$ bit operations, as in Corollary 4.1. □

5 Non-singular leading matrices using random shifts and reversals

In order to exploit the powers projections of Corollary 4.2 for a minimal polynomial computation, and derive the last invariant factor of the Smith normal form of S_y, we need to address a column reducedness issue. This is what we do in this section. If the input polynomials a and b lead to S_x and S_y with singular leading coefficients, then we construct two new polynomials a' and b' which allow to get around the difficulty.

Lemma 5.1 (Conditioning of S_x). *Given $\alpha \in \mathbb{K}$ not a root of $\mathrm{Res}_x(a,b)$ (the ideal is zero-dimensional), in arithmetic time $\tilde{O}(de)$ we can compute two polynomials a' and b' with degrees as those of a and b, such that the new Sylvester matrix S'_x is column reduced and the Smith normal form of S'_y is that of S_y.*

Proof. Consider $a^{(1)}(x,y) = a(x, y+\alpha)$ and $b^{(1)}(x,y) = b(x, y+\alpha)$. The new Sylvester matrix $S^{(1)}_x$ with respect to x has a non-singular constant term since $(\mathrm{Res}_x(a,b))(\alpha) \neq 0$. The Smith normal form of $S^{(1)}_y$ is equal to the Smith normal form of S_y. Indeed, let $Q_{\alpha,k} \in \mathbb{K}^{k\times k}$ be the matrix of the endomorphism that shifts a polynomial of degree less than k by α; $Q_{\alpha,k}$ is lower triangular with unit diagonal. We have

$$S^{(1)}_y = Q_{\alpha,e_a+e_b}\, S_y\, \mathrm{diag}(Q^{-1}_{\alpha,e_b}, Q^{-1}_{\alpha,e_a}), \tag{7}$$

hence $S^{(1)}_y$ and S_y are unimodularly equivalent. Then we consider the reversed polynomials a' and b' of $a^{(1)}$ and $b^{(1)}$ with respect to y, using the respective degrees e_a and e_b. Note that a' and b' must keep the same y-degrees, otherwise $S^{(1)}_x$ could not have a non-singular constant coefficient; for the same reason, the new matrix S'_x associated to a' and b' is column reduced. On the other hand, the Smith form with respect to y is unchanged since

$$S'_y = J_{e_a+e_b}\, S^{(1)}_y\, \mathrm{diag}(J_{e_b}, J_{e_a}), \tag{8}$$

where J_k is the reversal matrix of dimension k. The cost is dominated by the one of at most $2(d+1)$ shifts of polynomials of degree at most e in $\mathbb{K}[y]$, see e.g. [6][Chap. 1, Pb. 3.5]. □

Note that Lemma 5.1 preserves the Smith normal form of S_y but not necessarily its Hermite form. From Lemma 2.3, $a' = b' = 0$ has no roots at infinity with respect to y, so the last invariant factor of S_y is the minimal polynomial of the multiplication by x in the new quotient algebra (Lemma 2.1). The latter may have changed, with an extra factor coming from possible roots at infinity for $a = b = 0$.

We now do the same type of manipulation for the column reducedness of S_y and need a preliminary observation on reversed polynomial matrices. The reversal by columns of a matrix polynomial is the matrix whose entries are reversed with respect to the degree of their column.

Lemma 5.2 (Reversed Smith normal form). *The last invariant factor of the reversal of $A \in \mathbb{K}[x]^{n\times n}$ by columns is the reversal of the last invariant factor of A made monic and multiplied by some power of x.*

Proof. Let X in $\mathbb{K}[x]^{n\times n}$ with a determinant which is a power of x be such that the reversal R of A by columns is $A(1/x)X$. Let S_A be the Smith normal form of A, with unimodular matrices U and V such that $AV = US_A$. We have

$$RX^{-1}V(1/x) = U(1/x)S_A(1/x). \tag{9}$$

Let S_A^* be the diagonal matrix whose entries are the reversals of the diagonal entries of S, made monic by division by their leading coefficients. By multiplying Eq. (9) by an appropriate power of x, we obtain

$$RW_1 = W_2S_A^*$$

for two matrices W_1 and W_2 in $\mathbb{K}[x]^{n\times n}$ whose determinants are powers of x. Now let S_R^* be the diagonal matrix whose diagonal entries are the invariant factors of R divided by the largest power of x they contain. Using similar manipulations as above we get

$$S_R^*W_3 = W_4S_A^*$$

for two matrices W_3 and W_4 in $\mathbb{K}[x]^{n\times n}$ whose determinants are also powers of x. By the multiplicativity of the Smith normal form [49, Ch. II, Thm. 2.15], noting that S_A^* and S_R^* are themselves in Smith normal form, we arrive at $S_R^* = S_A^*$. The claim follows since the last invariant factor of R is the one of S_R^* multiplied by some power of x, and the last invariant factor of S_A^* is the reversal of the last invariant factor of A divided by its leading coefficient. □

Lemma 5.3 (Conditioning of S_x and S_y). *Given $\alpha, \beta \in \mathbb{K}$ not roots of $\mathrm{Res}_x(a,b)$ and $\mathrm{Res}_y(a,b) \in \mathbb{K}[x]$, respectively (the ideal is zero-dimensional), in arithmetic time $\tilde{O}(de)$ we can compute two polynomials a' and b' with degrees as those of a and b such that: the new Sylvester matrices S'_x and S'_y are column reduced; the last invariant factor of S_y can be deduced from that of S'_y using $\tilde{O}(de)$ additional arithmetic operations.*

Proof. By applying Lemma 5.1 we can assume that a and b are such that S_x is column reduced, without modifying S_y so $\mathrm{Res}_y(a,b)$ either. We use arguments similar to those used in the proof of Lemma 5.1. We first take $a^{(1)}(x,y) = a(x+\beta,y)$ and $b^{(1)}(x,y) = b(x+\beta,y)$. The new Sylvester matrix $S_y^{(1)}$ with respect to y has a non-singular constant term since $(\mathrm{Res}_y(a,b))(\beta) \neq 0$. We denote the last invariant factor of S_y by $\sigma \in \mathbb{K}[x]$. The last invariant factor of $S_y^{(1)}$ is $\sigma_\beta = \sigma(x+\beta)$ and satisfies $\sigma_\beta(0) \neq 0$. Then we consider the reversed polynomials a' and b' of $a^{(1)}$ and $b^{(1)}$ with respect to x, using the respective degrees d_a and d_b. Since $S_y^{(1)}$ has a non-singular constant term, a' and b' keep the same x-degrees and the new matrix S'_y associated to a' and b' is column reduced.

We now prove the claims with a' and b'. We have just seen for the column reducedness of the y-Sylvester matrix. With respect to x, the Sylvester matrix is column reduced after the initial application of Lemma 5.1. Using Eqs. (7) and (8) from the proof of the latter lemma, now with β, $S_x^{(1)}$, and S'_x, we deduce that S'_x remains column reduced. Finally, we compute the last invariant factor σ of S_y from the one of S'_y. The Sylvester matrix S'_y is the reversal of $S_y^{(1)}$. Let σ'_β the reversal polynomial of σ_β. From Lemma 5.2 we deduce that the last invariant factor of S'_y is $cx^l\sigma'_\beta$ for some integer $l \geq 0$, and a non-zero $c \in \mathbb{K}$. Since $\sigma_\beta(0) \neq 0$, the reversal of $cx^l\sigma'_\beta$ is $c\sigma_\beta$. Using a shift by $-\beta$ and making the polynomial monic provides us with σ.

In addition to the cost in Lemma 5.1, we essentially have to perform at most $2(e+1)$ shifts of polynomials of degree at most d in $\mathbb{K}[x]$, plus a final shift of a polynomial of degree $O(de)$, see e.g. [6][Chap. 1, Pb. 3.5]. □

6 Invariant factor computation

For an appropriate random linear form ℓ, the minimal polynomial μ of the multiplication by x in $\mathbb{K}[x,y]/\langle a,b\rangle$ is also the one of the linearly generated sequence $(\ell\circ\varphi)(x^i)_{i\geq 0}$ with high probability [58, Sec. 4; 39, Lem. 6]. In essence, this minimal polynomial approach is a transcription of that of Wiedemann [64], with multiplication matrices rather than sparse ones [57]. This allows, in this section, to first bound the complexity of the minimal polynomial problem from the power projection complexity bound we have obtained previously (Corollary 4.2). Since μ has degree at most $2de$, it can ideed be computed from the first $4de$ terms of the power projection sequence. However, this is only valid when the involved Sylvester matrices are column reduced. Up to random shifts and reversals (Section 5), we then describe how the last invariant factor of S_y can be derived from the minimal polynomial of the multiplication by x in a slightly modified quotient algebra.

Theorem 6.1. *Consider two polynomials $a,b\in\mathbb{F}_q[x,y]_{\leq(d,e)}$ and assume that the associated Sylvester matrices S_x and S_y are column reduced. For every constant $\epsilon>0$, if $q\geq\delta^{1+\epsilon}$ with $\delta=4de$, there exists a randomized Monte Carlo algorithm which computes the minimal polynomial of the multiplication by x in $\mathbb{F}_q[x,y]/\langle a,b\rangle$ using $O((de)^{1+\epsilon}\log(q)^{1+o(1)})$ bit operations. The algorithm returns a divisor of the minimal polynomial, to which it is equal with probability at least $1-2de/q\geq 1/2$.*

Proof. The modular power projections as in Corollary 4.2 are computed for a random linear map ℓ. The sequence $\{(\ell\circ\varphi)(x^i)\}_{i\geq 0}$ is linearly generated; its minimal polynomial μ' is a divisor of the minimal polynomial μ of the multiplication by x in $\mathbb{A}$. Since $\deg\mu\leq 2de$, μ' can be computed using $\tilde{O}(de)$ additional operations in $\mathbb{K}$ from the $4de$ first terms of the sequence [22, Algo 12.9]. We can conclude by proving that $\mu'=\mu$ with high probability. Following the construction of φ in Eq. (4), one can define the multiplication map

$$\begin{aligned}\psi:\mathbb{F}_q[x,y]_{<(d,n_y)}&\to\mathbb{F}_q[x,y]_{<(d,n_y)}\\ f&\mapsto xf \bmod I.\end{aligned}\tag{10}$$

For an appropriate basis of $\mathbb{F}_q[x,y]_{<(d,n_y)}$ as a $\mathbb{F}_q$-vector space, we consider that ψ is represented by a matrix $M\in\mathbb{F}_q^{(dn_y)\times(dn_y)}$ and that 1 is represented by the vector $u\in\mathbb{F}_q^{dn_y}$. According to what we have seen in Section 3.4, we also represent linear forms in the dual of $\mathbb{F}_q[x,y]_{<(d,n_y)}$ by vectors in $\mathbb{F}_q^{dn_y}$. With this, μ is the minimal polynomial of u with respect to M. Hence for a random linear form ℓ represented by $v\in\mathbb{F}_q^{dn_y}$, the minimal polynomial of the linearly generated sequence $\{(\ell\circ\varphi)(x^i)\}_{i\geq 0}=\{v^{\mathsf{T}}M^iu\}_{k\geq 0}$ is μ with probability at least $1-\deg\mu/q$ [37, Lem. 2; 38, Lem. 1]. □

Corollary 6.1. *Consider two coprime polynomials $a,b\in\mathbb{F}_q[x,y]_{\leq(d,e)}$. For every constant $\epsilon>0$, there exists a randomized Monte Carlo algorithm which computes the last invariant factor of the Sylvester matrix associated to a and b with respect to either x or y, using $O((de)^{1+\epsilon}\log(q)^{1+o(1)})$ bit operations. The algorithm either returns the target invariant factor, and this with probability at least $1/2$, one of its divisors, or "failure".*

Proof. When $q\geq(12de)^{1+\epsilon}$, we randomly choose random α and β in $\mathbb{F}_q$, then check whether S'_x and S'_y as in Lemma 5.3 are column reduced. The check is performed using $\tilde{O}(d+e)$ operations, see Lemma 2.1 and e.g. [22, Thm. 11.10]. Since $\mathrm{Res}_y(a,b)\in\mathbb{F}_q[x]$ and $\mathrm{Res}_x(a,b)\in\mathbb{F}_q[y]$ have degree at most $2de$, the probability of success is at least $1-4de/q$. If the Sylvester matrices are column reduced, from Theorem 6.1, we then compute the minimal polynomial of the multiplication by x (or y) in the quotient algebra associated to S'_y (or S'_x). Lemma 2.1 tells us that we have actually computed the last invariant factor of S'_y (or S'_x) with probability at least $1-2de/q$. From Lemma 5.3 again, we finally derive the last invariant factor of S_y (or S_x). If q is too small, we construct an extension field of $\mathbb{F}_q$ with cardinality at least $(12de)^{1+\epsilon}$, that is of degree $O(\log(de))$. This can be done using an expected number of $\tilde{O}((\log(de)^2+\log(de)\log(q))$ bit operations [57] (see also [14] and [22, Sec. 14.9] in this regard). We then work in this extension, the costs induced are logarithmic factors which do not change our target cost bound, and the probability of success can be adjusted. □

7 Elimination ideal and resultant

When the system $a=b=0$ has no roots at infinity with respect to y, from Lemma 2.1 and Corollary 6.1 we obtain a Monte Carlo algorithm for computing the minimal polynomial of the multiplication by x, it is a generator of the elimination ideal $\langle a,b\rangle\cap\mathbb{F}_q[x]$.

Still with the absence of roots at infinity with respect to y, Lemma 2.2 indicates that if, moreover, the ideal has a shape basis $I = \langle \mu(x), y - \lambda(x) \rangle$ [23, 2], then the resultant of a and b is known. Note that the extra non-zero constant in Lemma 2.2 can be computed at the cost of $\tilde{O}(de)$ operations in $\mathbb{F}_q$ using evaluation in x.

For the resultant, we see that this leads to a weaker genericity assumption than in [28], where the total degree is used. Assume that the ideal $\langle a, b \rangle$ is in generic position for the lexicographic order $y > x$ so that $res_y(a, b) = c\mu$ with $c \neq 0 \in \mathbb{F}_q$. In this case, from Lemma 2.2 and Lemma 2.1 again, we can compute the resultant as the last invariant factor of S_y without the use of an additional condition with respect to the graded reverse lexicographic order [28]. This further allows us to deal with more general situations than that of the total degree since we obtain the resultant in all cases where $\mathrm{Res}_y(a, b) = c\mu$. This condition is sufficient but not necessary (Examples 2.1 and 3.3), the resultant can be computed when S_y has a unique non-trivial invariant factor. Note that the latter property can be formalized in the Zariski sense, for example by relying on ideals in general position with no roots at infinity [16, Sec. 3.5]. More precisely, there exists a non-zero polynomial Φ in $2(d + 1)(e + 1)$ variables over $\mathbb{K}$, such that the Smith form of S_y has a unique non-trivial invariant factor if Φ does not vanish at the coefficients of a and b.

The generic resultant algorithm becomes of the Las Vegas type when the degree of the resultant is known in advance, especially if the Sylvester matrix S_y is column reduced. In the latter case the degree of the resultant is indeed the sum of the column degrees of S_y [32, Eq. (24), p. 385].

References

[1] M.-E. Alonso, E. Becker, M.-F. Roy, and T. Wörmann. Zeros, multiplicities, and idempotents for zero-dimensional systems. In *Algorithms in Algebraic Geometry and Applications*, pages 1–15. PM vol. 143, Birkhaüser, 1996.

[2] E. Becker, T. Mora, M. G. Marinari, and C. Traverso. The Shape of the Shape Lemma. In *Proc. ISSAC*, pages 129–133. ACM Press, 1994.

[3] B. Beckermann and G. Labahn. A Uniform Approach for the Fast Computation of Matrix-Type Padé Approximants. *SIAM J. Matrix Analysis and Applications*, 15(3):804–823, 1994.

[4] J. Berthomieu, V. Neiger, and M. Safey El Din. Faster Change of Order Algorithm for Gröbner Bases under Shape and Stability Assumptions. In *Proc. ISSAC*, pages 409–418. ACM Press, 2022.

[5] V. Bhargava, S. Ghosh, Z. Guo, M. Kumar, and C. Umans. Fast Multivariate Multipoint Evaluation Over All Finite Fields. In *Proc. FOCS*, pages 221–232. IEEE, 2022.

[6] D. Bini and V. Y. Pan. *Polynomial and Matrix Computations*. Birkhäuser, 1994.

[7] J. L. Bordewijk. *Inter-reciprocity applied to electrical networks*. PhD thesis, Technische Hogeschool Delft, 1956.

[8] A. Bostan, P. Flajolet, B. Salvy, and E. Schost. Fast computation of special resultants. *J. Symb. Comput.*, 41(1):1–29, 2006.

[9] A. Bostan, G. Lecerf, and É. Schost. Tellegen's principle into practice. In *Proc. ISSAC*, pages 37–44. ACM Press, 2003.

[10] A. Bostan, B. Salvy, and É. Schost. Fast Algorithms for Zero-Dimensional Polynomial Systems using Duality. *Appl. Algebr. Eng. Comm.*, 14:239–272, 2003.

[11] R. P. Brent and H. T. Kung. Fast algorithms for manipulating formal power series. *J. ACM*, 25(4):581–595, 1978.

[12] P. Bürgisser, M. Clausen, and M. A. Shokrollahi. *Algebraic complexity theory*, volume 315 of *Grundlehren der mathematischen Wissenschaften*. Springer, 1997.

[13] D. Coppersmith, A. M. Odlzyko, and R. Schroeppel. Discrete logarithms in GF(p). *Algorithmica*, 1(1-4):1–15, 1986.

[14] J.-M. Couveignes and R. Lercier. Fast construction of irreducible polynomials over finite fields. *Isr. J. Math.*, 194(1):77–105, 2013.

[15] D. A. Cox and C. D'Andrea. Subresultants and the Shape Lemma. arXiv:2112.10306, 2021.

[16] D. A. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry*. Springer-Verlag, New-York, 1998. 2nd edition 2005.

[17] X. Dahan. Lexicographic Gröbner bases of bivariate polynomials modulo a univariate one. *J. Symb. Comput.*, 110:24–65, 2022.

[18] J.-C. Faugère, P. Gaudry, L. Huot, and G. Renault. Sub-cubic change of ordering for Gröbner basis: a probabilistic approach. In *Proc. ISSAC*, pages 170–177. ACM Press, 2014.

[19] J.-C. Faugère and C. Mou. Fast algorithm for change of ordering of zero-dimensional Gröbner bases with sparse multiplication matrices. In *Proc. ISSAC*, pages 115–122. ACM Press, 2011.

[20] J.-C. Faugère and C. Mou. Sparse FGLM algorithms. *J. Symb. Comput.*, 80(3):538–569, 2017.

[21] C. M. Fiduccia. *On the algebraic complexity of matrix multiplication.* PhD thesis, Brown University, 1973.

[22] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, 1999. Third edition 2013.

[23] P. Gianni, B. Trager, and G. Zacharias. Gröbner bases and primary decomposition of polynomial ideals. *J. Symb. Computation*, 6(2-3):149–167, 1988.

[24] L. Gonzalez-Vega, F. Rouillier, and M.-F. Roy. Symbolic Recipes for Polynomial System Solving. In *Algorithms and Computation in Mathematics, Some Tapas of Computer Algebra*, pages 34–65. Springer, 1999.

[25] J. van der Hoeven. On the Complexity of Multivariate Polynomial Division. In *Proc. ACA 2015*, pages 447–458. Springer, PROMS 198, 2017.

[26] J. van der Hoeven and R. Larrieu. Fast Gröbner basis computation and polynomial reduction for generic bivariate ideals. *Appl. Algebr. Eng. Comm.*, 30(6):509–539, 2019.

[27] J. van der Hoeven and G. Lecerf. Fast multivariate multi-point evaluation revisited. *J. of Complexity*, 56, 2020.

[28] J. van der Hoeven and G. Lecerf. Fast computation of generic bivariate resultants. *J. of Complexity*, 62, 2021.

[29] J. van der Hoeven and G. Lecerf. On the Complexity Exponent of Polynomial System Solving. *Found. Comput. Math.*, 21(1):1–57, 2021.

[30] S. G. Hyun, S. Melczer, É. Schost, and C. St-Pierre. Change of basis for m-primary ideals in one and two variables. In *Proc. ISSAC*, pages 227–234. ACM Press, 2019.

[31] N. Jacobson. *Basic Algebra I*. Dover Publications Inc., 2009. Second Edition W.H. Freeman 1985.

[32] T. Kailath. *Linear Systems*. Prentice-Hall, 1980.

[33] T. Kailath, S. Y. Kung, and M. Morf. Displacement ranks of matrices and linear equations. *J. Math. Anal. Appl.*, 68(2):395–407, 1979.

[34] E. Kaltofen. Asymptotically fast solution of Toeplitz-like singular linear systems. In *Proc. ISSAC*, pages 297–304. ACM Press, 1994.

[35] E. Kaltofen. Challenges of Symbolic Computation: My Favorite Open Problems. *J. Symbolic Computation*, 29(6):891–919, 2000.

[36] E. Kaltofen and Y. Lakshman. Improved Sparse Multivariate Polynomial Interpolation Algorithms. In *Proc. ISSAC*, pages 467–474. Springer, LNCS 358, 1988.

[37] E. Kaltofen and V. Y. Pan. Processor efficient parallel solution of linear systems over an abstract field. In *Proc. SPAA*, pages 180–191. ACM, 1991.

[38] E. Kaltofen and D. Saunders. On Wiedemann's method of solving sparse linear systems. In *AAECC-9*, volume 539 of *LNCS*, pages 29–38. Springer Verlag, 1991.

[39] E. Kaltofen and V. Shoup. Subquadratic-time factoring of polynomials over finite fields. *Mathematics of Computation*, 67(233):1179–1197, 1998.

[40] K. S. Kedlaya and C. Umans. Fast Polynomial Factorization and Modular Composition. *SIAM J. on Computing*, 40(6):1767–1802, 2011.

[41] G. Labahn. Inversion components of block Hankel-like matrices. *Linear Algebra Appl.*, 177:7–48, 1992.

[42] D. Lazard. Résolution des systèmes d'équations algébriques. *Theor. Comput. Sci.*, 15(1):77–110, 1981.

[43] D. Lazard. Ideal Bases and Primary Decomposition: Case of Two Variables. *J. Symb. Comput.*, 1(3):261–270, 1985.

[44] D. Lazard. Solving zero-dimensional algebraic systems. *J. Symb. Comput.*, 13(2):117–131, 1992.

[45] R. Lebreton, E. Mehrabi, and É. Schost. On the complexity of solving bivariate systems: the case of non-singular solutions. In *Proc. ISSAC*, pages 251–258, 2013.

[46] G. Lecerf. On the complexity of the Lickteig-Roy subresultant algorithm. HAL report, CNRS & École Polytechnique, 2017.

[47] E. Mehrabi and É. Schost. A softly optimal Monte Carlo algorithm for solving bivariate polynomial systems over the integers. *J. of Complexity*, 34:78–128, 2016.

[48] B. Mourrain and V. Y. Pan. Multivariate Polynomials, Duality, and Structured matrices. *J. of Complexity*, 16(1):110–180, 2000.

[49] M. Newman. *Integral Matrices*. Academic Press, 1972. First edition.

[50] V. Y. Pan. *Structured Matrices and Polynomials: Unified Superfast Algorithms*. Springer, 2001.

[51] C. Pascal and É. Schost. Change of order for bivariate triangular sets. In *Proc. ISSAC*, pages 277–284. ACM Press, 2006.

[52] C. Pernet, H. Signargout, and G. Villard. High-order lifting for polynomial Sylvester matrices. Hal-03740320, 2022.

[53] A. Poteaux and É. Schost. Modular Composition Modulo Triangular Sets and Applications. *Comput. Complex.*, 22(3):463–516, 2013.

[54] J. Rifà and J. Borrell. Improving the time complexity of the computation of irreducible and primitive polynomials in finite fields. In *Proc. AAECC*, LNCS 539, pages 352–359, 1991.

[55] F. Rouillier. Solving Zero-Dimensional Systems Through the Rational Univariate Representation. *Appl. Algebra Eng. Commun. Comput.*, 9:433–461, 1999.

[56] V. Shoup. A Fast Deterministic Algorithm for Factoring Polynomials over Finite Fields of Small Characteristic. In *Proc. ISSAC*, pages 14–21, 1991.

[57] V. Shoup. Fast Construction of Irreducible Polynomials over Finite Fields. *J. Symb. Comput.*, 17(5):371–391, 1994.
[58] V. Shoup. A new polynomial factorization algorithm and its implementation. *J. Symb. Comput.*, 20(4):363–397, 1995.
[59] V. Shoup. Efficient computation of minimal polynomials in algebraic extensions of finite fields. In *Proc. ISSAC*, pages 53–58. ACM Press, 1999.
[60] A. Storjohann. *Algorithms for Matrix Canonical Forms*. PhD thesis, Institut für Wissenschaftliches Rechnen, ETH-Zentrum, Zurich, Switzerland, November 2000.
[61] J. A. Thiong Ly. Note for computing the minimun polynomial of elements in large finite fields. In *Proc. Coding Theo. App.*, LNCS 388, pages 185–192, 1989.
[62] G. Villard. Fast Parallel Algorithms for Matrix Reduction to Normal Forms. *Appl. Algebra Eng. Commun. Comput.*, 8(6):511–537, 1997.
[63] G. Villard. On Computing the Resultant of Generic Bivariate Polynomials. In *Proc. ISSAC*, pages 391–398. ACM Press, 2018.
[64] D. Wiedemann. Solving sparse linear equations over finite fields. *IEEE Trans. Information Theory*, 32(1):54–62, 1986.