NEARLY OPTIMAL ALGORITHMS FOR THE DECOMPOSITION OF MULTIVARIATE RATIONAL FUNCTIONS AND THE EXTENDED LÜROTH'S THEOREM

GUILLAUME CHÈZE

ABSTRACT. The extended Lüroth's Theorem says that if the transcendence degree of $\mathbb{K}(\mathsf{f}_1,\ldots,\mathsf{f}_m)/\mathbb{K}$ is 1 then there exists $f \in \mathbb{K}(\underline{X})$ such that $\mathbb{K}(\mathsf{f}_1,\ldots,\mathsf{f}_m)$ is equal to $\mathbb{K}(f)$. In this paper we show how to compute f with a probabilistic algorithm. We also describe a probabilistic and a deterministic algorithm for the decomposition of multivariate rational functions. The probabilistic algorithms proposed in this paper are softly optimal when n is fixed and d tends to infinity. We also give an indecomposability test based on gcd computations and Newton's polytope. In the last section, we show that we get a polynomial time algorithm, with a minor modification in the exponential time decomposition algorithm proposed by Gutierez-Rubio-Sevilla in 2001.

INTRODUCTION

Polynomial decomposition is the problem of representing a given polynomial $f(x)$ as a functional composition $g(h(x))$ of polynomials of smaller degree. This decomposition has been widely studied since 1922, see [27], and efficient algorithms are known in the univariate case, see [3, 9, 20, 37, 38] and in the multivariate case [11, 37, 40].

The decomposition of rational functions has also been studied, [41, 1]. In the multivariate case the situation is the following:
Let $f(X_1,\ldots,X_n) = f_1(X_1,\ldots,X_n)/f_2(X_1,\ldots,X_n) \in \mathbb{K}(X_1,\ldots,X_n)$ be a rational function, where $\mathbb{K}$ is a field and $n \geq 2$. It is commonly said to be composite if it can be written $f = u \circ h$ where $h(X_1,\ldots,X_n) \in \mathbb{K}(X_1,\ldots,X_n)$ and $u \in \mathbb{K}(T)$ such that $\deg(u) \geq 2$ (recall that the degree of a rational function is the maximum of the degrees of its numerator and denominator after reduction), otherwise f is said to be non-composite.

This decomposition appears when we study the kernel of a derivation, see [24]. In [24] the author gives a multivariate rational function decomposition algorithm, but this algorithm is not optimal and works only for fields of characteristic zero. In this paper, we give a probabilistic optimal algorithm. In other words, our algorithm decomposes $f \in \mathbb{K}(X_1,\ldots,X_n)$ with $\tilde{\mathcal{O}}(d^n)$ arithmetic operations, where d is the degree of f. We suppose in this work that d tends to infinity and n is fixed. We use the classical $\mathcal{O}$ and $\tilde{\mathcal{O}}$ ("soft $\mathcal{O}$") notation in the neighborhood of infinity as defined in [39, Chapter 25.7]. Informally speaking, "soft $\mathcal{O}$"s are used for readability in order to hide logarithmic factors in complexity estimates. Then, the size of the input and the number of arithmetic operations performed by our algorithm have the same order of magnitude. This is the reason why we call our algorithm "optimal".

Furthermore, our algorithm also works if the characteristic of $\mathbb{K}$ is greater than $d(d-1)+1$.

This decomposition also appears when we study intermediate fields of an unirational field. In this situation, the problem is the following: we have m multivariate rational functions $\mathsf{f}_1(\underline{X}), \ldots, \mathsf{f}_m(\underline{X}) \in \mathbb{K}(\underline{X})$, and we want to know if there exists a proper intermediate field $\mathbb{F}$ such that $\mathbb{K}(\mathsf{f}_1, \ldots, \mathsf{f}_m) \subset \mathbb{F} \subset \mathbb{K}(\underline{X})$. In the affirmative case, we want to compute $\mathbb{F}$. If $tr.deg_{\mathbb{K}}(\mathbb{F}) = 1$ then by the extended Lüroth's Theorem, see [31, Theorem 3 p. 15] we have $\mathbb{F} = \mathbb{K}(f)$.

Theorem 1 (Extended Lüroth's Theorem). *Let $\mathbb{F}$ be a field such that $\mathbb{K} \subset \mathbb{F} \subset \mathbb{K}(X_1, \ldots, X_n)$ and $tr.deg_{\mathbb{K}}(\mathbb{F}) = 1$. Then there exists $f \in \mathbb{K}(X_1, \ldots, X_n)$ such that $\mathbb{F} = \mathbb{K}(f)$.*

The classical Lüroth's Theorem is stated with univariate rational functions. Theorem 1 gives an extension to multivariate rational functions. This extended theorem was first proved by Gordan in characteristic zero, see [13], and by Igusa in general, see [17]. There exist algorithms to compute f, called a Lüroth's generator, see e.g. [15, 25].
Thanks to the Extended Lüroth's Theorem the computation of intermediate fields is divided into two parts: first we compute a Lüroth's generator f, and second we decompose f. Then $f = u \circ h$, and $\mathbb{F} = \mathbb{K}(h)$ is an intermediate field. In [15] the authors show that the decomposition of f bijectively corresponds to intermediate fields. They also give algorithms to compute a Lüroth's generator and to decompose it. Unfortunately, the decomposition algorithm has an exponential time complexity, but the complexity analysis of this algorithm is too pessimistic. Indeed, in the last section of this paper we show that we can modify it and get an algorithm with a polynomial time complexity.

The decomposition of rational functions also appears when we study the spectrum of a rational function. In this paper we use this point of view in order to give fast algorithms.
Let $\overline{\mathbb{K}}$ be an algebraic closure of $\mathbb{K}$. Let $f = f_1/f_2 \in \mathbb{K}(X_1, \ldots, X_n)$ be a rational function of degree d. The set

$$\sigma(f_1, f_2) = \{(\mu : \lambda) \in \mathbb{P}^1_{\mathbb{K}} \quad | \quad \mu f_1 - \lambda f_2 \text{ is reducible in } \overline{\mathbb{K}}[X_1, \ldots, X_n],$$
$$\text{or } \deg(\mu f_1 - \lambda f_2) < d \}$$

is the spectrum of $f = f_1/f_2$. We recall that a polynomial reducible in $\overline{\mathbb{K}}[X_1, \ldots, X_n]$ is said to be absolutely reducible.
A classical theorem of Bertini and Krull, see Theorem 22, implies that $\sigma(f_1, f_2)$ is finite if f_1/f_2 is non-composite. Actually, $\sigma(f_1, f_2)$ is finite if and only if f_1/f_2 is non-composite and if and only if the pencil of algebraic curves $\mu f_1 - \lambda f_2 = 0$ has an irreducible general element (see for instance [18, Chapitre 2, Théorème 3.4.6] and [7, Theorem 2.2] for detailed proofs).
To the author's knowledge, the first effective result about the spectrum has been given by Poincaré [26]. He showed that $|\sigma(f_1, f_2)| \leq (2d-1)^2 + 2d + 2$. This bound was improved by Ruppert [28] who proved that

$$|\sigma(f_1, f_2)| \leq d^2 - 1.$$

This result was obtained as a byproduct of a very interesting technique developed to decide the reducibility of an algebraic plane curve.

Several papers improve this result, see e.g. [23, 36, 2, 7, 4].

The previous result says that if f_1/f_2 is a non-composite reduced rational function then for all but a finite number of $\lambda \in \mathbb{K}$ we have: $f_1 + \lambda f_2$ is absolutely irreducible (i.e. irreducible in $\overline{\mathbb{K}}[X_1, \ldots, X_n]$). Furthermore, the number of "bad" values of λ is lower than $d^2 - 1$. Thus we can deduce a probabilistic test for the decomposition of a rational function, based on an absolute irreducibility test. In this paper we will give a decomposition algorithm based on this kind of idea. Furthermore, we will see that this algorithm is softly optimal when the following hypotheses are satisfied:

Hypothesis (C):

$\mathbb{K}$ is a perfect field of characteristic 0 or at least $d(d-1) + 1$.

Hypothesis (H):

$$\begin{cases} (i)\deg(f_1 + \Lambda f_2) = \deg_{X_n}(f_1 + \Lambda f_2), \text{ where } \Lambda \text{ is a new variable,} \\ (ii)\, Res_{X_n}\Big(f_1(\underline{0}, X_n) + \Lambda f_2(\underline{0}, X_n),\, \partial_{X_n} f_1(\underline{0}, X_n) + \Lambda \partial_{X_n} f_2(\underline{0}, X_n)\Big) \neq 0 \text{ in } \mathbb{K}[\Lambda]. \end{cases}$$

where $\deg_{X_n} f$ represents the partial degree of f in the variable X_n, $\deg f$ is the total degree of f and Res_{X_n} denotes the resultant relatively to the variable X_n.

These hypotheses are necessary, because we will use the factorization algorithms proposed in [22], where these kinds of hypotheses are needed. Actually, in [22] the author studies the factorization of a polynomial F and uses hypothesis (C) and hypothesis (L), where (L) is the following:

Hypothesis (L):

$$\begin{cases} (i)\deg_{X_n} F = \deg F, \text{ and } F \text{ is monic in } X_n, \\ (ii)\, Res_{X_n}\big(F(\underline{0}, X_n), \frac{\partial F}{\partial X_n}(\underline{0}, X_n)\big) \neq 0. \end{cases}$$

If F is squarefree, then hypothesis (L) is not restrictive since it can be assured by means of a generic linear change of variables, but we will not discuss this question here (for a complete treatment in the bivariate case, see [10, Proposition 1]).

Roughly speaking, our hypothesis (H) is the hypothesis (L) applied to the polynomial $f_1 + \Lambda f_2$. In (H,i) we do not assume that $f_1 + \Lambda f_2$ is monic in X_n. Indeed, after a generic linear change of coordinates, the leading coefficient relatively to X_n can be written: $a + \Lambda b$, with $a, b \in \mathbb{K}$. In our probabilistic algorithm, we evaluate Λ to $\lambda \notin \sigma(f_1, f_2)$, thus $\deg(f_1 + \lambda f_2) = \deg(f_1 + \Lambda f_2)$ and $a + \lambda b \neq 0$. Then we can consider the monic part of $f_1 + \lambda f_2$ and we get a polynomial satisfying (L,i). Then (H,i) is sufficient in our situation. Furthermore, in this paper, we assume f_1/f_2 to be reduced, i.e. f_1 and f_2 are coprime. We recall in Lemma 6 that in this situation $f_1 + \Lambda f_2$ is squarefree. Thus hypothesis (H) is not restrictive.

Complexity model. In this paper the complexity estimates charge a constant cost for each arithmetic operation ($+$, $-$, $\times$, $\div$) and the equality test. All the constants in the base fields (or rings) are thought to be freely at our disposal.

In this paper we suppose that *the number of variables n is fixed* and that the degree d tends to infinity. Furthermore, we say that an algorithm is softly optimal if it works with $\tilde{\mathcal{O}}(N)$ arithmetic operations where N is the size of the input.

Polynomials are represented by dense vectors of their coefficients in the usual monomial basis. For each integer d, we assume that we are given a computation tree that computes the product of two univariate polynomials of degree at most d with at most $\tilde{\mathcal{O}}(d)$ operations, independently of the base ring, see [39, Theorem 8.23]. We use the constant ω to denote a *feasible matrix multiplication* exponent as defined in [39, Chapter 12]: two $n \times n$ matrices over $\mathbb{K}$ can be multiplied with $\mathcal{O}(n^\omega)$ field operations. As in [8] we require that $2 < \omega \leq 2.376$. We recall that the computation of a solution basis of a linear system with m equations and $d \leq m$ unknowns over $\mathbb{K}$ takes $\mathcal{O}(md^{\omega-1})$ operations in $\mathbb{K}$ [8, Chapter 2] (see also [33, Theorem 2.10]).
In [22] the author gives a probabilistic (resp. deterministic) algorithm for the multivariate rational factorization. The rational factorization of a polynomial f is the factorization in $\mathbb{K}[\underline{X}]$, where $\mathbb{K}$ is the coefficient field of f. This algorithm uses one factorization of a univariate polynomial of degree d and $\tilde{\mathcal{O}}(d^n)$ (resp. $\tilde{\mathcal{O}}(d^{n+\omega-1})$) arithmetic operations, where d is the total degree of the polynomial and $n \geq 3$ is the number of variables. If $n = 2$, in [21],[22, Errata], the author gives a probabilistic (resp. deterministic) algorithm for the rational factorization. The number of arithmetic operations of this algorithm belongs to $\tilde{\mathcal{O}}(d^3)$ (resp. $\tilde{\mathcal{O}}(d^{\omega+1})$). We note that for $n \geq 3$ if the cost of the univariate polynomial factorization belongs to $\tilde{\mathcal{O}}(d^n)$ then the probabilistic algorithm is softly optimal.

Main Theorems. The following theorems give the complexity results about our algorithms. Although we will use no probabilistic model of computation, we will informally say *probabilistic algorithms* when speaking about the computation trees occurring in the next theorems. For the sake of precision, we prefer to express the probabilistic aspects in terms of families of computation trees. Almost all the trees of a family are expected to be executable on a given input (if the cardinality of $\mathbb{K}$ is large enough).

Theorem 2. *Let $f = f_1/f_2$ be a multivariate rational function in $\mathbb{K}(X_1, \ldots, X_n)$ of degree d, there exists a family of computation trees over $\mathbb{K}$ parametrized by $z := (\underline{a}, \underline{b}) \in \mathbb{K}^{2n}$ such that:*

- *Any executable tree of the family returns a decomposition $u \circ h$ of f with h a non-composite rational function.*
- *If $\underline{a}, \underline{b}$ are not the roots of some non-zero polynomials the tree corresponding to z is executable.*

Furthermore, we have:

1. *An executable tree performs two factorizations in $\mathbb{K}[X_1, \ldots, X_n]$ of polynomials with degree d, and one computation of u.*
2. *Under hypothesis (C) and (H) we have this estimate: an executable tree performs one factorization of a univariate polynomial of degree d over $\mathbb{K}$ plus a number of operations in $\mathbb{K}$ belonging to $\tilde{\mathcal{O}}(d^n)$ if $n \geq 3$, or to $\tilde{\mathcal{O}}(d^3)$ if $n = 2$.*

Since we use the dense representation of f_1 and f_2, the size of f is of the order of magnitude d^n. The previous statement thus asserts that the complexity of our probabilistic algorithm is softly optimal for $n \geq 3$.
We precise the condition "If $\underline{a}, \underline{b}$ are not the roots of some non-zero polynomials" in Remark 13 and Remark 15.
In characteristic zero we can say that for almost all z the tree corresponding to z

is executable.

We also give a deterministic decomposition algorithm.

Theorem 3. *If $\mathbb{K}$ is a field with a least $\max(d^2, \frac{3}{2}d^2 - 2d + 1)$ elements, then the decomposition $f = u \circ h$, with h non-composite, can be computed with at most $\mathcal{O}(d^2)$ absolute factorizations of polynomials with degree d, and at most $\mathcal{O}(d^2)$ computations of u where f and h are given.*

If we can use the algorithm proposed in [10] and [22], as we will see in Remark 18, our deterministic algorithm uses one factorization of a univariate polynomial of degree d with algebraic coefficients of degree at most d, and at most $\tilde{\mathcal{O}}(d^{n+\omega+2})$ if $n \geq 3$ or $\tilde{\mathcal{O}}(d^6)$ if $n = 2$ arithmetic operations in $\mathbb{K}$.

With the tools used for the decomposition algorithms, we can compute a Lüroth's generator.

Theorem 4. *Let $\mathsf{f}_1, \ldots, \mathsf{f}_m \in \mathbb{K}(X_1, \ldots, X_n)$ be m rational functions of degree at most d. There exists a family of computation trees over $\mathbb{K}$ parametrized by $z = (z_1, \ldots, z_m) \in \mathbb{K}^{2nm}$, such that:*
If for all $i = 1, \ldots, m$, $z_i \in \mathbb{K}^{2n}$ belongs to an open Zariski set related to $\mathsf{f}_1, \ldots, \mathsf{f}_i$ then the tree corresponding to z is executable on $\mathsf{f}_1, \ldots, \mathsf{f}_m$ and it returns a Lüroth's generator of $\mathbb{K}(\mathsf{f}_1, \ldots, \mathsf{f}_m)$.
Furthermore, we have:

  (1) *An executable tree performs $2m$ gcd computations in $\mathbb{K}[X_1, \ldots, X_n]$ with polynomials of degree at most d.*
  (2) *If $\mathbb{K}$ has at least $(4d+2)d$ elements then we have the estimate: an executable tree performs $\tilde{\mathcal{O}}(md^n)$ arithmetic operations in $\mathbb{K}$.*

As before, this algorithm is softly optimal because the order of magnitude of the input is md^n. A precise description of the open Zariski set is given in Remark 29.

In the last section we prove the following result:

Theorem 5. *Let $f = f_1/f_2 \in \mathbb{K}(\underline{X})$.*
$f = u \circ h$, with $h = h_1/h_2$ if and only if $H(\underline{X}, \underline{Y}) = h_1(\underline{X})h_2(\underline{Y}) - h_2(\underline{X})h_1(\underline{Y})$ divides $F(\underline{X}, \underline{Y}) = f_1(\underline{X})f_2(\underline{Y}) - f_2(\underline{X})f_1(\underline{Y})$.
Furthermore, if h_1/h_2 is a reduced non-composite rational function then H is one of the irreducible factors with the smallest degree relatively to $\underline{X}$ of F.

The first part of this theorem is already known, see [30]. Here, we prove that H is irreducible if h_1/h_2 is non-composite. This result implies that we can modify the exponential time decomposition algorithm presented in [15] and get a polynomial time algorithm.

Comparison with other algorithms. There already exist several algorithms for the decomposition of rational functions. In [15], the authors provide two algorithms to decompose a multivariate rational function. These algorithms run in exponential time in the worst case. In the first one we have to factorize $f_1(\underline{X})f_2(\underline{Y}) - f_1(\underline{Y})f_2(\underline{X})$ and to look for factors of the following kind $h_1(\underline{X})h_2(\underline{Y}) - h_1(\underline{Y})h_2(\underline{X})$. The authors say that in the worst case the number of candidates to be tested is exponential in $d = \deg(f_1/f_2)$. In the last section we show that actually the number of candidates

is bounded by d. Thus we can get a polynomial time algorithm.

In the second algorithm, for each pair of factors (h_1, h_2) of f_1 and f_2 (i.e. h_1 divides f_1 and h_2 divides f_2), we have to test if there exists $u \in \mathbb{K}(T)$ such that $f_1/f_2 = u(h_1/h_2)$. Thus in the worst case we also have an exponential number of candidates to be tested.

To the author's knowledge, the first polynomial time algorithm is due to J.-M. Ollagnier, see [24] . This algorithm relies on the study of the kernel of the following derivation: $\delta_\omega(F) = \omega \wedge dF$, where $F \in \mathbb{K}[\underline{X}]$ and $\omega = f_2 df_1 - f_1 df_2$. In [24] the author shows that we can reduce the decomposition of a rational function to linear algebra. The bottleneck of this algorithm is the computation of the kernel of a matrix. The size of this matrix is $\mathcal{O}(d^n) \times \mathcal{O}(d^n)$, then the complexity of this deterministic algorithm belongs to $\mathcal{O}(d^{n\omega})$. In [24], as in this paper, the study of the pencil $\mu f_1 - \lambda f_2$ plays a crucial role.

Structure of this paper. In Section 1, we give a toolbox where we recall some results about decomposition and factorization. In Section 2, we describe our algorithms to decompose multivariate rational functions. In Section 3, we give an indecomposability test based on the study of a Newton's polytope. In Section 4, we give two algorithms to compute a Lüroth's generator. In Section 5 we show that the decomposition algorithm presented in [15] can be modified to get a polynomial time complexity algorithm.

Notations. All the rational functions are supposed to be reduced.

Given a polynomial f, $\deg(f)$ denotes its total degree.

$\overline{\mathbb{K}}$ is an algebraic closure of $\mathbb{K}$.

For the sake of simplicity, sometimes we write $\mathbb{K}[\underline{X}]$ instead of $\mathbb{K}[X_1, \ldots, X_n]$, for $n \geq 2$.

$Res(A, B)$ denotes the resultant of two univariate polynomials A and B.

For any polynomial $P \in \overline{\mathbb{K}}[\underline{X}]$, we write $\mathcal{U}(P) := \{a \in \mathbb{K}^n \mid P(a) \neq 0\}$.

1. Prerequisite

The following result implies, as mentioned in the introduction, that hypothesis (H) is not restrictive.

Lemma 6. *If f_1/f_2 is reduced in $\mathbb{K}(X_1, \ldots, X_n)$, where $n \geq 1$ and Λ is a variable, then $f_1 + \Lambda f_2$ is squarefree.*

Now we introduce our main tools.

Proposition 7. *Let $f = f_1/f_2$ be a rational function in $\mathbb{K}(X_1, \ldots, X_n)$.*

f is composite if and only if $\mu f_1 - \lambda f_2$ is reducible in $\overline{\mathbb{K}}[\underline{X}]$ for all $\mu, \lambda \in \overline{\mathbb{K}}$ such that $\deg(\mu f_1 - \lambda f_2) = \deg(f)$.

We also have: f is non-composite if and only if its spectrum $\sigma(f_1, f_2)$ is finite,

if and only if $f_1 - T f_2$ is absolutely irreducible in $\overline{\mathbb{K}(T)}[\underline{X}]$, where T is a new variable.

Furthermore if $\deg(f) = d$ then $\sigma(f_1, f_2)$ contains at most $d^2 - 1$ elements.

Proof. The first part of this result was known by Poincaré see [26], for a modern statement and a proof, see [7, Corollary 2.3].

The bound $|\sigma(h_1, h_2)| \leq d^2 - 1$ is proved for any field in the bivariate case in

[23]. We deduce the multivariate case easily thanks to the Bertini's irreducibility theorem, see e.g. [7] or the proof of Theorem 13 in [4] for an application of the Bertini's irreducibility theorem in this context.

$\square$

Lemma 8. *Let $h = h_1/h_2$ be a rational function in $\mathbb{K}(\underline{X})$, $u = u_1/u_2$ a rational function in $\mathbb{K}(T)$ and set $f = u \circ h$ with $f = f_1/f_2 \in \mathbb{K}(\underline{X})$. For all $\lambda \in \mathbb{K}$ such that $\deg(u_1 - \lambda u_2) = \deg u$, we have*

$$f_1 - \lambda f_2 = e(h_1 - t_1 h_2) \cdots (h_1 - t_k h_2)$$

where $e \in \mathbb{K}$, $k = \deg u$ and $t_i \in \overline{\mathbb{K}}$ are the roots of the univariate polynomial $u_1(T) - \lambda u_2(T)$.

Proof. See the proof of Lemma 39 in Section 5. Lemma 39 is a generalization of Lemma 8. We state Lemma 8 in our toolbox because the generalization will be only used in Section 5. $\square$

Remark 9. If $t_i \in \mathbb{K}$ then $h_1 - t_i h_2 \in \mathbb{K}[X_1, \ldots, X_n]$ is an irreducible factor of $f_1 - \lambda f_2$. Thus with a rational factorization we get information about the decomposition of f. This remark will be used during our probabilistic decomposition algorithm in order to avoid an absolute factorization.

2. DECOMPOSITION ALGORITHMS

2.1. **Computation of u.** Suppose that $f = f_1/f_2 = u \circ h \in \mathbb{K}(X_1, \ldots, X_n)$, $h \in \mathbb{K}(X_1, \ldots, X_n)$, and $u \in \mathbb{K}(T)$. We set $h = h_1/h_2$.
Usually, when h_1 and h_2 are given we get $u = u_1/u_2$ by solving a linear system, see [15, Corollary 2]. Let $\mathcal{M}(h_1, h_2)$ be the matrix corresponding to this linear system in the monomial basis. In our situation the size of $\mathcal{M}(h_1, h_2)$ is $\mathcal{O}(d^n) \times \mathcal{O}(d)$. Thus we can find u with $\tilde{\mathcal{O}}(d^{n+\omega-1})$ operations in $\mathbb{K}$.

We can get u with another approach. This approach is based on a strategy due to Zippel in [41]. Zippel showed in the univariate case that we can compute u quickly. His strategy is the following: compute the power series H such that $h \circ H(X) = X$, then compute $f \circ H$, and finally deduce u with a Padé approximant. All these steps can be done with $\tilde{\mathcal{O}}(d)$ or $\tilde{\mathcal{O}}(d^{3/2})$ arithmetic operations, see [8, Chapter 1], and [6]. Thus we deduce that in the univariate case, u can be computed with $\tilde{\mathcal{O}}(d^{3/2})$ arithmetic operations.
In the multivariate case with hypothesis (H), we have $\deg(f) = \deg_{X_n}(f)$. Thus $f(\underline{0}, X_n) = u \circ h(\underline{0}, X_n)$ is not a constant. Then we can apply Zippel's strategy to $f(\underline{0}, X_n)$ in order to find u. This method is correct because if f and h are given then there is a unique u such that $f = u \circ h$, see [15, Corollary 2]. Thus we have proved the following result:

Lemma 10. *Let $f, h \in \mathbb{K}(X_1, \ldots, X_n)$ be rational functions. We suppose that f satisfies hypothesis (H) and we set $\deg(f) = d$. If there exists $u \in \mathbb{K}(T)$ such that $f = u \circ h$ then we can compute u with $\tilde{\mathcal{O}}(d^n)$ arithmetic operations.*

Proof. We compute $f(\underline{0}, X_n)$ with $\tilde{\mathcal{O}}(d^n)$ arithmetic operations. Then we compute u as explained above with $\tilde{\mathcal{O}}(d^{3/2})$ arithmetic operations. $\square$

2.2. A probabilistic algorithm. Decomp

Input: $f = f_1/f_2 \in \mathbb{K}(X_1, \ldots, X_n)$, $z := (\underline{a}, \underline{b}) \in \mathbb{K}^{2n}$.

Output: A decomposition of f if it exists, with $f = u \circ h$, $u = u_1/u_2$, $h = h_1/h_2$ non-composite and $\deg u \geq 2$.

(1) We set $F_a = f_2(\underline{a})f_1(\underline{X}) - f_1(\underline{a})f_2(\underline{X})$, $F_b = f_2(\underline{b})f_1(\underline{X}) - f_1(\underline{b})f_2(\underline{X})$.

(2) Factorize F_a and F_b.

(3) If F_a or F_b is irreducible then Return "r is non-composite".

(4) Let $\mathcal{F}_a$ (resp. $\mathcal{F}_b$) be an irreducible factor of F_a (resp. F_b) with the smallest degree.

(5) Set $h = \mathcal{F}_a/\mathcal{F}_b$.

(6) Compute u such that $f = u \circ h$ as explained in Section 2.1.

(7) Return u, h.

Exemple 11.

a- We consider $f = f_1/f_2$, with $f_1 = X^3 + Y^3 + 1$ and $f_2 = 3XY$. We set $\underline{a} = (0, 0)$, $\underline{b} = (0, 1)$. Then $F_a = -3XY$ and $F_b = 3X^3 + 3Y^3 - 6XY + 3$. F_a is reducible but F_b is irreducible then we conclude that f is non-composite.

b- Now, we apply the algorithm Decomp to the rational function $f = u \circ h$, where $u = (T^2 + 1)/T$ and $h = h_1/h_2$ with $h_1 = X^3 + Y^3 + 1$ and $h_2 = 3XY$. We have seen above that h is non-composite.

In this situation with $\underline{a} = (0, 0)$ and $\underline{b} = (0, 1)$ we get:

$$F_a = -3.X.Y.(X^3 + Y^3 + 1), \text{ and}$$

$$F_b = -12.X.Y.(X^3 + Y^3 + 1).$$

Then the algorithm cannot give a correct output in this situation. Here, we have $f_2(\underline{a}) = f_2(\underline{b})$, we will see that we must avoid this situation. If we set $\underline{a} = (2, 1)$ and $\underline{b} = (1, -1)$ then:

$$F_a = 60.(X^3 + Y^3 - 5XY + 1).(X^3 + Y^3 - \frac{3}{5}XY + 1), \text{ and}$$

$$F_b = -3.(X^3 + Y^3 + XY + 1).(X^3 + Y^3 + 3XY + 1).$$

Then we get $\mathcal{F}_a = X^3 + Y^3 - 5XY + 1$ and $\mathcal{F}_b = X^3 + Y^3 + XY + 1$. The algorithm Decomp returns $h = \mathcal{F}_a/\mathcal{F}_b$. This is a correct output since $U \circ \mathcal{F}_a/\mathcal{F}_b = h_1/h_2$, where $U = (T/6 + 5/6)/(-T/2 + 1/2)$.

Proposition 12. *If $\underline{a}, \underline{b}$ are not the roots of some non-zero polynomials then the algorithm corresponding to $z = (\underline{a}, \underline{b})$ is correct.*

Proof. First, we suppose that f is non-composite and we set

$$\text{Spect}_{f_1, f_2}(T_1, T_2) = \prod_{(\mu:\lambda) \in \sigma(f_1, f_2)} (\mu T_2 - \lambda T_1).$$

We have $\text{Spect}_{f_1, f_2}(\mu, \lambda) = 0$ if and only if $(\mu : \lambda) \in \sigma(f_1, f_2)$.

If $\text{Spect}_{f_1, f_2}(f_2(\underline{a}), f_1(\underline{a})).\text{Spect}_{f_1, f_2}(f_2(\underline{b}), f_1(\underline{b})) \neq 0$ then F_a and F_b are absolutely irreducible and $\deg F_a = \deg F_b = \deg f$.

This gives: if $\underline{a}$ and $\underline{b}$ avoid the roots of

$$S(\underline{A}, \underline{B}) := \text{Spect}_{f_1, f_2}(f_2(\underline{A}), f_1(\underline{A})).\text{Spect}_{f_1, f_2}(f_2(\underline{B}), f_1(\underline{B})),$$

where $\deg S \leq 2d(d^2 - 1)$ by Proposition 7, then the algorithm returns: "r is non-composite".

Second, we suppose $f = v \circ H$, with $H \in \mathbb{K}(X_1, \ldots, X_n)$ a non-composite rational function. We set $v = v_1/v_2$, $H = H_1/H_2$ such that these two rational functions are reduced. We also suppose that $f_2(\underline{a})$ and $f_2(\underline{b})$ are nonzero.
If $\deg F_a = \deg F_b = \deg f$ then $\underline{a}$ and $\underline{b}$ are not the roots of a polynomial D of degree d. Thanks to Lemma 8 we have:

$$F_a = e(H_1 - t_1 H_2) \cdots (H_1 - t_k H_2),$$

$$F_b = e'(H_1 - s_1 H_2) \cdots (H_1 - s_k H_2),$$

with $e, e' \in \mathbb{K}$, $t_i, s_j \in \overline{\mathbb{K}}$.
As $H_1(\underline{a})/H_2(\underline{a})$ (resp. $H_1(\underline{b})/H_2(\underline{b})$) is a root of $f_2(\underline{a})v_1(T) - f_1(\underline{a})v_2(T)$ (resp. $f_2(\underline{b})v_1(T) - f_1(\underline{b})v_2(T)$), we set $t_1 = H_1(\underline{a})/H_2(\underline{a})$ and $s_1 = H_1(\underline{b})/H_2(\underline{b})$, and we remark that $t_1, s_1 \in \mathbb{K}$. We set

$$\text{Spect}_{H_1, H_2}(T) = \prod_{\lambda \in \sigma(H_1, H_2) \cap \mathbb{K}} (T - \lambda).$$

If $\text{Spect}_{H_1, H_2}(t_1) \neq 0$ (resp. $\text{Spect}_{H_1, H_2}(s_1) \neq 0$) then $H_1 - t_1 H_2$ (resp. $H_1 - s_1 H_2$) is absolutely irreducible.
If

$$R(\underline{a}, \underline{b}) = Res_T\big(f_2(\underline{a})v_1(T) - f_1(\underline{a})v_2(T),\ f_2(\underline{b})v_1(T) - f_1(\underline{b})v_2(T)\big) \neq 0$$

then $t_i \neq s_j$ for all i, j. We remark that R is a nonzero polynomial by Lemma 6 since v_1 and v_2 are coprime. Thus step 4 gives $\mathcal{F}_a = H_1 - tH_2$, $\mathcal{F}_b = H_1 - sH_2$ with $t, s \in \mathbb{K}$ and $t \neq s$. Then $h = \mathcal{F}_a/\mathcal{F}_b$ is non-composite, because H_1/H_2 is non-composite. $\square$

Remark 13. Now, with the notations of the previous proof, we can explain in details the meaning of: "*If $\underline{a}, \underline{b}$ are not the roots of some non-zero polynomials*" in Proposition 12 and Theorem 2. This means:
If f is non-composite then there exists a nonzero polynomial

$$P(\underline{A}, \underline{B}) := S(\underline{A}, \underline{B})$$

of degree at most $2d(d^2 - 1)$ such that for any $(\underline{a}, \underline{b}) \in \mathcal{U}(P)$ the algorithm corresponding to z is executable and returns a correct output.
If f is composite then there exists a nonzero polynomial

$$D_1(\underline{A}, \underline{B}) := f_2(\underline{A}).f_2(\underline{B}).D(\underline{A}).D(\underline{B})$$

of degree at most $4d$ such that;
for any $(\underline{a}, \underline{b}) \in \mathcal{U}(D_1)$, there exist nonzero polynomials

$$D_2(\underline{A}) := \prod_{\lambda \in \sigma(H_1, H_2) \cap \mathbb{K}} \big(H_2(\underline{A}) - \lambda H_1(\underline{A})\big)$$

of degree at most $(d^2 - 1).d/2$, and

$$R(\underline{A}, \underline{B})$$

where $\deg_{\underline{A}} R \leq d^2/2$ and $\deg_{\underline{B}} R \leq d^2/2$, such that; for any $(\underline{a}, \underline{b}) \in \mathcal{U}\big(D_2(\underline{A}).D_2(\underline{B}).R(\underline{A}, \underline{B})\big)$, the algorithm corresponding to $z = (\underline{a}, \underline{b})$ is executable and returns a correct output.

Proposition 14. *Under hypotheses (C) and (H), if $\underline{a}$ and $\underline{b}$ are not the roots of a non-zero polynomial then we can use the algorithm proposed in [22]. Then the algorithm* **Decomp** *performs one factorization of a univariate polynomial of degree d over $\mathbb{K}$ plus a number of operations in $\mathbb{K}$ belonging to $\tilde{\mathcal{O}}(d^n)$ if $n \geq 3$ or to $\tilde{\mathcal{O}}(d^3)$ if $n = 2$.*

Proof. As f satisfies (H,i), we deduce that if $\underline{a}$ and $\underline{b}$ are not the roots of a polynomial D of degree d, then the monic part relatively to X_n of F_a (resp. F_b) satisfies (L,i).
We set:

$$\mathcal{D}(\Lambda) = Res_{X_n}\big(f_1(\underline{0}, X_n) - \Lambda f_2(\underline{0}, X_n), \, \partial_{X_n} f_1(\underline{0}, X_n) - \Lambda \partial_{X_n} f_2(\underline{0}, X_n)\big).$$

By hypothesis (H,ii), $\mathcal{D}(\Lambda) \neq 0$ in $\mathbb{K}[\Lambda]$. Furthermore if $f_2(\underline{a})$ and $f_2(\underline{b})$ are nonzero and $\mathcal{D}\big(f_1(\underline{a})/f_2(\underline{a})\big) \neq 0$ (resp. $\mathcal{D}\big(f_1(\underline{b})/f_2(\underline{b})\big) \neq 0$) then hypothesis (L,$ii$) is satisfied for F_a (resp. F_b). Then we can use Lecerf's algorithm, see [22]. This gives: if $\underline{a}$ and $\underline{b}$ avoid the roots of

$$\overline{\mathcal{D}}(\underline{A}, \underline{B}) = \mathcal{D}\big(f_1(\underline{A})/f_2(\underline{A})\big).\mathcal{D}\big(f_1(\underline{B})/f_2(\underline{B})\big).\big(f_2(\underline{A}).f_2(\underline{B})\big)^{\deg \mathcal{D}+1},$$

and $\deg \overline{\mathcal{D}} \leq 2\big(d(d-1)d+d\big)$ then we can use the algorithm proposed by G. Lecerf in [22].
The complexity result comes from Lemma 10, and [22, Proposition 5], [21, Proposition 2] and [22, Errata]. $\qquad\square$

Remark 15. The meaning of the condition *"if $\underline{a}$ and $\underline{b}$ are not the roots of a non-zero polynomial"* in Proposition 14 is the following: If we want to use Lecerf's factorization algorithm in order to get the complexity estimate given in the second part of Theorem 2, then $\underline{a}$ and $\underline{b}$ must also avoid the roots of the polynomial

$$D(\underline{A}).D(\underline{B}).\overline{\mathcal{D}}(\underline{A}, \underline{B}),$$

where $\deg D \leq d$ and $\deg \overline{\mathcal{D}} \leq 2(d^2(d-1)+d)$.

It follows that Theorem 2 comes from Proposition 12 and Proposition 14.

2.3. **A deterministic algorithm. Decomp Det**
Input: $f = f_1/f_2 \in \mathbb{K}(X_1, \ldots, X_n)$, $S = \{s_0, \ldots, s_{\mathcal{B}}\}$ a subset of $\mathbb{K}$ with at least $\mathcal{B} + 1 = \max(d^2, \frac{3}{2}d^2 - 2d + 1)$ distinct elements.
Output: A decomposition of f if it exists, with $f = u \circ h$, $u = u_1/u_2$, $h = h_1/h_2$ non-composite and $\deg u \geq 2$.

t:=false, $\lambda := 0$.
While t=false do

    (1) If $\deg(f_1 + s_\lambda f_2) = \deg(f)$ then go to step 2 else $\lambda := \lambda + 1$.
    (2) Compute the absolute factorization of $F_\lambda := f_1 + s_\lambda f_2$.
    (3) If F_λ is absolutely irreducible then Return "f is non-composite".
    (4) If F_λ is absolutely reducible then
        (a) If two distinct absolute irreducible factors f_1, f_2 belong to $\mathbb{K}[\underline{X}]$ then we set $h_1 := \mathsf{f}_1$ and $h_2 := \mathsf{f}_2$,
            If there exists an absolute irreducible factor $\mathsf{f}_1 := \mathcal{F}_1 + \epsilon \mathcal{F}_2$, with $\epsilon \in \overline{\mathbb{K}} \setminus \mathbb{K}$ and $\mathcal{F}_1, \mathcal{F}_2 \in \mathbb{K}[\underline{X}]$ then we set $h_1 := \mathcal{F}_1$, $h_2 := \mathcal{F}_2$,
            Else $\lambda := \lambda + 1$ and go to step 1.

(b) Compute u (if it exists) such that $f = u \circ h$ as explained in Section 2.1.

(c) If u exists then t:=true else $\lambda := \lambda + 1$.

Return u, h.

Exemple 16.

a- We consider $f = f_1/f_2$, where $f_1 = 3XY$ and $f_2 = X^3 + Y^3 + 1$. This gives $F_0 = 3.X.Y$, then F_0 is reducible, and this gives $h = X/Y$. We do not find a rational function u such that $f = u \circ (X/Y)$ then we consider $F_1 = f_1 + f_2$. F_1 is absolutely irreducible, then the algorithm Decomp Det returns f is non-composite.

b- Now, we apply the algorithm Decomp Det to the rational function $f = u \circ h$, where $u = (T^2 + 1)/T$ and $h = (X^3 + Y^3 + 1)/(3XY)$. As we have seen above h is non-composite.

In this situation we have:

$$F_0 = (X^3 + Y^3 + 1 + 3.i.X.Y)(X^3 + Y^3 + 1 - 3.i.X.Y),$$

where $i^2 = -1$.

Then we have $f_1 = X^3 + Y^3 + 1 + 3.i.X.Y$, $\mathcal{F}_1 = X^3 + Y^3 + 1$, $\mathcal{F}_2 = 3XY$. The algorithm returns $\mathcal{F}_1/\mathcal{F}_2 = h$.

Proposition 17. *The algorithm is correct. Furthermore we go back to step 1 at most $\mathcal{O}(d^2)$ times.*

Proof. First, we suppose that f is non-composite. By Proposition 7 there exists $s_{\lambda_0} \in S$ such that $s_{\lambda_0} \notin \sigma(f_1, f_2)$ because S contains at least d^2 elements. Thus $f_1 + s_{\lambda_0} f_2$ is absolutely irreducible and step 3 returns f non-composite.

We remark that if $f_1 + s_\lambda f_2$ is reducible then we cannot find u during step 4b because f is non-composite. Then if f is non-composite the algorithm is correct.

Second, we suppose that f is composite and $f = v \circ H$ with $H = H_1/H_2$ a reduced and non-composite rational function, $\deg v \geq 2$ and $v = v_1/v_2$ is a reduced rational function.

$f_1 + s_\lambda f_2 = e \prod_i (H_1 + t_i H_2)$ by Lemma 8, where $(v_1 + s_\lambda v_2)(t_i) = 0$.

There exists $s_{\lambda_0} \in S$ such that $D(s_{\lambda_0}) \neq 0$, where

$$D(\Lambda) = Res(v_1 + \Lambda v_2, v_1' + \Lambda v_2') \times \prod_{x_i \in \sigma(H_1, H_2) \cap \mathbb{K}} \big(v_2(x_i) - \Lambda v_1(x_i)\big).$$

Indeed $D(\Lambda)$ is a nonzero polynomial by Lemma 6 since v_1 and v_2 are coprime. Furthermore, by Proposition 7, we have

$$\deg D \leq \deg v(\deg v - 1) + \big((\deg H)^2 - 1\big).\deg v.$$

As $\deg v . \deg H = d$ and $\deg v \geq 2$, we get

$$\deg D \leq 3/2 d^2 - 2d.$$

As S contains at least $3/2 d^2 - 2d + 1$ distinct elements, there exists $s_{\lambda_0} \in S$ such that $D(s_{\lambda_0}) \neq 0$ and then for all i, $t_i \notin \sigma(H_1, H_2)$, and $t_i \neq t_j$ for all $i \neq j$.

Then for λ_0 we construct h_1 and h_2 as explained in step 4a. (If $t_1, t_2 \in \mathbb{K}$ are distinct then we have two absolutely irreducible factors in $\mathbb{K}[\underline{X}]$, else if $t_1 \in \overline{\mathbb{K}} \setminus \mathbb{K}$ then we construct h_1 and h_2 with only one absolutely irreducible factor.) We have

$h_1/h_2 = w \circ H_1/H_2$ where $w \in \mathbb{K}(T)$ and $\deg w = 1$.
We remark that if f is composite then we find a decomposition $f = u \circ h$ with h non-composite. Indeed, there exist $(\mu : \lambda)$ and $(\mu' : \lambda') \neq (\mu : \lambda) \in \mathbb{P}^1_{\mathbb{K}}$ such that $\mu h_1 + \lambda h_2$ and $\mu' h_1 + \lambda' h_2$ are absolutely irreducible. (It is obvious if $t_1, t_2 \in \mathbb{K}$. If $t_1 \in \overline{\mathbb{K}} \setminus \mathbb{K}$ there exists a conjugate t'_1 of t_1 over $\mathbb{K}$ such that $h_1 + t'_1 h_2$ is absolutely irreducible.) Then h_1/h_2 is non composite by Proposition 7. Thus if f is non-composite the output is correct.   □

Theorem 3 is a direct corollary of Proposition 17.

Remark 18. In [10] the authors show that we can compute, under the hypothesis (C), the absolute factorization of a bivariate squarefree polynomial with at most $\tilde{\mathcal{O}}(d^4)$ arithmetic operations. As we go back to step 1 at most $\mathcal{O}(d^2)$ times we deduce that the algorithm Decomp Det uses at most $\tilde{\mathcal{O}}(d^6)$ arithmetic operations. When $n \geq 3$, a complexity analysis of an absolute factorization algorithm as studied in [10] is not done, but we can estimate the cost of our deterministic algorithm. Indeed, we can reduce absolute factorization to factorization over a suitable algebraic extension $\mathbb{K}[\alpha]$ of degree at most d over $\mathbb{K}$, [34, 35, 12, 19]. With this strategy and with the deterministic factorization algorithm proposed in [22] we get an absolute factorization algorithm which performs at most $\tilde{\mathcal{O}}(d^{n+\omega-1})$ arithmetic operations in $\mathbb{K}[\alpha]$. Thus the algorithm performs $\tilde{\mathcal{O}}(d^{n+\omega})$ arithmetic operations in $\mathbb{K}$, because $[\mathbb{K}[\alpha] : \mathbb{K}] \leq d$. As we go back to step 1 at most $\mathcal{O}(d^2)$ times we deduce that, if we can use Lecerf's deterministic factorization algorithm, the algorithm Decomp Det uses at most $\tilde{\mathcal{O}}(d^{n+\omega+2})$ arithmetic operations and one factorization of a univariate polynomial of degree d with coefficients in $\mathbb{K}[\alpha]$.

3. An indecomposability test using Newton's polytope

In Section 2, if f_1 and f_2 are sparse our algorithms do not use this information. In this section we give an indecomposability test based on some properties of the Newton's polytope. The idea is to generalize this remark: if $\deg f$ is a prime integer then f is non-composite. This is obvious because $f = u \circ h$ implies $\deg f = \deg u. \deg h$, and $\deg u \geq 2$.

Definition 19. Let $f(\underline{X}) \in \mathbb{K}[X_1, \ldots, X_n]$, the support of $f(\underline{X})$ is the set S_f of integer points $(i_1, \ldots, i_n)$ such that the monomial $X_1^{i_1} \cdots X_n^{i_n}$ appears in f with a nonzero coefficient.
We denote by $N(f)$ the convex hull (in the real space $\mathbb{R}^n$) of S_f. This set $N(f)$ is called the Newton's polytope of f.

Definition 20. We set $N(f_1/f_2) = N(f_1 - \Lambda f_2)$ where Λ is a variable, and where $f_1 - \Lambda f_2$ is considered as a polynomial with coefficients in $\mathbb{K}[\Lambda]$.

Remark 21. As Λ is a variable $N(f_1 - \Lambda f_2)$ is the convex hull of $S_{f_1} \cup S_{f_2}$.

We recall the classical Bertini-Krull's theorem in our context, see [31, Theorem 37].

Theorem 22. *(Bertini-Krull) Let f_1/f_2 a reduced rational function. Then the following conditions are equivalent:*

(1) f_1/f_2 is composite,

(2) (a) either there exist h_1, $h_2 \in \overline{\mathbb{K}}[\underline{X}]$ with $\deg_X f_1(\underline{X}) - \Lambda f_2(\underline{X}) > \max(\deg h_1, \deg h_2)$ and $a_i(\Lambda) \in \overline{\mathbb{K}}[\Lambda]$, such that

$$f_1(\underline{X}) - \Lambda f_2(\underline{X}) = \sum_{i=0}^{e} a_i(\Lambda) h_1(\underline{X})^i h_2(\underline{X})^{e-i};$$

(b) or the characteristic p of $\mathbb{K}$ is positive and $f_1(\underline{X}) - \Lambda f_2(\underline{X}) \in \overline{\mathbb{K}}[\Lambda][X_1^p, \ldots, X_n^p]$.

Lemma 23. *If f_1/f_2 is a composite rational function and the characteristic p of $\mathbb{K}$ is such that $p = 0$ or $p > d$, then there exist $e \in \mathbb{N}$, $h_1, h_2 \in \mathbb{K}[\underline{X}]$ such that $N(f_1/f_2) = eN(h_1/h_2)$.*

Proof. By Theorem 22 we have $f_1(\underline{X}) - \Lambda f_2(\underline{X}) = \sum_{i=0}^{e} a_i(\Lambda) h_1(\underline{X})^i h_2(\underline{X})^{e-i}$. We denote by $u(\Lambda, \chi)$ the polynomial

$$u(\Lambda, \chi) = \sum_{i=0}^{e} a_i(\Lambda) \chi^i = a_e(\Lambda) \prod_{i=1}^{e} (\chi - \varphi_i(\Lambda)),$$

where $\varphi_i(\Lambda) \in \overline{\mathbb{K}(\Lambda)}$.
Thus

$$f_1(\underline{X}) - \Lambda f_2(\underline{X}) = a_e(\Lambda) \prod_{i=1}^{e} \left(h_1(\underline{X}) - \varphi_i(\Lambda) h_2(\underline{X}) \right).$$

All the factors $h_1(\underline{X}) - \varphi_i(\Lambda) h_2(\underline{X}) \in \overline{\mathbb{K}(\Lambda)}[\underline{X}]$ have the same support.
Indeed, if we suppose the converse then there exist a coefficient $c_1 \in \overline{\mathbb{K}}$ of h_1 and a coefficient $c_2 \in \overline{\mathbb{K}}$ of h_2 and two indices i and j such that:

$$c_1 - \varphi_i(\Lambda) c_2 = 0, \qquad c_1 - \varphi_j(\Lambda) c_2 \neq 0.$$

Then $c_2 \neq 0$ and $\varphi_i(\Lambda) = c_1/c_2 \in \overline{\mathbb{K}}$. Thus $h_1 - \varphi_i(\Lambda) h_2 \in \overline{\mathbb{K}}[\underline{X}]$ is a factor of $f_1(\underline{X}) - \Lambda f_2(\underline{X})$. This implies $f_1(\underline{X}) - \Lambda f_2(\underline{X})$ is reducible in $\overline{\mathbb{K}}[\Lambda][\underline{X}]$. This is impossible because f_1 and f_2 are coprime.
Then, for all $i = 1, \ldots, e$, we have:

$$N\left(h_1 - \varphi_i(\Lambda) h_2 \right) = N(h_1 - \Lambda h_2) = N(h_1/h_2).$$

We recall that $F = F_1.F_2$ implies $N(F) = N(F_1) + N(F_2)$, see for example [14, Lemma 5], where the sum is the Minkowski's sum of convex sets. Thus we have:

$$N(f_1/f_2) = N(f_1 - \Lambda f_2) = \sum_{i=1}^{e} N\left(h_1 - \varphi_i(\Lambda) h_2 \right) = \sum_{i=1}^{e} N(h_1/h_2) = eN(h_1/h_2).$$

This is the desired result. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \square$

The previous lemma says that if f is composite then all the vertices of $N(f)$ have a common factor: e. This gives our indecomposability test designed for sparse polynomials f_1 and f_2:

Corollary 24 (Indecomposability test). *Let p be the characteristic of $\mathbb{K}$, and $p = 0$ or $p > d$.*
Let $(i_1^{(1)}, \ldots, i_n^{(1)}), \ldots, (i_1^{(k)}, \ldots, i_n^{(k)})$ be the vertices of $N(f)$.
If $\gcd(i_1^{(1)}, \ldots, i_n^{(1)}, \ldots, i_1^{(k)}, \ldots, i_n^{(k)}) = 1$ then f is non-composite.

4. Computation of a Lüroth's generator

In this section we show how to compute a Lüroth's generator. We give two algorithms. The first one follows the strategy proposed in [32] for univariate rational functions. The second one uses the algorithm Decomp and the computation of a greatest common right component of a univariate rational function.

4.1. **Generalization of Sederberg's algorithm.** In this subsection, we generalize Sederberg's algorithm. Sederberg's algorithm, see [32], is a probabilistic algorithm to compute a Lüroth's generator in the univariate case. Here, we show that the same strategy works in the multivariate case. Our algorithm is also a kind of probabilistic version of the algorithm presented in [15]. Indeed, here we compute gcd of polynomials of the following kind $f_2(\underline{a})f_1(\underline{X}) - f_1(\underline{a})f_2(\underline{X})$, where $\underline{a} \in \mathbb{K}^n$. In [15], the authors compute gcd of polynomials of the following kind $f_2(\underline{Y})f_1(\underline{X}) - f_1(\underline{Y})f_2(\underline{X})$, where $\underline{Y}$ are new independent variables.

Sederberg Generalized
Input: $f(\underline{X}) = f_1/f_2(\underline{X})$, $g(\underline{X}) = g_1/g_2(\underline{X}) \in \mathbb{K}(X_1, \ldots, X_n)$ two reduced rational functions, $\underline{a}, \underline{b} \in \mathbb{K}^n$, $n \geq 2$.
Output: $h(\underline{X}) \in \mathbb{K}(\underline{X})$ such that $\mathbb{K}(f, g) = \mathbb{K}(h)$, if h exists.

(1) $F_a := f_2(\underline{a})f_1(\underline{X}) - f_1(\underline{a})f_2(\underline{X})$, $G_a := g_2(\underline{a})g_1(\underline{X}) - g_1(\underline{a})g_2(\underline{X})$.
   $H_a := \gcd(F_a, G_a)$.
   If H_a is constant then Return "No Lüroth's generator", else go to 2.
(2) $F_b := f_2(\underline{b})f_1(\underline{X}) - f_1(\underline{b})f_2(\underline{X})$, $G_b := g_2(\underline{b})g_1(\underline{X}) - g_1(\underline{b})g_2(\underline{X})$.
   $H_b := \gcd(F_b, G_b)$.
   If H_b is constant then Return "No Lüroth's generator", else go to 3.
(3) Return $h := H_a/H_b$.

Exemple 25.   a- We set $f = X$, and $g = Y$, $\underline{a} = (0,0)$, $\underline{b} = (1,0)$. Thus $F_a = X$, $G_a = Y$ and $H_a = 1$. The algorithm Sederberg Generalized gives $\mathbb{K}(f,g) = \mathbb{K}(X, Y)$ has "No Lüroth's generator".

   b- We consider $f = U \circ h$ and $g = V \circ h$ where $h = (X^3 + Y^3 + 1)/(3XY)$, $U = T^2/(T+1)$, $V = (T+2)/(T^3 + 3)$. h is a non-composite rational function.
   We set $\underline{a} = (0,0)$, $\underline{b} = (2,1)$. In this situation we have:

$$H_a = 3XY, \text{ and } H_b = 12.(X^3 + Y^3 - 5XY + 1).$$

   The algorithm Sederberg Generalized returns H_a/H_b. This is a correct output because $\mathbb{K}(f,g) = \mathbb{K}(h)$ and $h = u \circ (H_a/H_b)$ where u is the rational function $u = (20T + 1)/(12T)$.
   Now, if we set $\underline{a} = (0,0)$, $\underline{b} = (0,1)$ then we get $H_a = 3XY$ and $H_b = 12XY$. In this situation the output H_a/H_b is not correct. We are in a situation where $h(\underline{a}) = h(\underline{b})$ and we will see that we must avoid this situation.

Proposition 26. *There exists an open Zariski set $U \subset \mathbb{K}^{2n}$ related to f_1 and f_2, such that for all $(\underline{a}, \underline{b}) \in U$ the tree corresponding to $(\underline{a}, \underline{b})$ is executable on f, g and returns (if it exists) h such that $\mathbb{K}(h) = \mathbb{K}(f, g)$.*

In order to prove this proposition we recall some results.

Definition 27. Given $f_1, \ldots, f_m \in \mathbb{K}(\underline{X})$, we say that they have a common right component (CRC) h, if there are rational functions $u_i \in \mathbb{K}(T)$, $i = 1, \ldots, m$, such that $f_i = u_i \circ h$, and $\deg u_i > 1$.

h is a greatest common right component (GCRC) of $f_1, \ldots, f_m$ if the $u_i's$ have not a common right component of degree greater than one.

Proposition 28. $\mathbb{K}(f_1, \ldots, f_m) = \mathbb{K}(h)$ *if and only if* h *is a GCRC of* $f_1, \ldots, f_m$.

Proof. This proposition is proved in the univariate case in [1] but the proof can be extended to the multivariate case in a straightforward way. $\square$

Proof of Proposition 26. Firstly, we suppose that there exists a Lüroth's generator $h = h_1/h_2$, where h_1/h_2 is reduced. Then, by Proposition 28, $f = u \circ h$ and $g = v \circ h$ where $u, v \in \mathbb{K}(T)$ do not have a common right component of degree greater than one. Thus $\mathbb{K}(u(T), v(T)) = \mathbb{K}(T)$. Then there exist $Q_1, Q_2 \in \mathbb{K}[U, V]$ such that $Q_1(u(T), v(T))/Q_2(u(T), v(T)) = T$.

Furthermore by Lemma 8,

$$F_a = f_2(\underline{a})f_1(\underline{X}) - f_1(\underline{a})f_2(\underline{X}) = e \prod_i \big(h_1(\underline{X}) - t_i h_2(\underline{X})\big)$$

where $e \in \mathbb{K}$ and t_i are the roots of

$$f_2(\underline{a})u_1(T) - f_1(\underline{a})u_2(T) =: u_a,$$

and

$$G_a = g_2(\underline{a})g_1(\underline{X}) - g_1(\underline{a})g_2(\underline{X}) = e' \prod_i \big(h_1(\underline{X}) - s_i h_2(\underline{X})\big)$$

where $e' \in \mathbb{K}$ and s_i are the roots of

$$g_2(\underline{a})v_1(T) - g_1(\underline{a})v_2(T) =: v_a.$$

We get: $h(\underline{a})$ is a common root of u_a and v_a. Thus $h_1(\underline{X}) - h(\underline{a})h_2(\underline{X})$ divides F_a and G_a.

If $f_2(\underline{a}).g_2(\underline{a}).Q_2\big(u(h(\underline{a})), v(h(\underline{a}))\big) \neq 0$ then $h(\underline{a})$ is the unique common root of u_a and v_a. Indeed if there exists another root x such that $u_a(x) = v_a(x) = 0$, then $u(h(\underline{a})) = f_1(\underline{a})/f_2(\underline{a}) = u(x)$ and $v(h(\underline{a})) = g_1(\underline{a})/g_2(\underline{a}) = v(x)$.

It follows:

$$h(\underline{a}) = \frac{Q_1\big(u(h(\underline{a})), v(h(\underline{a}))\big)}{Q_2\big(u(h(\underline{a})), v(h(\underline{a}))\big)} = \frac{Q_1\big(u(x), v(x)\big)}{Q_2\big(u(x), v(x)\big)} = x.$$

Now we remark that if $t \neq s$ then $\gcd(h_1 + th_2, h_1 + sh_2)$ is constant.

We get then: $\gcd(F_a, G_a) = h_1(\underline{X}) - h(\underline{a})h_2(\underline{X})$.

In the same way: $\gcd(F_b, G_b) = h_1(\underline{X}) - h(\underline{b})h_2(\underline{X})$.

If $h(\underline{a}) \neq h(\underline{b})$, this gives the desired result, because $\mathbb{K}(h) = \mathbb{K}(H)$ when $H = U \circ h$ with $U = \big(T - h(\underline{a})\big)/\big(T - h(\underline{b})\big)$.

Secondly, we suppose that there does not exist a Lüroth's generator.

Then we have $f = u \circ h$ and $g = v \circ H$, with $h, H \in \mathbb{K}(\underline{X})$ non-composite and algebraically independent.

Thus $F_a(\underline{X}) = e. \prod_i \big(h_1(\underline{X}) - t_i h_2(\underline{X})\big)$ as before, with $h_1(\underline{X}) - t_i h_2(\underline{X})$ absolutely irreducible if $t_i \notin \sigma(h_1, h_2)$. The condition $t_i \notin \sigma(h_1, h_2)$ means

$$R(\underline{a}) = Res_T\big(f_2(\underline{a})u_1(T) - f_1(\underline{a})u_2(T), \mathrm{Spect}_{h_1, h_2}(T)\big) \neq 0,$$

where $\text{Spect}_{h_1,h_2}(T) = \prod_{\lambda \in \sigma(h_1,h_2) \cap \mathbb{K}}(T - \lambda)$.

In the same way, we have $G_a = e' . \prod_i (H_1(\underline{X}) - s_i H_2(\underline{X}))$ with $H_1(\underline{X}) - s_i H_2(\underline{X})$ absolutely irreducible if

$$S(\underline{a}) = Res_T\big(g_2(\underline{a})v_1(T) - g_1(\underline{a})v_2(T), \text{Spect}_{H_1,H_2}(T)\big) \neq 0.$$

Thus F_a and G_a have a non trivial common divisor if and only if there exist t_i, s_j and $\alpha \in \mathbb{K} \setminus \{0\}$ such that:

$$(\star)\, \alpha\big(h_1(\underline{X}) - t_i h_2(\underline{X})\big) = H_1(\underline{X}) - s_j H_2(\underline{X}).$$

In the same way, F_b and G_b have a non trivial common divisor if and only if there exists t'_i, s'_j and $\alpha' \in \mathbb{K} \setminus \{0\}$ such that:

$$(\star\star)\, \alpha'\big(h_1(\underline{X}) - t'_i h_2(\underline{X})\big) = H_1(\underline{X}) - s'_j H_2(\underline{X}).$$

$(\star)$ and $(\star\star)$ give:

$$\begin{pmatrix} \alpha & -\alpha t_i \\ \alpha' & -\alpha' t'_i \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} 1 & -s_j \\ 1 & -s'_j \end{pmatrix} \begin{pmatrix} H_1 \\ H_2 \end{pmatrix}.$$

If

$$D(\underline{a},\underline{b}) = Res_T\big(g_2(\underline{a})v_1(T) - g_1(\underline{a})v_2(T),\, g_2(\underline{b})v_1(T) - g_1(\underline{b})v_2(T)\big) \neq 0$$

then $s_j \neq s'_j$ and the previous system gives $H = u \circ h$, with $\deg u = 1$. Thus h and H are algebraically dependent and this is absurd. Thus F_a and G_a (resp. F_b and G_b) have no common divisor.

Hence, if no Lüroth's generator exists and $f_2(\underline{a}).g_2(\underline{b}).R(\underline{a}).S(\underline{a}).R(\underline{b}).S(\underline{b}).D(\underline{a},\underline{b})$ is not equal to zero, then $\gcd(F_a, G_a)$ is constant and $\gcd(F_b, G_b)$ is constant. Thus the algorithm returns "No Lüroth's generator". $\square$

Remark 29. With the notations of the previous proof, we remark that $\underline{a}$ and $\underline{b}$ must avoid the roots of: $f_2(\underline{X}), g_2(\underline{X}), h_2(\underline{X}), Q_2\big(f(\underline{X})\big), g(\underline{X})\big), R(\underline{X}), S(\underline{X})$, and $(\underline{a},\underline{b})$ must avoid the roots of $h_1(\underline{A})h_2(\underline{B}) - h_1(\underline{B})h_2(\underline{A})$ and $D(\underline{A},\underline{B})$.

We can easily bound the degree of each polynomial: $\deg f_i \leq d$, $\deg g_i \leq d$, $\deg h_i \leq d/2$, $\deg Q_2 \leq d(d-1)$ see [5, Proposition 2.1], $\deg R \leq d(d^2-1)$, $\deg S \leq d(d^2-1)$, and $\deg D \leq d^3$.

Then if $\mathbb{K}$ is "big enough" the open Zariski set U is not the empty set.

Remark 30. In the algorithm Sederberg Generalized we cannot consider two random linear combinations of f_1, f_2 and g_1, g_2. Indeed, with random linear combinations and with the notations of the previous proof, u_a and v_a do not have a unique common root in $\mathbb{K}$. Thus with random linear combinations the strategy used in Proposition 26 is not valid.

Proposition 31. *If $\mathbb{K}$ is a field with at least $(4d+2)d$ elements then the algorithm Sederberg Generalized uses $\tilde{\mathcal{O}}(d^n)$ arithmetic operations.*

Proof. The computations of $f_i(\underline{a})$, $g_i(\underline{a})$, $f_i(\underline{b})$, $g_i(\underline{b})$ needs $\tilde{\mathcal{O}}(d^n)$ arithmetic operations. The complexity of an n-variate gcd computation needs $\tilde{\mathcal{O}}(d^n)$ arithmetic operations. Indeed, as $\mathbb{K}$ is a field with at least $(4d+2)d$ elements with Lemma 6.44 in [39] we can generalize to n variables the algorithm 6.36 presented in [39] and obtain a result like Corollary 11.9 in [39]. This gives the desired result. $\square$

Remark 32. When it is possible, a polynomial generator is desirable. The algorithm Sederberg Generalized always returns a rational generator. We can test if we have a polynomial generator in the following way: We test if there exist $\alpha, \beta \in \mathbb{K}$ such that $\alpha H_a + \beta = H_b$. If such constants exist then H_a (or H_b) is a polynomial generator. This improvement is correct because we have seen during the proof of Proposition 26 that $H_a = h_1 - h(\underline{a})h_2$ and $H_b = h_1 - h(\underline{b})h_2$. Thus if a polynomial generator h_1 exists we have $H_a = h_1 - h_1(\underline{a})$ and $H_b = h_1 - h_1(\underline{b})$. As gcd are known up to a multiplicative constant there exist $\alpha, \beta \in \mathbb{K}$ such that $\alpha H_a + \beta = H_b$. Conversely, if we have $\alpha H_a + \beta = H_b$ then $H_a/H_b = u \circ H_a$ with $u = T/(\alpha T + \beta)$, thus $\mathbb{K}(H_a/H_b) = \mathbb{K}(H_a)$.

The computation of α and β needs $\mathcal{O}(d^n)$ arithmetic operations. Indeed, we solve a linear system with $\mathcal{O}(d^n)$ equations and two unknowns. Thus we can find a polynomial generator with the algorithm Sederberg Generalized with $\tilde{\mathcal{O}}(d^n)$ arithmetic operations.

4.2. **Another strategy based on decomposition.** Now, we give another algorithm to compute a Lüroth's generator. Here we use the relation between decomposition and computation of a Lüroth's generator.

Lüroth with Decomp
Input: $f(\underline{X}) = f_1/f_2(\underline{X}), g(\underline{X}) = g_1/g_2(\underline{X}) \in \mathbb{K}(X_1, \ldots, X_n)$ two reduced rational functions, $\underline{z} := (\underline{a}, \underline{b}) \in \mathbb{K}^{2n}$.
Output: $h(\underline{X}) \in \mathbb{K}(\underline{X})$ such that $\mathbb{K}(f, g) = \mathbb{K}(h)$, if h exists.

(1) Decompose f with the algorithm Decomp, then $f = u \circ h$.
(2) Compute v (if it exists) such that $g = v \circ h$.
(3) If v do not exist then Return "No Lüroth's generator", else go to 4.
(4) Compute w the GCRC of u and v with Sederberg's algorithm.
(5) Return $w \circ h$.

Proposition 33. *The algorithm Lüroth's with Decomp is correct for z satisfying the hypothesis of Theorem 2.*

Proof. This algorithm computes a GCRC of f and g, thus by Proposition 28, this gives the desired result. $\qquad \square$

Proposition 34. *Under hypotheses (C) and (H), the algorithm Lüroth's with Decomp performs one factorization of a univariate polynomial of degree d over $\mathbb{K}$ plus a number of operations in $\mathbb{K}$ belonging to $\tilde{\mathcal{O}}(d^n)$ if $n \geq 3$ or to $\tilde{\mathcal{O}}(d^3)$ if $n = 2$.*

Proof. The first step of the algorithm performs one factorization of a univariate polynomial of degree d over $\mathbb{K}$ plus a number of operations in $\mathbb{K}$ belonging to $\tilde{\mathcal{O}}(d^n)$ if $n \geq 3$ or to $\tilde{\mathcal{O}}(d^3)$ if $n = 2$ by Proposition 14.
With the strategy presented in Section 2.1, the second step can be done with $\tilde{\mathcal{O}}(d^n)$ arithmetic operations.
The last step can be done in an efficient probabilistic way, see [32]. The algorithm presented in [32] computes only two gcd's of univariate polynomials of degree lower than d.
Then the total cost of the algorithm belongs to $\tilde{\mathcal{O}}(d^n)$ if $n \geq 3$ or to $\tilde{\mathcal{O}}(d^3)$ if $n = 2$. $\qquad \square$

Remark 35. During the algorithm Lüroth with Decomp we have to avoid the roots of nonzero polynomials considered in Remark 13 and Remark 15 because we use

the algorithm Decomp. Furthermore during the algorithm Lüroth with Decomp, we use Sederberg's algorithm, this algorithm is also probabilistic and has in input two parameters $x_1, x_2 \in \mathbb{K}$. If x_1 and x_2 are not the roots of a nonzero polynomials then the output is correct, see [32].

Thus the nonzero polynomials are just the ones used for the algorithm Decomp and for Sederberg's algorithm.

4.3. Computation of a Lüroth's generator.

Lüroth's generator

Input: $\mathsf{f}_1(\underline{X}), \ldots, \mathsf{f}_m(\underline{X}) \in \mathbb{K}(\underline{X})$, m reduced rational functions, $z := \underline{z}_2, \ldots, \underline{z}_m \in \mathbb{K}^{2n}$, $n \geq 2$.

Output: $h(\underline{X}) \in \mathbb{K}(\underline{X})$ such that $\mathbb{K}(\mathsf{f}_1, \ldots, \mathsf{f}_m) = \mathbb{K}(h)$, if h exists.

   (1) Compute a Lüroth's generator of $\mathbb{K}(\mathsf{f}_1, \mathsf{f}_2)$ with Sederberg Generalized applied to $\mathsf{f}_1, \mathsf{f}_2$, with z_2 .
   (2) If a Lüroth's generator h is found then go to step 3 else Return "No Lüroth's generator".
   (3) For $i = 3, \ldots, m$,
       (a) Compute a Lüroth's generator of $\mathbb{K}(h, \mathsf{f}_i)$ with Sederberg Generalized applied to h, f_i, with z_i.
       (b) If a Lüroth's generator H is found then $h := H$ else Return "No Lüroth's generator".
   (4) Return h.

Proposition 36. *The algorithm Lüroth's generator is correct for z satisfying the hypothesis of Theorem 4.*

Proof. We just have to remark that $\mathbb{K}(\mathsf{f}_1, \ldots, \mathsf{f}_{i-1}, \mathsf{f}_i) = \mathbb{K}(\mathsf{f}_1, \ldots, \mathsf{f}_{i-1})(\mathsf{f}_i)$. $\square$

Proposition 37. *If $\mathbb{K}$ has at least $(4d + 2)d$ elements, then the algorithm Lüroth's generator can be performed with $\tilde{\mathcal{O}}(md^n)$ arithmetic operations in $\mathbb{K}$.*

Proof. We use m times the algorithm Sederberg Generalized. Thus, thanks to Proposition 31 we get the desired complexity. $\square$

Remark 38. During the algorithm Lüroth's generator we can use the algorithm Lüroth with Decomp instead of Sederberg Generalized. In the bivariate case, the complexity becomes then $\tilde{\mathcal{O}}(d^3)$. In this case the algorithm is not softly optimal, but the algorithm can also return u such that $f = u \circ h$.

We conclude that Proposition 36 and Proposition 37 prove Theorem 4.

5. Study of the Gutierez-Rubio-Sevilla's algorithm

In this section we study the complexity of the decomposition algorithm given in [15]. More precisely, we explain how to modify it in order to get a polynomial time algorithm instead of an exponential time algorithm.

5.1. Some preliminary results.

The following lemma is a generalization of Lemma 8.

Lemma 39. *Let $h = h_1/h_2$ be a rational function in $\mathbb{K}(\underline{X})$, $u = u_1/u_2$ a rational function in $\mathbb{K}(T)$ and set $f = u \circ h$ with $f = f_1/f_2 \in \mathbb{K}(\underline{X})$. Let $\lambda, \mu \in \mathbb{L}$, where $\mathbb{L}$ is a field and $\mathbb{K} \subset \mathbb{L}$. We have:*

$$\mu f_1 - \lambda f_2 = (\mu u_1 - \lambda u_2)(h).h_2^{\deg u}.$$

Proof. We have

$$\frac{\mu f_1 - \lambda f_2}{f_2} = \mu\frac{u_1(h)}{u_2(h)} - \lambda\frac{u_2(h)}{u_2(h)} = \frac{\mu u_1(h) - \lambda u_2(h)}{u_2(h)}.$$

Thus: $(\star)$ $(\mu f_1 - \lambda f_2).u_2(h) = (\mu u_1 - \lambda u_2)(h).f_2$.
Furthermore

$$(\star\star)\ \frac{f_1}{f_2} = \frac{u_1(h)}{u_2(h)} = \frac{\left(\sum_{i=0}^{d_1} a_i h_1^i h_2^{d_1-i}\right).h_2^{d_2}}{\left(\sum_{i=0}^{d_2} b_i h_1^i h_2^{d_2-i}\right).h_2^{d_1}},$$

where $u_1(T) = \sum_{i=0}^{d_1} a_i T^i$, $u_2(T) = \sum_{i=0}^{d_2} b_i T^i$.
Then $f_2 = \left(\sum_{i=0}^{d_2} b_i h_1^i h_2^{d_2-i}\right).h_2^{\max(d_1-d_2,0)}$ because f is reduced and the degree of
the right term of $(\star\star)$ is lower or equal to $\deg(f)$.
It follows $f_2 = u_2(h).h_2^{\max(d_1-d_2,0)+d_2} = u_2(h).h_2^{\deg u}$, then thanks to $(\star)$ we deduce
the desired result. $\square$

Proposition 40. *Let $f \in \mathbb{K}(\underline{X})$ be a rational function such that $f = u \circ h$ and
$f = u \circ \varphi$, where u is a rational function in $\mathbb{K}(T)$, h a non-composite rational
function and φ a rational function.*
*Then φ is non-composite and there exists $w \in \mathbb{K}(T)$ such that $h = w \circ \varphi$ and
$\deg w = 1$.*

Remark 41. w is not necessarily the identity. For example if $u = x^2 + 1/x^2$ and
$w = 1/x$ then $u \circ w = u$. Thus we can get $f = (u \circ w) \circ \varphi = u \circ \varphi$ and $f = u \circ (w \circ \varphi) =
u \circ h$. See [16] for more statements on the particular situation $u \circ w = u$.

Proof. We set $u = u_1/u_2$ and $\varphi = \varphi_1/\varphi_2$.
Let $\lambda, \mu \in \overline{\mathbb{K}}$ such that $\deg(\mu u_1 - \lambda u_2) = \deg u$, by Lemma 39 we have

$$\mu f_1 - \lambda f_2 = e \prod_{i=1}^{\deg u} (h_1 - x_i h_2),$$

where $e \in \overline{\mathbb{K}}$ and $x_i \in \overline{\mathbb{K}}$ are the roots of $\mu u_1 - \lambda u_2$.
We can suppose that $h_1 - x_i h_2$ are absolutely irreducible and $x_i \neq x_j$ if $i \neq j$.
Indeed, the "bad" values of $(\mu : \lambda)$ are $(u_2(x) : u_1(x))$ where $x \in \sigma(h_1, h_2)$ and
are the roots of $R(\mu, \lambda) = Res(\mu u_1 - \lambda u_2, \mu u_1' - \lambda u_2')$. As $\sigma(h_1, h_2)$ is finite and $\overline{\mathbb{K}}$
infinite, we deduce that "good" values of $(\mu : \lambda)$ exist.
We can also suppose that $\deg \varphi_1 - x_i \varphi_2 = \deg \varphi$, because we just have to avoid a
finite number of x_i.
Then Lemma 39 also implies

$$\mu f_1 - \lambda f_2 = e \prod_{i=1}^{\deg u} (\varphi_1 - x_i \varphi_2).$$

We have $\varphi_1 - x_i \varphi_2$ is absolutely irreducible, else $\mu f_1 - \lambda f_2$ has more than $\deg u$
absolute irreducible factors: this is a contradiction with $h_1 - x_i h_2$ being absolutely
irreducible.
Then φ is non-composite by Proposition 7.
Furthermore, there exist i_k, j_k, with $k = 1, \ldots, \deg u$ such that $h_1 - x_{i_k} h_2$ equal
$\varphi_1 - x_{j_k} \varphi_2$ up to a multiplicative constant. As in the proof of Proposition 26 it
follows $\varphi = w \circ h$ with $w \in \overline{\mathbb{K}}(T)$ and $\deg w = 1$. As h and φ belongs to $\mathbb{K}(\underline{X})$ we
have $w \in \mathbb{K}(T)$. (Indeed we just have to solve a linear system in $\mathbb{K}$ to get w.) $\square$

5.2. **Study of the absolute irreducible factors of near-separated polynomials.** The decomposition algorithm given in [15] is based on the following theorem; see [30]. In this subsection we improve this result.

Theorem 42. *Let $f = f_1/f_2 \in \mathbb{K}(\underline{X})$.*
$f = u \circ h$, with $h = h_1/h_2$ if and only if $H(\underline{X}, \underline{Y}) = h_1(\underline{X})h_2(\underline{Y}) - h_2(\underline{X})h_1(\underline{Y})$ divides $F(\underline{X}, \underline{Y}) = f_1(\underline{X})f_2(\underline{Y}) - f_2(\underline{X})f_1(\underline{Y})$.

In the following we use a result due to Schinzel.

Definition 43. A rational function is reducible over $\mathbb{K}$ if the numerator in its reduced form is reducible over $\mathbb{K}$.

Lemma 44. *Let $\Psi(T, \underline{Y})$ and $f(\underline{X})$ be non-constant rational functions over $\mathbb{K}$, the former of non-negative degree with respect to T and to at least one Y_i.*
If the function
$$\psi\big(f(\underline{X}), \underline{Y}\big)$$
is reducible over $\mathbb{K}$ then $f = u \circ h$, $u \in \mathbb{K}(T)$, $h \in \mathbb{K}(\underline{X})$ and $\psi\big(u(T), \underline{Y}\big)$ is reducible over $\mathbb{K}$.

Proof. See [29, Lemma 1]. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Proposition 45. *Let $f = f_1/f_2 \in \mathbb{K}(\underline{X})$, $\hat{f} = \hat{f}_1/\hat{f}_2 \in \mathbb{K}(\underline{Y})$ be two non-constant rational functions.*
If f and $\hat{f}$ are non-composite then $F(\underline{X}, \underline{Y}) = f_1(\underline{X})\hat{f}_2(\underline{Y}) - f_2(\underline{X})\hat{f}_1(\underline{Y})$ is irreducible in $\mathbb{K}[\underline{X}, \underline{Y}]$.

Proof. We set $\psi(T, \underline{Y}) = \hat{f}(\underline{Y}) - T$.
Then
$$\psi\big(f(\underline{X}), \underline{Y}\big) = \frac{\hat{f}_1(\underline{Y})f_2(\underline{X}) - f_1(\underline{X})\hat{f}_2(\underline{Y})}{f_2(\underline{X})\hat{f}_2(\underline{Y})}.$$
If we suppose $F(\underline{X}, \underline{Y})$ reducible then $f = u \circ h$ and $\psi\big(u(T), \underline{Y}\big)$ is reducible by Lemma 44.
As f is non-composite $\deg u = 1$ thus we can set $u(T) = (aT + b)/(\alpha T + \beta)$. Then $\psi\big(u(T), \underline{Y}\big)$ is reducible means $\hat{f}_1(\underline{Y})(\alpha T + \beta) - \hat{f}_2(\underline{Y})(aT + b)$ is reducible over $\mathbb{K}$. By Proposition 7 this is absurd because $\hat{f}$ is non-composite. Hence $F(\underline{X}, \underline{Y})$ is irreducible. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Now we can improve Theorem 42.

Theorem 46. *Let $f = f_1/f_2 \in \mathbb{K}(\underline{X})$ a non-constant rational function.*
If $f = u \circ h$, where $u = u_1/u_2 \in \mathbb{K}(T)$ and $h = h_1/h_2 \in \mathbb{K}(\underline{X})$ are rational functions, with $\deg u \geq 2$ and h non-composite, then the irreducible factors with the smallest degree relatively to $\underline{X}$ of
$$F(\underline{X}, \underline{Y}) = f_1(\underline{X})f_2(\underline{Y}) - f_2(\underline{X})f_1(\underline{Y})$$
are of the kind
$$H(X, Y) = h_1(\underline{X})\varphi_{i,2}(\underline{Y}) - h_2(\underline{X})\varphi_{i,1}(\underline{Y}),$$
where $\varphi_i = \varphi_{i,1}/\varphi_{i,2}$ are non-composite rational functions such that $h = w \circ \varphi_i$ with $\deg w = 1$.

Theorem 5 is a direct consequence of Theorem 46.

Proof. By Lemma 39, we have

$$(\star) \ F(\underline{X}, \underline{Y}) = U_{f_1,f_2}\big(h(\underline{X})\big).h_2(\underline{X})^{\deg u},$$

where

$$U_{f_1,f_2}(T) = f_2(\underline{Y})u_1(T) - f_1(\underline{Y})u_2(T).$$

As $f = u \circ h$, $h(Y)$ is a root of U_{f_1,f_2}. Then

$$U_{f_1,f_2}(T) = \big(h_2(\underline{Y})T - h_1(\underline{Y})\big)A(\underline{Y}, T),$$

where $A(\underline{Y}, T) \in \mathbb{K}[\underline{Y}, T]$. Thus $(\star)$ implies $h_1(\underline{X})h_2(\underline{Y}) - h_2(\underline{X})h_1(\underline{Y})$ divides $F(\underline{X}, \underline{Y})$.

Now, we suppose that $\varphi(\underline{Y}) \in \mathbb{K}(\underline{Y})$ is another root of $U_{f_1,f_2}(T)$. Then

$$u\big(\varphi(\underline{Y})\big) = f(\underline{Y}) = u\big(h(\underline{Y})\big).$$

Thus, by Lemma 40, we have φ is non-composite and $h = w \circ \varphi$ with $\deg w = 1$. As before, we can write $U_{f_1,f_2} = \big(\varphi_2(\underline{Y})T - \varphi_1(\underline{Y})\big).B(\underline{Y}, T)$, where $B(\underline{Y}, T) \in \mathbb{K}[\underline{Y}, T]$. Thus $\varphi_2(\underline{Y})h_1(\underline{X}) - \varphi_1(\underline{Y})h_2(\underline{X})$ divides $F(\underline{X}, \underline{Y})$ by $(\star)$.

Now, we write

$$(\star\star) \ U_{f_1,f_2}(T) = \prod_{i \in I} \big(\varphi_{i,2}(\underline{Y})T - \varphi_{i,1}(\underline{Y})\big).\prod_{j \in J} C_j^{e_j}(\underline{Y}, T),$$

where $\varphi_i = \varphi_{i,1}/\varphi_{i,2}(\underline{Y})$ is a reduced non-composite rational function as explained above and $C_j(\underline{Y}, T) \in \mathbb{K}[\underline{Y}, T]$ is irreducible with $\deg_T C_j \geq 2$.

We evaluate T to h in $(\star\star)$ and multiply the result by $h_2^{\deg u}$:

$$
\begin{aligned}
U_{f_1,f_2}\big(h(\underline{X})\big).h_2(\underline{X})^{\deg u} \ = \ & \prod_{i \in I} \big(\varphi_{i,2}(\underline{Y})h_1(\underline{X}) - \varphi_{i,1}(\underline{Y})h_2(\underline{X})\big) \\
& \times \Big(\prod_{j \in J} C_j^{e_j}\big(\underline{Y}, h(\underline{X})\big)\Big).h_2(\underline{X})^{\sum_{j \in J} e_j \deg_T C_j}.
\end{aligned}
$$

The factors $\varphi_{i,2}(\underline{Y})h_1(\underline{X}) - \varphi_{i,1}(\underline{Y})h_2(\underline{X})$ are irreducible by Proposition 45. Furthermore, by Lemma 44 as h is non-composite and $C_j(\underline{Y}, T)$ is irreducible, we have $C_j\big(\underline{Y}, h(\underline{X})\big).h_2(\underline{X})^{\deg_T C_j}$ is irreducible in $\mathbb{K}[\underline{X}, \underline{Y}]$.

We also have

$$
\begin{aligned}
\deg_X C_j\big(\underline{Y}, h(\underline{X})\big)h_2(\underline{X})^{\deg_T C_j} \ = \ & \deg_T C_j.\deg h \\
\geq \ & 2 \deg h \\
> \ & \deg_X \varphi_{i,2}(\underline{Y})h_1(\underline{X}) - \varphi_{i,1}(\underline{Y})h_2(\underline{X}).
\end{aligned}
$$

Then $H(X, Y) = \varphi_{i,2}(\underline{Y})h_1(\underline{X}) - \varphi_{i,1}(\underline{Y})h_2(\underline{X})$ are the factors with the smallest degree relatively to $\underline{X}$. □

5.3. **Improvement of the GRS algorithm.** Now we describe the decomposition algorithm presented in [15].

   GRS decomposition algorithm
Input: $f(\underline{X}) = f_1/f_2(\underline{X})$, $n \geq 2$.
Output: $u \in \mathbb{K}(T)$, $h(\underline{X}) \in \mathbb{K}(\underline{X})$ such that $f = u \circ h$, or "f is non-composite".

   (1) Factor $F(\underline{X}, \underline{Y})$. Let $D = \{H_1, \ldots, H_m\}$ be the set of factors of F (up to product by constants). We set $i = 1$.

(2) If H_i can be written $H_i(\underline{X}, \underline{Y}) = h_1(\underline{X})h_2(\underline{Y}) - h_1(\underline{Y})h_2(\underline{X})$ then h_1/h_2 is a right component for f. Then compute u by solving a linear system and Return u, h.

(3) If $i < m$ then $i := i+1$ and go to step 2, else Return "f is non-composite".

This algorithm has an exponential time complexity. Indeed, the set D contains at most 2^d polynomials, where d is the degree of f.

However, we can improve this algorithm. Thanks to Proposition 45, we remark that if f is non-composite then F is irreducible. Furthermore, if $f = u \circ h$ with h non-composite, then $H(X, Y) = h_1(\underline{X})h_2(\underline{Y}) - h_1(\underline{X})h_2(\underline{Y})$ is an irreducible factor of $F(\underline{X}, \underline{Y})$, by Theorem 46. Thus we have to study at most $\deg F$ irreducible factors. Thus we can substitute the set D by the set of *irreducible* factors. (We can also substitute the set D by the set of *irreducible factors with the smallest degree relatively to $\underline{X}$*). As Step 1 and Step 2 can be done in a polynomial time, it follows:

Proposition 47. *If in the GRS decomposition algorithm we set: "D is the set of irreducible factors of F", then this modified algorithm has a polynomial time complexity.*

Remark 48. The bottleneck of this modified algorithm is the factorization of F. If we apply the deterministic algorithm proposed in [22] then the modified GRS decomposition algorithm uses $\tilde{\mathcal{O}}(d^{2n+\omega-1})$ arithmetic operations, where d is the degree of f and n the number of variables.

Exemple 49. Now, we illustrate the GRS decomposition algorithm with $f = u \circ h$, where $u = (T^2 + 1)/T$, $h = h_1/h_2$, and $h_1 = X_1^3 + X_2^3 + 1$, $h_2 = 3X_1X_2$. h is a non-composite rational function.
In this situation, we have the following factorization of $F(X_1, X_2, Y_1, Y_2)$:

$$F(X_1, X_2, Y_1, Y_2) = 3.H_1(X_1, X_2, Y_1, Y_2).H_2(X_1, X_2, Y_1, Y_2), \text{ where}$$

$$
\begin{aligned}
H_1(X_1, X_2, Y_1, Y_2) &= X_1^3 Y_1 Y_2 + X_2^3 Y_1 Y_2 + Y_1 Y_2 - Y_1^3 X_1 X_2 - Y_2^3 X_1 X_2 - X_1 X_2 \\
&= h_1(X_1, X_2)h_2(Y_1, Y_2) - h_1(Y_1, Y_2)h_2(X_1, X_2), \\
H_2(X_1, X_2, Y_1, Y_2) &= 1 + X_1^3 + X_2^3 + Y_1^3 + Y_2^3 + X_1^3 Y_1^3 + X_1^3 Y_2^3 + X_2^3 Y_1^3 + X_2^3 Y_2^3 \\
&\quad -9X_1 X_2 Y_1 Y_2 \\
&= h_1(X_1, X_2)h_1(Y_1, Y_2) - h_2(Y_1, Y_2)h_2(X_1, X_2).
\end{aligned}
$$

Then we can recover the decomposition $f = u \circ h$ with the GRS decomposition algorithm.

References

[1] Cesar Alonso, Jaime Gutierrez, and Tomas Recio. A rational function decomposition algorithm by near-separated polynomials. *J. Symbolic Comput.*, 19(6):527–544, 1995.

[2] Shreeram S. Abhyankar, William J. Heinzer, and Avinash Sathaye. Translates of polynomials. In *A tribute to C. S. Seshadri (Chennai, 2002)*, Trends Math., pages 51–124. Birkhäuser, Basel, 2003.

[3] V. S. Alagar and Mai Thanh. Fast polynomial decomposition algorithms. In *EUROCAL '85, Vol. 2 (Linz, 1985)*, volume 204 of *Lecture Notes in Comput. Sci.*, pages 150–153. Springer, Berlin, 1985.

[4] Laurent Busé and Guillaume Chèze. On the total order of reducibility of a pencil of algebraic plane curves. Preprint, 2008.

[5] Laurent Busé and Carlos D'Andrea. A matrix-based approach to properness and inversion problems for rational surfaces. *Appl. Algebra Engrg. Comm. Comput.*, 17(6):393–407, 2006.

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

[7] Arnaud Bodin. Reducibility of rational functions in several variables. *Israel J. Math.*, 164:333–347, 2008.

[8] Dario Bini and Victor Y. Pan. *Polynomial and matrix computations. Vol. 1.* Progress in Theoretical Computer Science. Birkhäuser Boston Inc., Boston, MA, 1994. Fundamental algorithms.

[9] David R. Barton and Richard Zippel. Polynomial decomposition algorithms. *J. Symbolic Comput.*, 1(2):159–168, 1985.

[10] Guillaume Chèze and Grégoire Lecerf. Lifting and recombination techniques for absolute factorization. *J. Complexity*, 23(3):380–420, 2007.

[11] M. Dickerson. Polynomial decomposition algorithms for multivariate polynomials. Technical Report TR87-826, Comput. Sci., Cornell Univ., 1987.

[12] Roberto Dvornicich and Carlo Traverso. Newton symmetric functions and the arithmetic of algebraically closed fields. In *Applied algebra, algebraic algorithms and error-correcting codes (Menorca, 1987)*, volume 356 of *Lecture Notes in Comput. Sci.*, pages 216–224. Springer, Berlin, 1989.

[13] Paul Gordan. ueber biquadratische Gleichungen. *Math. Ann.*, 29(3):318–326, 1887.

[14] Shuhong Gao and Virgínia M. Rodrigues. Irreducibility of polynomials modulo p via Newton polytopes. *J. Number Theory*, 101(1):32–47, 2003.

[15] Jaime Gutierrez, Rosario Rubio, and David Sevilla. Unirational fields of transcendence degree one and functional decomposition. In *ISSAC '01: Proceedings of the 2001 international symposium on Symbolic and algebraic computation*, pages 167–174, New York, NY, USA, 2001. ACM Press.

[16] Jaime Gutierrez and David Sevilla. Building counterexamples to generalizations for rational functions of Ritt's decomposition theorem. *J. Algebra*, 303(2):655–667, 2006.

[17] Jun-ichi Igusa. On a theorem of Lueroth. *Mem. Coll. Sci. Univ. Kyoto Ser. A. Math.*, 26:251–253, 1951.

[18] J. P. Jouanolou. *Équations de Pfaff algébriques*, volume 708 of *Lecture Notes in Mathematics*. Springer, Berlin, 1979.

[19] Erich Kaltofen. Fast parallel absolute irreducibility testing. *J. Symbolic Comput.*, 1(1):57–67, 1985.

[20] Dexter Kozen and Susan Landau. Polynomial decomposition algorithms. *J. Symbolic Comput.*, 7(5):445–456, 1989.

[21] Grégoire Lecerf. Sharp precision in Hensel lifting for bivariate polynomial factorization. *Math. Comp.*, 75(254):921–933 (electronic), 2006.

[22] Grégoire Lecerf. Improved dense multivariate polynomial factorization algorithms. *J. Symbolic Comput.*, 42(4):477–494, 2007.

[23] Dino Lorenzini. Reducibility of polynomials in two variables. *J. Algebra*, 156(1):65–75, 1993.

[24] Jean Moulin Ollagnier. Algebraic closure of a rational function. *Qual. Theory Dyn. Syst.*, 5(2):285–300, 2004.

[25] Jörn Müller-Quade and Rainer Steinwandt. Recognizing simple subextensions of purely transcendental field extensions. *Appl. Algebra Engrg. Comm. Comput.*, 11(1):35–41, 2000.

[26] H. Poincaré. Sur l'intégration algébrique des équations différentielles du premier ordre. *Rendiconti des Circolo Mathematico di Palermo*, 5:161–191, 1891.

[27] J. F. Ritt. Prime and composite polynomials. *Trans. Amer. Math. Soc.*, 23(1):51–66, 1922.

[28] Wolfgang Ruppert. Reduzibilität Ebener Kurven. *J. Reine Angew. Math.*, 369:167–191, 1986.

[29] A. Schinzel. Reducibility of polynomials in several variables. II. *Pacific J. Math.*, 118(2):531–563, 1985.

[30] Josef Schicho. A note on a theorem of Fried and MacRae. *Arch. Math. (Basel)*, 65(3):239–243, 1995.

[31] A. Schinzel. *Polynomials with special regard to reducibility*, volume 77 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2000. With an appendix by Umberto Zannier.

[32] Thomas W. Sederberg. Improperly parametrized rational curves. *Computer Aided Geometric Design*, 3(1):67–75, 1986.

[33] A. Storjohann. *Algorithms for matrix canonical forms.* PhD thesis, ETH Zurich, Zurich, Switzerland, 2000.

[34] Barry M. Trager. Algebraic factoring and rational function integration. In *Proceedings of the third ACM symposium on Symbolic and Algebraic Computation*, pages 219–226. ACM Press, 1976.

[35] Carlo Traverso. A study on algebraic algorithms: the normalization. *Rend. Sem. Mat. Univ. Politec. Torino*, (Special Issue):111–130 (1987), 1986. Conference on algebraic varieties of small dimension (Turin, 1985).

[36] Angelo Vistoli. The number of reducible hypersurfaces in a pencil. *Invent. Math.*, 112(2):247–262, 1993.

[37] Joachim von zur Gathen. Functional decomposition of polynomials: the tame case. *J. Symbolic Comput.*, 9(3):281–299, 1990.

[38] Joachim von zur Gathen. Functional decomposition of polynomials: the wild case. *J. Symbolic Comput.*, 10(5):437–452, 1990.

[39] Joachim von zur Gathen and Jürgen Gerhard. *Modern computer algebra*. Cambridge University Press, Cambridge, second edition, 2003.

[40] Joachim von zur Gathen, Jaime Gutierrez, and Rosario Rubio. Multivariate polynomial decomposition. *Appl. Algebra Engrg. Comm. Comput.*, 14(1):11–31, 2003.

[41] R. Zippel. Rational function decomposition. In *Proceedings of the 1991 international symposium on Symbolic and algebraic computation*, pages 1–6. ACM Press, 1991.

Institut de Mathématiques de Toulouse, Université Paul Sabatier Toulouse 3, MIP Bât 1R3, 31 062 TOULOUSE cedex 9, France

E-mail address: guillaume.cheze@math.univ-toulouse.fr