Computing low-degree factors of lacunary polynomials: a Newton-Puiseux approach

Bruno Grenet*
LIX – UMR 7161
École Polytechnique
91128 Palaiseau Cedex, France
bruno.grenet@lix.polytechnique.fr

January 19, 2014

Abstract

We present a new algorithm for the computation of the irreducible factors of degree at most d, with multiplicity, of multivariate lacunary polynomials over fields of characteristic zero. The algorithm reduces this computation to the computation of irreducible factors of degree at most d of univariate lacunary polynomials and to the factorization of low-degree multivariate polynomials. The reduction runs in time polynomial in the size of the input polynomial and in d. As a result, we obtain a new polynomial-time algorithm for the computation of low-degree factors, with multiplicity, of multivariate lacunary polynomials over number fields, but our method also gives partial results for other fields, such as the fields of p-adic numbers or for absolute or approximate factorization for instance.

The core of our reduction uses the Newton polygon of the input polynomial, and its validity is based on the Newton-Puiseux expansion of roots of bivariate polynomials. In particular, we bound the valuation of $f(X, \phi)$ where f is a lacunary polynomial and ϕ a Puiseux series whose vanishing polynomial has low degree.

*Supported by the LIX-Qualcomm-Carnot fellowship.

1 Introduction

This article proposes a new algorithm for computing low-degree factors of lacunary polynomials over fields of characteristic 0. The *lacunary representation* of a polynomial

$$f(X_1,\dots,X_n)=\sum_{j=1}^{k} c_j X_1^{\alpha_{1,j}}\cdots X_n^{\alpha_{n,j}}$$

is the list $\{(c_j,\alpha_{1,j},\dots,\alpha_{n,j}) : 1\le j\le k\}$. We define the lacunary size of f, denoted by $\operatorname{size}(f)$, as the size of the binary representation of this list. It takes into account the size of the coefficients, and thus depends on the field they belong to. An important remark is that the size is proportional to the logarithm of the degree.

Over algebraic number fields, the factorization problem can be solved in time polynomial in the degree of the input polynomial (see for instance [18] and references therein). It is also the case of absolute factorization, that is factorization over the algebraic closure of $\mathbb{Q}$ [6]. In the case of lacunary polynomials, these algorithms are not adapted since they are exponential in the size of the representation.

Actually, the computation of the irreducible factorization of a polynomial given in lacunary representation cannot be performed in polynomial time. For instance over $\mathbb{Q}$, the polynomial X^p-1 has a size of order $\log(p)$, while one of its irreducible factors, namely $(1+X+\cdots+X^{p-1})$, has a size of order p.

Therefore, a natural restriction consists in computing low-degree factors only. A line of work yielded an algorithm that, given a lacunary polynomial $f\in\mathbb{K}[X_1,\dots,X_n]$ and an integer d as input, where $\mathbb{K}$ is an algebraic number fields, computes all the irreducible factors of f of degree at most d in time polynomial in $\operatorname{size}(f)$ and d [7, 16, 11, 12]. These papers use so-called Gap Theorems: If an irreducible polynomial g of degree at most d divides $f=\sum_{j=1}^{k} c_j\boldsymbol{X}^{\boldsymbol{\alpha}_j}$, then there exists an index ℓ such that g divides both $\sum_{j=1}^{\ell} c_j\boldsymbol{X}^{\boldsymbol{\alpha}_j}$ and $\sum_{j=\ell+1}^{k} c_j\boldsymbol{X}^{\boldsymbol{\alpha}_j}$. This allows to reduce the computation to the case of low-degree polynomials, for which one applies the classical algorithms. These Gap Theorems are based on number-theoretic results.

We recently proposed a new approach for this problem. In [4, 3], we gave a new algorithm for the computation of the multilinear factors of multivariate lacunary polynomials. The algorithm we obtained is simpler and faster than the previous ones. Moreover, since it is not based on number-theoretic results, it can be used for a larger range of fields, for instance for absolute or approximate factorization, or for finite fields of large characteristic. In this paper, we propose a generalization of this algorithm to the case of factors of degree at most d. We briefly explain the strategy proposed in [4, 3] in the simplest case of linear factors of bivariate polynomials.

Let $f = \sum_{j=1}^{k} c_j X^{\alpha_j} Y^{\beta_j} \in \mathbb{K}[X, Y]$ for some field $\mathbb{K}$ of characteristic 0, with $\alpha_j \leq \alpha_{j+1}$ for all j. A linear polynomial $(Y - uX - v)$ divides f if and only if $f(X, uX + v) = 0$. We proved that for $uv \neq 0$, if $f(X, uX + v)$ is nonzero then its valuation, that is the largest power of X dividing it, is bounded by $\alpha_1 + \binom{\ell}{2}$. From this, we deduced a Gap Theorem: Suppose that $f = f_1 + f_2$ where $f_1 = \sum_{j=1}^{\ell} c_j X^{\alpha_j} Y^{\beta_j}$, and $\alpha_{\ell+1} > \alpha_1 + \binom{\ell}{2}$. Then for all $uv \neq 0$, $f(X, uX + v) = 0$ if and only if $f_1(X, uX + v) = f_2(X, uX + v) = 0$. In other words, $(Y - uX - v)$ divides f if and only if it divides both f_1 and f_2. From this Gap Theorem, an algorithm for computing linear factors $(Y - uX - v)$ with $uv \neq 0$ follows quite easily: Apply the Gap Theorem recursively to express f as a sum of low-degree polynomials, and compute their common linear factors using any classical factorization algorithm. The computation of the remaining possible linear factors such as $(Y - uX)$ or $(X - v)$ reduces to univariate lacunary factorization.

To use the same strategy with degree-d factors, we need some new ingredients. First, we view a degree-d bivariate irreducible polynomial $g \in \mathbb{K}[X, Y]$ as a polynomial in Y whose coefficients are polynomials in X. The roots of g can be expressed in an algebraic closure of $\mathbb{K}[X]$ using the notion of *Puiseux series*. If ϕ is such a root of g, then g divides $f \in \mathbb{K}[X, Y]$ if and only if $f(X, \phi) = 0$. We give a bound on the valuation of such an expression where f is a lacunary polynomial. This yields a new Gap Theorem. Yet, the bound and the Gap Theorem depend on the valuation of the root ϕ itself. This means that there are actually as many Gap Theorems as there are possible valuations of ϕ. A second ingredient is the use of the *Newton polygon* of f to a priori compute these valuations. As in the case of linear factors, there are some special cases reducing to the univariate case, namely the *quasi-homogeneous* factors: $g = \sum_j b_j X^{\gamma_j} Y^{\delta_j}$ is said quasi-homogeneous if there exist p and q such that $p\gamma_j + q\delta_j$ is constant for all j. The computation of these factors too makes use of the Newton polygon of f.

As we shall see, our algorithm computes in very different manner the quasi-homogeneous factors and the other ones, called *non-homogeneous*. In the sequel, we give two algorithms: We first show how to compute the quasi-homogeneous factors, given an oracle to compute the irreducible factors of degree at most d of a univariate lacunary polynomial. The second algorithm reduces the computation of the non-homogeneous factors to the factorization of some bivariate low-degree polynomials.

Using both algorithms yields our first main result.

Theorem 1. *Let $\mathbb{K}$ be any field of characteristic 0. Given a lacunary polynomial $f \in \mathbb{K}[X, Y]$ of degree D with k nonzero terms and an integer d, the computation of the irreducible factors of degree at most d of f, with multiplicity, reduces to*

- *the computation of the irreducible factors of degree at most d of $k/2$ lacunary polynomials of $\mathbb{K}[X]$, and*
- *the factorization of polynomials of $\mathbb{K}[X, Y]$ of total degree sum at most $\mathcal{O}(d^4k^4)$,*

plus at most $(k \log D + d)^{\mathcal{O}(1)}$ bit operations.

In the multivariate case, we cannot directly apply the same algorithm as in the bivariate case since the resulting algorithm would be exponential in the number of variables. Nevertheless, we can prove the following result.

Theorem 2. *Let $\mathbb{K}$ be any field of characteristic 0. Given a lacunary polynomial $f \in \mathbb{K}[X_1, \dots, X_n]$ of degree D with k nonzero terms and an integer d, the computation of the irreducible factors of degree at most d of f, with multiplicity, reduces to*

- *the computation of the irreducible factors of degree at most d of $(nk)^{\mathcal{O}(1)}$ lacunary polynomials of $\mathbb{K}[X]$, and*
- *the factorization of polynomials of $\mathbb{K}[X, Y]$ of total degree sum at most $(nk \log(D) + d)^{\mathcal{O}(1)}$,*

plus at most $(nk \log D + d)^{\mathcal{O}(1)}$ bit operations.

In the case of number fields, Lenstra gave a polynomial-time algorithm to compute the factors of degree at most d of a univariate lacunary polynomial [16]. For the low-degree factorization of a polynomial g, there exist deterministic algorithms that run in time $(\text{size}(g) + \deg(g))^{\mathcal{O}(n)}$ and returns the list of factors in lacunary representation [9], and randomized algorithms that run in time $(\text{size}(g) + \deg(g))^{\mathcal{O}(1)}$ and returns the factors as straight-line programs [10] or blackboxes [14]. As a result, we obtain a new algorithm for the computation of factors of degree at most d of multivariate lacunary polynomials.

Corollary 3. *There exists an algorithm that, given as inputs an irreducible polynomial $\varphi \in \mathbb{Q}[\xi]$ representing a number field $\mathbb{K} = \mathbb{Q}[\xi]/\langle\varphi\rangle$, a lacunary polynomial $f \in \mathbb{K}[X_1, \dots, X_n]$ and an integer d, computes the lacunary representation of the irreducible factors of degree at most d of f, with multiplicity, in deterministic time $(\mathrm{size}(f) + d)^{\mathcal{O}(n)}$.*

If the factors are represented as straight-line programs or blackboxes, the algorithm is randomized and runs in time $(\mathrm{size}(f) + d)^{\mathcal{O}(1)}$.

Note that Theorems 1 and 2 are valid with any field of characteristic 0. For instance, for any field of characteristic 0 for which a polynomial-time algorithm is known for multivariate low-degree factorization, we obtain a polynomial-time algorithm to compute the non-homogeneous low-degree factors of multivariate lacunary polynomials. This fields include the algebraic closure $\overline{\mathbb{Q}}$ of $\mathbb{Q}$ (absolute factorization [6]), the fields of real or complex numbers (approximate factorization [13]), or the fields of p-adic numbers [5]. Note that for $\overline{\mathbb{Q}}$ and $\mathbb{C}$, one cannot expect to have a polynomial-time algorithm computing all low-degree factors of multivariate lacunary polynomials since as we shall see, the computation of quasi-homogeneous factors is equivalent to univariate lacunary factorization. The number of irreducible linear factors of a univariate polynomial over an algebraically closed field equals its degree, thus there are too many quasi-homogeneous factors to compute them in polynomial time. The case of polynomials with real (approximate) coefficients is open to the best of my knowledge. In this case, Descartes' rule of signs implies that the number of real roots, or linear factors, is bounded by $2k - 1$ where k is the number of terms. Therefore, it is an intriguing question whether these roots can be computed in polynomial time.

We conjecture that the reductions presented in the current paper are valid in large positive characteristic, as in [4, 3], using Hahn series rather than Puiseux series. Another intriguing question is the validity of the approach in fields of small positive characteristic.

Organization In Section 2, we collect some known facts about Newton polygons and Puiseux series. Section 3 is devoted to the computation of the quasi-homogeneous factors and Section 4 to the computation of non-homogeneous factors, both for bivariate polynomials. In Section 5 we give a proof sketch for the case of multivariate polynomials.

Convention We shall use the expression "*degree-d factors*" to denote the *irreducible factors of degree at most d* of a polynomial.

Acknowledgments I am grateful to P. Koiran, N. Portier and Y. Strozecki for the numerous discussions we had on this work. I also wish to thank A. Bostan, P. Lairez and B. Salvy for their help with Puiseux series.

2 Newton polygons and Puiseux series

We recall a few facts about Newton polygons and Puiseux series. For more on this topic, we refer the reader to [17, 1].

Let $f = \sum_j c_j X^{\alpha_j} Y^{\beta_j}$. Its *support* is the set $\mathrm{Supp}(f) = \{(\beta_j, \alpha_j) : c_j \neq 0\}$. The *Newton polygon* of f, denoted by $\mathrm{Newt}(f)$, is the convex hull of its support. Note that the definitions of the support and the Newton polygon depend on the asymmetric roles given to the two variables X and Y. For two convex polygon A and B, their Minkowski sum is the set $A + B = \{a + b : a \in A, b \in B\}$.

Theorem 4 (Ostrowski)**.** *Let f, g, $h \in \mathbb{K}[X,Y]$ such that $f = gh$. Then $\mathrm{Newt}(f) = \mathrm{Newt}(g) + \mathrm{Newt}(h)$.*

We shall use this theorem to a priori determine the shape of the factors of a bivariate lacunary polynomial. We can see the Newton polygon of f as a set of edges. Then by Ostrowski's Theorem, each edge of a factor of f has to be parallel to an edge of $\mathrm{Newt}(f)$. Moreover, if we consider a degree-d factor g, its Newton polygon is inside a square whose sides have length d. For an edge of endpoints (i,j) and (i',j'), its *slope* is defined as $(j'-j)/(i'-i)$. Thus the slopes of the edges of $\mathrm{Newt}(g)$ have the form p/q, $p \in \mathbb{Z}$, $q \in \mathbb{N}$, with $|p|, q \leq d$. By convention, we say that a vertical edge has slope $1/0$. In particular, only edges of $\mathrm{Newt}(f)$ with a slope p/q with $|p|, q \leq d$ can be edges of the Newton polygon of a factor of f.

Let $g \in \mathbb{K}[X,Y]$ be any bivariate polynomial, viewed as a polynomial in Y with coefficients in $\mathbb{K}[X]$. We are interested in the roots of g in the algebraic closure of $\mathbb{K}(X)$. This algebraic closure can be described using the *field of Puiseux series over the algebraic closure $\overline{\mathbb{K}}$ of* $\mathbb{K}$, denoted by $\overline{\mathbb{K}}\langle\langle X\rangle\rangle$. Its elements are formal sums $\phi = \sum_{t \geq t_0} f_t X^{t/d}$ where $f_t \in \overline{\mathbb{K}}$, $f_{t_0} \neq 0$, $t_0 \in \mathbb{Z}$ and $d \in \mathbb{N}$. All we need for our purpose is that $\overline{\mathbb{K}}\langle\langle X\rangle\rangle$ *contains* an algebraic closure of $\mathbb{K}(X)$. In other words, any root of $g \in \mathbb{K}[X][Y]$ can be described by a Puiseux series.

We define the valuation of a polynomial $f \in \mathbb{K}[X]$ by $\mathrm{val}(f) = \max\{v : X^v \text{ divides } f\}$. This valuation is easily extended to the field of Puiseux series: If $\phi = \sum_{t \geq t_0} f_t X^{t/d}$ with $f_{t_0/d} \neq 0$, then $\mathrm{val}(\phi) = t_0/d$. For bivariate polynomials in $\mathbb{K}[X,Y]$, we define similarly the valuations with respect to X and with respect to Y, and denote them by val_X and val_Y respectively.

Since $\overline{\mathbb{K}}\langle\langle X\rangle\rangle$ contains an algebraic closure of $\mathbb{K}(X)$, a bivariate polynomial $g \in \mathbb{K}[X][Y]$ of degree d in Y has exactly d roots (counted with multiplicity) in $\overline{\mathbb{K}}\langle\langle X\rangle\rangle$. That is, there exist $\phi_1, \ldots, \phi_d \in \overline{\mathbb{K}}\langle\langle X\rangle\rangle$ and $g_0 \in \mathbb{K}[X]$ such that

$$g(X,Y) = g_0(X)\prod_{i=1}^{d}(Y - \phi_i(X)).$$

The set $\{\mathrm{val}(\phi_i) : 1 \le i \le d\}$ can be described in terms of the Newton polygon of g.

Theorem 5 (Newton-Puiseux)**.** *Let $g \in \mathbb{K}[X][Y]$. It has a root of valuation v in $\overline{\mathbb{K}}\langle\langle X\rangle\rangle$ if and only if there is an edge of slope $-v$ in the lower hull of* $\mathrm{Newt}(g)$.

The lower hull of $\mathrm{Newt}(g)$ is the set of edges of $\mathrm{Newt}(g)$ which are below $\mathrm{Newt}(g)$, excluding vertical edges. We define in the same way the upper hull of $\mathrm{Newt}(g)$.

As a consequence of Ostrowski's Theorem and Newton-Puiseux Theorem, we can get informations on the roots of the factors of a polynomial by inspecting its Newton polygon.

Corollary 6. *Let $f, g \in \mathbb{K}[X,Y]$, where g is a degree-d factor of f. Then g has a root $\phi \in \overline{\mathbb{K}}\langle\langle X\rangle\rangle$ of valuation v only if there is an edge in the lower hull of* $\mathrm{Newt}(f)$ *of slope $-v = -p/q$ where $q > 0$ and $|p|, q \le d$.*

Let us suppose that we are given a bivariate lacunary polynomial $f = \sum_{j=1}^{k} c_j X^{\alpha_j} Y^{\beta_j}$ as the list of its nonzero terms, represented by triples (c_j, α_j, β_j). The common first step of our algorithms consists in computing the Newton polygon of f. This can be done in time polynomial in k and $\log(\deg(f))$ using for instance Graham's scan [8].

3 Quasi-homogeneous factors

The aim of this section is to reduce the computation of the degree-d quasi-homogeneous factors of a bivariate lacunary polynomial to univariate lacunary factorization. We first collect some useful facts on quasi-homogeneous polynomials. A polynomial $g = \sum_j b_j X^{\gamma_j} Y^{\delta_j}$ is said *(p,q)-homogeneous of order ω* if there exist two relatively prime integers p and q, $q \ge 0$, and $\omega \ge 0$ such that $p\gamma_j + q\delta_j = \omega$ for all j. In terms of the Newton polygon, this means that $\mathrm{Newt}(g)$ is contained in a line of slope $-q/p$. Note that there are two degenerate cases: $(1,0)$-homogeneous polynomials belong to $\mathbb{K}[Y]$ and $(0,1)$-homogeneous polynomials to $\mathbb{K}[X]$.

The product of two (p,q)-homogeneous polynomials of order ω_1 and ω_2 respectively is (p,q)-homogeneous of order $\omega_1+\omega_2$. Conversely, any factor of a (p,q)-homogeneous polynomial is itself (p,q)-homogeneous.

We shall also need a notion of (p,q)-homogenization of a univariate polynomial: If $p,q>0$, the (p,q)-homogenization of $h\in\mathbb{K}[X]$ is $h_{p,q}=Y^{p\deg(h)}h(X^q/Y^p)$. A monomial X^δ of h becomes $X^{q\delta}Y^{p(\deg(h)-\delta)}$. For all δ, $p(\deg(h)-\delta)\geq 0$ and $pq\delta+qp(\deg(h)-\delta)=pq\deg(h)$ is independent of δ. Thus $h_{p,q}$ is a (p,q)-homogeneous polynomial. If $p<0$ and $q>0$, the (p,q)-homogenization is defined by $h_{p,q}(X,Y)=h(X^qY^{-p})$. Since $p<0$, $h_{p,q}$ is a polynomial and one easily checks that it is (p,q)-homogeneous of order 0. The $(0,1)$-homogenization of $h\in\mathbb{K}[X]$ is h itself. The $(1,0)$-homogenization is only defined for $h\in\mathbb{K}[Y]$ and is the identity too. It is clear that for all p and q, the (p,q)-homogenization of a product h_1h_2 equals the product of the (p,q)-homogenizations of h_1 and h_2.

We define the *normalization* of a bivariate polynomial g, denoted by g°, as $g^\circ(X,Y)=X^{-\operatorname{val}_X(g)}Y^{-\operatorname{val}_Y(g)}g(X,Y)$, such that $\operatorname{val}_X(g^\circ)=\operatorname{val}_Y(g^\circ)=0$. Note that the (p,q)-homogenization of $h\in\mathbb{K}[X]$ is a normalized polynomial, and every irreducible polynomial is in particular normalized. If $g=\sum_j b_jX^{\gamma_j}Y^{\delta_j}$ is normalized and (p,q)-homogeneous $(p,q\neq 0)$ of order ω, then $q|\gamma_j$ and $p|\delta_j$ for all j. Indeed, since g is normalized, there exists j_1 such that $\gamma_{j_1}=0$. Hence $q\delta_{j_1}=\omega$ and $q|\omega$. Let us thus write $\omega=q\omega'$. Now for all j, $p\gamma_j=q\omega'-q\delta_j$, whence γ_j is divisible by q since p and q are relatively prime. In the same way, p divides δ_j.

We first show that for all p and q, one can reduce the computation of the degree-d (p,q)-homogeneous factors of f to the computation of the degree-(d/q) factors of some univariate lacunary polynomials.

Theorem 7. *Let $f=\sum_{j=1}^k c_jX^{\alpha_j}Y^{\beta_j}\in\mathbb{K}[X,Y]$. Let us write $f=f_1+\cdots+f_s$ such that each f_t is (p,q)-homogeneous, $pq\neq 0$. Then for any (p,q)-homogeneous irreducible polynomial g, $\operatorname{mult}_g(f)=\min_{1\leq t\leq s}(\operatorname{mult}_g(f_t))$.*

Moreover, let f_t° be the normalization of f_t for all t. Then

$$\operatorname{mult}_g(f_t)=\operatorname{mult}_{g(X^{1/q},1)}(f_t^\circ(X^{1/q},1)).$$

Proof. If $f=g^\mu h$, one can write $h=h_1+\cdots+h_{s'}$ as a sum of (p,q)-homogeneous parts. Then each $g^\mu h_t$ is (p,q)-homogeneous, and they have pairwise distinct orders. Hence $s'=s$ and, up to reordering, $g^\mu h_t=f_t$ for all t. Therefore, $\operatorname{mult}_g(f)\geq\min_t(\operatorname{mult}_g(f_t))$. The converse inequality is obvious.

For the second part, let us assume that f is itself (p,q)-homogeneous and normalized. As mentioned earlier, $q|\alpha_j$ and $p|\beta_j$. Therefore $f(X^{1/q},1)$

is a polynomial. Let $f_q(X) = f(X^{1/q}, 1)$ and $g_q(X) = g(X^{1/q}, 1)$ and suppose that $f = g^\mu h$. Then h is also (p,q)-homogeneous. Since f is normalized, so do h and the exponents of X in h are multiples of q. In other words, $f_q(X) = g_q(X)^\mu h(X^{1/q}, 1)$ is an equality of polynomials. Conversely, suppose that there exist h_q such that $f_q(X) = g_q^\mu(X) h_q(X)$. To prove that g^μ divides f, it suffices to (p,q)-homogenize this equality. One can easily check that the (p,q)-homogenization of f_q and g_q are f and g respectively. Thus if we denote by h the (p,q)-homogenization of h_q, $f = g^\mu h$. □

The cases $p = 0$ and $q = 0$ are treated in a similar way. If $q = 0$ for instance, g belongs to $\mathbb{K}[Y]$. Therefore, we can write $f = f_1 + \cdots + f_s$ such that each f_t is $(1,0)$-homogeneous and $\text{mult}_g(f_t) = \text{mult}_g(f_t(1,Y))$.

We can now give an algorithm to compute the degree-d quasi-homogeneous factors of a bivariate lacunary polynomial, provided we dispose of an algorithm for the computation of the degree-d factors of univariate polynomials. We assume that such an algorithm is given as an oracle.

Algorithm 1.
Input: *A lacunary polynomial $f \in \mathbb{K}[X,Y]$ and an integer d.*
Output: *The list L of the degree-d quasi-homogeneous factors of f, with their multiplicities.*
Oracle: *Given a lacunary polynomial $f_0 \in \mathbb{K}[X]$ and d, computes the degree-d factors of f.*

1. *Compute* Newt(f) *and initialize $L \leftarrow \emptyset$.*
2. *For each pair of parallel edges in* Newt(f)*, of slopes $-q/p$ with $|p|, q \le d$:*[1]
   (a) *Compute the (p,q)-homogeneous components $f_1, \dots, f_s$ of f, and their normalizations $f_1^\circ, \dots, f_s^\circ$;*
   (b) *For $t = 1$ to s:*
      i. *Compute the list $(h_1, \mu_1), \dots, (h_s, \mu_{s_t})$ of degree-(d/q) factors of $f_t^\circ(X^{1/q}, 1)$, or $f_t^\circ(1,Y)$ if $q = 0$, using the oracle;*
      ii. *Let g_u be the (p,q)-homogenization of h_u for all u;*
      iii. *Let L_t be the list of pairs (g_u, μ_u) such that* $\deg(g_u) \le d$.
   (c) *$L \leftarrow L \cup \bigcap_{t=1}^{s} L_t$.*
3. *Return L.*

In the algorithm, union and intersection have to be viewed as multisets operations: if L_1 contains (g, μ_1) and L_2 contains (g, μ_2), then $L_1 \cup L_2$ contains $(g, \max(\mu_1, \mu_2))$ and $L_1 \cap L_2$ contains $(g, \min(\mu_1, \mu_2))$.

[1] Vertical edges are said to have slope $-1/0$ by convention.

Proposition 8. *Algorithm 1 is correct. If the input polynomial has degree D and k terms, the algorithm uses at most $(k \log D + d)^{\mathcal{O}(1)}$ bit operations, and the sum of the sizes of all the univariate lacunary polynomials given to the oracle is at most $\frac{k}{2}\,\mathrm{size}(f)$.*

Proof. A (p,q)-homogeneous polynomial has a Newton polygon contained in a line of slope $-q/p$. By Ostrowski's Theorem, f can have a (p,q)-homogeneous degree-d factor only if its Newton polygon has two parallel edges of slopes $-q/p$ with $|p|, q \leq d$. (There is a degenerate case: $(0,q)$-homogeneous factors are factors depending only on the variable X and correspond to vertical edges.) Therefore, the set of pairs (p,q) is correctly computed.

Now for all the pairs (p,q) computed in the first step, the algorithm computes the (p,q)-homogeneous factors of f of degree d. The correctness of this part follows directly from Theorem 7. It is enough for the oracle to compute degree-(d/q) factors since for g a degree-d factor of f_t, $g(X^{1/q}, 1)$ has degree d/q. Note that the factors we compute may still be of degree larger than d, hence we discard the higher-degree factors.

There are at most $k/2$ pairs of parallel edges in $\mathrm{Newt}(f)$. For each such pair, since $f_1, \dots, f_s$ have a lacunary representation, $\sum_t \mathrm{size}(f_t) = \mathrm{size}(f)$, whence the result. □

4 Non-homogeneous factors

In this section, we study the factors of a bivariate lacunary polynomial whose Newton polygon is not contained in a line, that is which are not quasi-homogeneous. For a bivariate lacunary polynomial f and an irreducible polynomial g having a root $\phi \in \overline{\mathbb{K}}\langle\langle X \rangle\rangle$, we first give a bound on the valuation of $f(X, \phi(X))$ in the first section. In the second section, we use this bound to give a Gap Theorem for non-homogeneous degree-d factors of bivariate lacunary polynomials. We deduce an algorithm to reduce the computation of these factors to some bivariate low-degree factorizations.

4.1 Bounds on the valuation

The aim of this section is to prove the following theorem.

Theorem 9. *Let $g \in \mathbb{K}[X][Y]$ be an irreducible polynomial of degree d such that $\frac{\partial g}{\partial Y} \neq 0$, and $\phi \in \overline{\mathbb{K}}\langle\langle X \rangle\rangle$ be a root of g of valuation v.*

Let $f = \sum_{j=1}^{\ell} c_j X^{\alpha_j} Y^{\beta_j}$ be a polynomial with exactly ℓ terms, and suppose that the family $(X^{\alpha_j}\phi^{\beta_j})_{1\le j\le \ell}$ is linearly independent.

Then

$$\mathrm{val}\big(f(X,\phi(X))\big) \le \min_{1\le j\le \ell}(\alpha_j + v\beta_j) + (2d(4d+1)-v)\binom{\ell}{2}.$$

The proof of this theorem is based on the Wronskian of a family of series.

Definition 10. Let $f_1,\dots,f_\ell \in \overline{\mathbb{K}}\langle\langle X\rangle\rangle$. Their *Wronskian* is the determinant of the *Wronskian matrix*

$$\mathrm{wr}(f_1,\dots,f_\ell) = \det\begin{bmatrix} f_1 & f_2 & \cdots & f_\ell \\ f_1' & f_2' & \cdots & f_\ell' \\ \vdots & \vdots & & \vdots \\ f_1^{(\ell-1)} & f_2^{(\ell-1)} & \cdots & f_\ell^{(\ell-1)} \end{bmatrix}.$$

The main property of the Wronskian is its relation to linear independence. The following result is classical (see for instance [2]).

Proposition 11. *The Wronskian of $f_1,\dots,f_\ell$ is nonzero if and only if the f_j's are linearly independent over $\overline{\mathbb{K}}$.*

We first need an easy lemma, already proved in [4, 3] in the context of polynomials. The exact same proof remains valid with Puiseux series.

Lemma 12. *Let $f_1, \dots, f_\ell \in \overline{\mathbb{K}}\langle\langle X\rangle\rangle$ be Puiseux series in the variable X. Then*

$$\mathrm{val}(\mathrm{wr}(f_1,\dots,f_\ell)) \ge \sum_{j=1}^{\ell}\mathrm{val}(f_j) - \binom{\ell}{2}.$$

We aim to upper bound the valuation of the Wronskian of the family $(X^{\alpha_1}\phi^{\beta_1},\dots,X^{\alpha_\ell}\phi^{\beta_\ell})$. We need first the following lemma, borrowed from [15].

Lemma 13. *Let g, ϕ and f be as in Theorem 9, and let $g_Y = \frac{\partial g}{\partial Y}$. Then*

$$\mathrm{wr}(X^{\alpha_1}\phi^{\beta_1},\dots,X^{\alpha_\ell}\phi^{\beta_\ell}) = X^{A-\binom{\ell}{2}}\phi^{B-\binom{\ell}{2}}\frac{h_\ell(X,\phi)}{g_Y^{\ell(\ell-1)}(X,\phi)}$$

where $A = \sum_j \alpha_j$, $B = \sum_j \beta_j$ and h_ℓ is a polynomial of degree $(1+2d)\binom{\ell}{2}$ in each variable.

It remains to obtain a valuation bound for a Puiseux series in terms of a vanishing polynomial.

Lemma 14. *Let g and ϕ be as in Theorem 9. Let $h(X, Y)$ be a polynomial of degree at most δ in each variable. Then $|\mathrm{val}(h(X, \phi))| \leq 2d\delta$.*

Proof. Let us consider the resultant

$$r(X, Y) = \mathrm{res}_Z(g(X, Z), Y - h(X, Z)).$$

Then $r(X, h(X, \phi)) = 0$ vanishes since ϕ is a common root of both polynomials in the resultant.

Let us now consider the degree of r in X. The coefficients of $g(X, Z)$ viewed as a polynomial in Z have degree at most d in X by definition. In the Sylvester matrix, δ rows are made with the coefficients of g since $Y - h(X, Z)$ has degree δ in Z. In the same way, the Sylvester matrix contains d rows with the coefficients of $Y - h(X, Z)$, each of which has degree at most δ in X. Altogether, each term in the resultant has degree at most $2d\delta$ in X.

We have shown that $h(X, \phi)$ is a Puiseux series which cancels a polynomial r of degree at most $2d\delta$ in X. By Newton-Puiseux Theorem, the absolute value of its valuation is at most $2d\delta$. □

We have all the ingredients to prove Theorem 9.

of Theorem 9. Let W be the Wronskian of the family $(X^{\alpha_1}\phi^{\beta_1}, \ldots, X^{\alpha_\ell}\phi^{\beta_\ell})$ and $\psi = f(X, \phi)$. Without loss of generality, let us assume that $\min_j(\alpha_j + v\beta_j)$ is attained for $j = 1$.

Using column operations on the Wronskian matrix, one can replace the first column by ψ and its derivatives. The determinant of the new matrix is the Wronskian W_ψ of ψ, $X^{\alpha_2}\phi^{\beta_2}$, ..., $X^{\alpha_\ell}\phi^{\beta_\ell}$. We have $W_\psi = a_1 W$ and their valuations coincide. By Lemma 12,

$$\mathrm{val}(W_\psi) \geq \mathrm{val}(\psi) + \sum_{j>1}(\alpha_j + v\beta_j) - \binom{\ell}{2}.$$

On the other hand, since the family $(X^{\alpha_j}\phi^{\beta_j})_j$ is linearly independent, there exists a nonzero h_ℓ such that

$$W = X^{A-\binom{\ell}{2}}\phi^{B-\binom{\ell}{2}}\frac{h_\ell(X, \phi)}{g_Y^{\ell(\ell-1)}(X, \phi)}$$

according to Lemma 13. Moreover $\mathrm{val}(h_\ell(X,\phi)) \leq 2d(2d+1)\binom{\ell}{2}$ and $\mathrm{val}(g_Y(X,\phi)) \geq -2d^2$ by Lemma 14. Therefore,

$$\mathrm{val}(W) \leq A - \binom{\ell}{2} + v\left(B - \binom{\ell}{2}\right) + 2d(4d+1)\binom{\ell}{2}.$$

Since $A = \sum_j \alpha_j$ and $B = \sum_j \beta_j$,

$$\mathrm{val}(\psi) \leq \alpha_1 + v\beta_1 - v\binom{\ell}{2} + 2d(4d+1)\binom{\ell}{2}.$$

The conclusion follows, since $\alpha_1 + v\beta_1 = \min_j(\alpha_j + v\beta_j)$. □

4.2 Gap Theorem and algorithm

Theorem 15 (Gap Theorem). *Let $f = f_1 + f_2$, where*

$$f_1 = \sum_{j=1}^{\ell} c_j X^{\alpha_j} Y^{\beta_j} \quad \text{and} \quad f_2 = \sum_{j=\ell+1}^{k} c_j X^{\alpha_j} Y^{\beta_j}$$

satisfy $\alpha_j + v\beta_j \leq \alpha_{j+1} + v\beta_{j+1}$ for $1 \leq j < k$. Assume that ℓ is the smallest index such that

$$\alpha_{\ell+1} + v\beta_{\ell+1} > (\alpha_1 + v\beta_1) + (2d(4d+1) - v)\binom{\ell}{2}.$$

Then for every irreducible polynomial g of degree at most d such that g has a root of valuation v in $\overline{\mathbb{K}}\langle\langle X\rangle\rangle$,

$$\mathrm{mult}_g(f) = \min(\mathrm{mult}_g(f_1), \mathrm{mult}_g(f_2)).$$

Proof. Let us view g as a polynomial in $\mathbb{K}[X][Y]$, and let $\phi \in \overline{\mathbb{K}}\langle\langle X\rangle\rangle$ be a root of g of valuation v. Then g divides f (resp. f_1, resp. f_2) if and only if $f(X,\phi) = 0$ (resp. $f_1(X,\phi) = 0$, resp. $f_2(X,\phi) = 0$). And if g divides both f_1 and f_2, it divides f. Let us assume that g does not divide f_1 and prove that in such a case, it does not divide f either. Let $\Delta = 2d(4d+1) - v$.

Since g does not divide f_1, $f_1(X,\phi)$ is nonzero. We can extract a basis from the family $(X^{\alpha_j}\phi^{\beta_j})_{1\leq j\leq \ell}$ and rewrite f_1 on this basis:

$$f_1 = \sum_{t=1}^{m} b_t X^{\alpha_{j_t}} Y^{\beta_{j_t}}$$

where the new coefficients are linear combinations of $c_1, \ldots, c_\ell$. Without loss of generality, we assume that $b_t \neq 0$ for all t. Using Theorem 9, the valuation of $f_1(X, \phi)$ is bounded by $\alpha_{j_1} + v\beta_{j_1} + \Delta\binom{m}{2}$. Furthermore, by minimality of ℓ, $\alpha_{j_1} + v\beta_{j_1} \leq \alpha_1 + v\beta_1 + \Delta\binom{j_1-1}{2}$. Thus

$$\operatorname{val}(f_1(X, \phi)) \leq \alpha_1 + v\beta_1 + \Delta\left(\binom{j_1-1}{2} + \binom{m}{2}\right).$$

Since $j_1 + m - 1 \leq \ell$, we deduce that $\operatorname{val}(f_1(X, \phi)) \leq \alpha_1 + v\beta_1 + \Delta\binom{\ell}{2}$ by superadditivity of the function $\ell \mapsto \binom{\ell}{2}$.

Now, $\operatorname{val}(f_2(X, \phi)) \geq \alpha_{\ell+1} + v\beta_{\ell+1} > \operatorname{val}(f_1(X, \phi))$ by hypothesis. Hence $f(X, \phi) = f_1(X, \phi) + f_2(X, \phi)$ cannot vanish. That is, g does not divide f.

To obtain the conclusion on the multiplicity of g as a factor of f, it remains to apply the same reasoning with the successive derivatives of f, f_2 and f_1. The point is that these derivatives have the same form as f, f_2 and f_1. □

In the Gap Theorem, we assumed that $\alpha_j + v\beta_j \leq \alpha_{j+1} + v\beta_{j+1}$ for all j. That is, we put an order on the monomials which depends on the value of v. Since we aim to use this theorem with different values on v, we restate it without referring to the order: Let $\mathcal{I} = \{1, \ldots, k\}$, and suppose that $\mathcal{I}$ can be partitioned into $\mathcal{I}_1 \sqcup \mathcal{I}_2$ such that $\mathcal{I}_1 = \{i \in \mathcal{I} : \alpha_i + v\beta_i \leq \min_j(\alpha_j + v\beta_j) + \Delta\binom{\ell}{2}\}$ where $\Delta = 2d(4d+1) - v$. Then any degree-d polynomial g which has a root of valuation v in $\overline{\mathbb{K}}\langle\langle X\rangle\rangle$ satisfies $\operatorname{mult}_g(f) = \min(\operatorname{mult}_g(f_{|\mathcal{I}_1}), \operatorname{mult}_g(f_{|\mathcal{I}_2}))$, where $f_{|\mathcal{I}_1} = \sum_{j \in \mathcal{I}_1} c_j X^{\alpha_j} Y^{\beta_j}$ and $f_{|\mathcal{I}_2}$ is defined similarly.

It is straightforward to extend the Gap Theorem to a partition of $\mathcal{I}$ into subsets $\mathcal{I}_1, \ldots, \mathcal{I}_s$, using recursion: Let us rename $\mathcal{I}_2$ into $\mathcal{J}$. Suppose we have partitioned $\mathcal{I}$ as $(\bigsqcup_{u=1}^{t} \mathcal{I}_u) \sqcup \mathcal{J}$. We can partition $\mathcal{J} = \mathcal{J}_1 \sqcup \mathcal{J}_2$ using the Gap Theorem with $f_{|\mathcal{J}}$. Then let $\mathcal{I}_{t+1} = \mathcal{J}_1$ and $\mathcal{J} = \mathcal{J}_2$. When the Gap Theorem stops working because there is no more gap, let $\mathcal{I}_s = \mathcal{J}$. For all t and all $j_1, j_2 \in \mathcal{I}_t$,

$$\left|(\alpha_{j_1} + v\beta_{j_1}) - (\alpha_{j_2} + v\beta_{j_2})\right| \leq \Delta\binom{|\mathcal{I}_t| - 1}{2}.$$

For $1 \leq t \leq s$, let $f_t = f_{|\mathcal{I}_t}$. The previous construction together with the Gap Theorem ensures that $\operatorname{mult}_g(f) = \min_t(\operatorname{mult}_g(f_t))$. Our goal is to refine the partition of $\mathcal{I}$ into smaller subsets such that the polynomials obtained from this partition after normalization have low degree.

We first prove an easy lemma useful to give bounds in the next theorem.

Lemma 16. *Let $v_1 = p_1/q_1$ and $v_2 = p_2/q_2$ two rational numbers such that $0 < p_1, q_1, p_2, q_2 \leq d$ and $v_1 > v_2$.*

Then $1/(v_1 - v_2) \leq d^2$ and $(v_1 + v_2)/(v_1 - v_2) \leq 2d^2$.

Proof. We have

$$\frac{p_1}{q_1} - \frac{p_2}{q_2} = \frac{p_1q_2 - p_2q_1}{q_1q_2}$$

and since $v_1 > v_2$, the numerator is a nonzero integer and $v_1 - v_2 \geq 1/d^2$. Similarly,

$$\frac{v_1 + v_2}{v_1 - v_2} = \frac{p_1q_2 + p_2q_1}{p_1q_2 - p_2q_1} \leq 2d^2. \qquad \square$$

Theorem 17. *Let $f, g \in \mathbb{K}[X, Y]$ such that f has k monomials and g has a degree d and is not quasi-homogeneous. There exists a deterministic algorithm that computes in time polynomial in k and d a set of at most k polynomials f_1°, ..., f_s°, such that each f_t° has ℓ_t nonzero terms, with $\sum_t \ell_t = k$, and degree at most $\mathcal{O}(d^4\binom{\ell_t - 1}{2})$, and such that*

$$\text{mult}_g(f) = \min_{1 \leq t \leq s}(\text{mult}_g(f_t^\circ)).$$

Proof. Since g is not quasi-homogeneous, its Newton polygon is not contained in a line. Therefore, it has at least two non-parallel edges e_1 and e_2. At least one of them, say e_1, belongs to the lower hull of $\text{Newt}(g)$. If its slope is $-v_1$, g has a root in $\overline{\mathbb{K}}\langle\langle X\rangle\rangle$ of valuation v_1 by Newton-Puiseux Theorem. Let $\mathcal{I} = \{1, \dots, k\}$. Using the Gap Theorem with $v = v_1$, one can partition $\mathcal{I} = \mathcal{I}_1 \sqcup \cdots \sqcup \mathcal{I}_{s'}$. The idea is to use a second time the Gap Theorem, with the second edge, to partition each $\mathcal{I}_t$. Let $f_t = f_{|\mathcal{I}_t}$ for all t.

There are three cases to handle, depending on the position of the edge e_2. It belongs either to the lower hull as e_1, or to the upper hull, or it is vertical. We suppose for the two first cases that the slope of e_2 is v_2. To simplify notations, let us define $D = 2d(4d + 1)$, $\Delta_1 = D - v_1$ and $\Delta_2 = D - v_2$.

The first case is simple: We apply in a straightforward way the Gap Theorem. This yields a partition of each $\mathcal{I}_t$ as $\mathcal{I}_{t,1} \sqcup \cdots \sqcup \mathcal{I}_{t,s_t}$. Consider one subset $\mathcal{I}_{t,u}$ and the corresponding polynomial $f_{t,u} = f_{|\mathcal{I}_{t,u}}$. Let us assume without loss of generality that $\alpha_i + v_i\beta_i = \min_{j \in \mathcal{I}_{t,u}}(\alpha_j + v_i\beta_j)$ for $i = 1, 2$. Then for all $j \in \mathcal{I}_{t,u}$ and for $i = 1, 2$, $\alpha_j + v_i\beta_j \leq \alpha_i + v_i\beta_i + \Delta_i\binom{\ell}{2}$.

Let $\ell_{t,u} = |\mathcal{I}_{t,u}|$. Then for all $p, q \in \mathcal{I}_{t,u}$,

$$\begin{aligned}\alpha_p - \alpha_q &= (\alpha_p - \alpha_1) + (\alpha_1 - \alpha_q)\\ &\leq v_1(\beta_1 - \beta_p) + \Delta_1 \binom{\ell_{t,u} - 1}{2} + v_1(\beta_q - \beta_1)\\ &\leq v_1(\beta_q - \beta_p) + \Delta_1 \binom{\ell_{t,u} - 1}{2}.\end{aligned}$$

This inequality remains true if we replace v_1 by v_2 and if p and q are exchanged. In other words,

$$\alpha_q - \alpha_p \leq v_2(\beta_p - \beta_q) + \Delta_2 \binom{\ell_{t,u} - 1}{2}.$$

We can sum both equations and reorganize to obtain

$$(\beta_p - \beta_q)(v_1 - v_2) \leq (\Delta_1 + \Delta_2) \binom{\ell_{t,u} - 1}{2}.$$

Since p and q can once again be exchanged, we conclude that for all p and q,

$$|\beta_p - \beta_q| \leq \frac{\Delta_1 + \Delta_2}{|v_1 - v_2|} \binom{\ell_{t,u} - 1}{2}.$$

Using very similar argument, one easily shows that

$$|\alpha_p - \alpha_q| \leq \frac{|v_1|\Delta_2 + |v_2|\Delta_1}{|v_1 - v_2|} \binom{\ell_{t,u} - 1}{2}.$$

By Lemma 16, $|\alpha_p - \alpha_q|, |\beta_p - \beta_q| \leq \mathcal{O}(d^4 \binom{\ell_{t,u}-1}{2})$. Therefore the polynomial $f_{t,u}^{\circ}$ obtained after normalization of $f_{t,u}$ has $\ell_{t,u}$ nonzero terms and degree at most $\mathcal{O}(d^4 \binom{\ell_{t,u}-1}{2})$. The theorem follows, with $s = \sum_t s_t$.

It remains to deal with the two other cases. The second case is quite similar to the first one. Since e_2 is in the upper hull of $\mathrm{Newt}(g)$, we consider for all t the reciprocal f_t^X of f_t with respect to the variable X, defined by $f_t^X(X, Y) = X^{\deg_X(f_t)} f_t(1/X, Y) = \sum_{j \in \mathcal{I}_t} c_j X^{\gamma_j} Y^{\beta_j}$ where $\gamma_j = \deg_X(f_t) - \alpha_j$ for all j. We define g^X similarly. Then g^X is irreducible since this is the case of g, it has the same degree as g, and $\mathrm{mult}_g(f_t) = \mathrm{mult}_{g^X}(f_t^X)$. We can apply the Gap Theorem to f_t^X with respect to e_2. The slope of e_2 in $\mathrm{Newt}(g^X)$ is the same as the slope of the corresponding edge in $\mathrm{Newt}(g)$. Therefore, the edge e_2 corresponds to a root of g^X of valuation v_2 in $\overline{\mathbb{K}}\langle\langle X \rangle\rangle$. Thus we partition $\mathcal{I}_t = \mathcal{I}_{t,1} \sqcup \cdots \sqcup \mathcal{I}_{t,s_t}$. Using the same computations as in the first case, we obtain that $|\gamma_p - \gamma_q|, |\beta_p - \beta_q| \leq \mathcal{O}(d^4\binom{\ell_{t,u}}{2})$ for all $p, q \in \mathcal{I}_{t,u}$. This implies the same bound on $|\alpha_p - \alpha_q|$. The theorem follows using the same arguments as in the first case, defining $f^\circ_{t,u}$ as the normalization of $f_{|\mathcal{I}_{t,u}}$ for all t and u.

The third case corresponds to e_2 being vertical. Note that we can assume without loss of generality that e_2 in *on the left* of the Newton polygon. Indeed, we need to use this third case only if the first two cases cannot be applied. In this case, the lower hull and the upper hull contain each an only edge, and these both edges are parallel. Therefore, there must exist in the Newton polygon two vertical edges and we can choose the one on the left. Now, we can apply the Gap Theorem by inverting the roles of X and Y. That is, we can consider $\bar{f}_t(X, Y) = f_t(Y, X)$ and $\bar{g}(X, Y) = g(Y, X)$. Then, the left vertical edge in $\mathrm{Newt}(g)$ corresponds to an horizontal edge in the lower hull of $\mathrm{Newt}(\bar{g})$. As in the first two cases, we can then partition each $\mathcal{I}_t$ to finally define some polynomials $f^\circ_{t,u}$. Hence, the degree in Y of each $f^\circ_{t,u}$ is bounded by $D\binom{\ell_{t,u}-1}{2}$ since an horizontal edge corresponds to a root of valuation 0, and the degree in X by $\mathcal{O}(d^3\binom{\ell_{t,u}-1}{2})$. □

Algorithm 2.
Input: *A lacunary polynomial $f \in \mathbb{K}[X, Y]$ and an integer d.*
Output: *The list L of the degree-d non-homogeneous factors of f, with their multiplicities.*
Oracle: *Given a low-degree polynomial $g \in \mathbb{K}[X, Y]$, computes the factorization of g.*

1. *Compute* $\mathrm{Newt}(f)$ *and initialize $L \leftarrow \emptyset$;*
2. *For each pair of non-parallel edges in* $\mathrm{Newt}(f)$*:*
   (a) *Compute $f^\circ_1, \dots, f^\circ_s$ according to Theorem 17;*
   (b) *Compute the list L_t of degree-d factors of each f°_t, using the oracle;*
   (c) *$L \leftarrow L \cup \bigcap_{t=1}^s L_t$.*
3. *Return L.*

Proposition 18. *Algorithm 2 is correct. If f has degree D and k nonzero terms, the algorithms uses at most $(k \log D + d)^{\mathcal{O}(1)}$ bit operations, and the sum of the degrees of the bivariate polynomials given to the oracle is at most $\mathcal{O}(d^4k^4)$.*

Proof. The correctness follows from Ostrowski's Theorem and Theorem 17.

Furthermore, for each pair of edges, the polynomials $f^\circ_1, \dots, f^\circ_s$ have degree at most $\mathcal{O}(d^4\binom{\ell_t-1}{2})$ for all $1 \leq t \leq s$, with $\sum_t \ell_t = k$. By superadditivity of the function $\ell \mapsto \binom{\ell}{2}$, $\sum_t \deg(f^\circ_t) \leq \mathcal{O}(d^4\binom{k-1}{2})$. Since there are at most $\binom{k}{2}$ pairs of distinct edges, the result follows. □

5 Multivariate polynomials

To extend our method to multivariate polynomials $f \in \mathbb{K}[X_1, \dots, X_n]$, a first idea consists in considering the n-dimensional Newton polytope of f. Yet the computation of the Newton polytope is not polynomial in n. Actually, we will use the $n(n-1)$ possible 2-dimensional Newton polygons. For, we extend our definition of $\mathrm{Newt}(f)$: If $i_1 \neq i_2$, $\mathrm{Newt}_{i_1,i_2}(f)$ is the Newton polygon of f viewed as an element of $R[X_{i_1}, X_{i_2}]$ where R is the polynomial ring in the $(n-2)$ other variables over $\mathbb{K}$.

As for the case of bivariate polynomials, there exists a special case. This special case corresponds to factors g whose n-dimensional support is contained in a line (and thus is 1-dimensional). As for quasi-homogeneous factors in the bivariate case, the computation of these factors reduces to univariate lacunary polynomials. Let us call these polynomials *unidimensional polynomials*. Note first that for such a factor g, $\mathrm{Newt}_{i_1,i_2}(g)$ is contained in a line for all i_1 and i_2. Consider the Newton polygons $\mathrm{Newt}_{1,i}$ for all $i > 1$. If f has a unidimensional factor g depending on X_1, there exists a corresponding pair of parallel edges in each $\mathrm{Newt}_{1,i}(g)$, which are horizontal if g does not depend on i. Actually, these pairs of edges correspond to a same pair of edges in the n-dimensional Newton polytope of g. The algorithm to compute unidimensional factors *depending on* X_1 is as follows: Consider all the parallel edges in $\mathrm{Newt}_{1,2}(f)$. For each such pair, pick one of the edges (say in the lower hull or on the left if it is vertical) and denote by (a_1, a_2) and (b_1, b_2) its endpoints. Then, each $\mathrm{Newt}_{1,i}(f)$ should have an edge of endpoints (a_1, a_i) and (b_1, b_i) for some a_i and b_i, as well as an edge parallel to this one if a_i and b_i are not both zero (in which case we are considering a factor which does not depend on X_i). Thus for each pair of parallel edges of $\mathrm{Newt}_{1,2}(f)$, we check if the corresponding edges exist in $\mathrm{Newt}_{1,i}(f)$ for $i > 2$. Now if we view f as a polynomial in X_1 and X_2, it is quasi-homogeneous and we can apply the algorithm for bivariate polynomials to eliminate the variable X_2. In the same way we eliminate all the variables X_i for $i = 2$ to n and we get univariate lacunary polynomials. If we have an oracle computing their low-degree factors, we can reconstruct, as in the bivariate case, the corresponding unidimensional factors, variable by variable. This gives all the factors depending on X_1. We apply the same algorithm forgetting the variable X_1 and replacing its role by X_2 to compute the factors depending on X_2 and not on X_1. We continue with all variables to get all the unidimensional factors. The running time of this algorithm is polynomial in n, k and $\log(D)$ where k is the number of nonzero terms in f and D its degree.

Let us now consider a *multidimensional* factor g, that is a factor whose support is not contained in a line. Then for every variable X_{i_1}, there exists at least one variable X_{i_2} such that $\mathrm{Newt}_{i_1,i_2}(g)$ is not contained in a line, but in one case: if g does not depend on X_{i_1}. The idea of the algorithm is the following: For all variables X_i, $i > 1$, consider the Newton polygons $\mathrm{Newt}_{1,i}(f)$. For each i, partition the set $\mathcal{I} = \{1, \ldots, k\}$ into $\mathcal{I}_1 \sqcup \cdots \sqcup \mathcal{I}_s$ according to the pairs of non-parallel edges, as in the proof of Theorem 17. Thus, we have $(n-1)$ partitions of $\mathcal{I}$. The idea is now to merge these partitions to build a unique partition. For, suppose we have two partitions $\mathcal{I} = \bigsqcup_t \mathcal{J}_t^1$ and $\mathcal{I} = \bigsqcup_t \mathcal{J}_t^2$ that we want to merge. We define a new partition $\mathcal{I} = \bigsqcup_t \mathcal{I}_t$ recursively. Let $\mathcal{I}_1 = \{1\}$. Then, for every $j \in \mathcal{I}_1$, if $j \in \mathcal{J}_t^1$ and $j \in \mathcal{J}_{t'}^2$, we replace $\mathcal{I}_1$ by $\mathcal{I}_1 \cup \mathcal{J}_t^1 \cup \mathcal{J}_{t'}^2$. Once every index j in $\mathcal{I}_1$ has be treated, we take the smallest index $j \notin \mathcal{I}_1$ and define $\mathcal{I}_2 = \{j\}$. We apply the same algorithm to $\mathcal{I}_2$ and recursively build a partition of $\mathcal{I}$.

If two distinct indices j_1 and j_2 belong to a same subset $\mathcal{J}_t^1$ (or $\mathcal{J}_t^2$) of a partition, we have $|\alpha_{1,j_1} - \alpha_{1,j_2}| \leq Cd^4|\mathcal{J}_t^1|^2$ for some constant C (cf. Theorem 17). Consider then two indices j_1 and j_2 in a same subset $\mathcal{I}_t$ of the new partition. They can be joined by a path of indices such that two consecutive indices in this path belong to a same $\mathcal{J}_t^1$ or a same $\mathcal{J}_t^2$. In other words, there exist indices $u_1 = j_1$, u_2, …, $u_{2m} = j_2$ such that $u_1, u_2 \in \mathcal{J}_{t_1}^1$, $u_3, u_4 \in \mathcal{J}_{t_3}^1$, …, $u_{2m-1}, u_{2m} \in \mathcal{J}_{t_{2m-1}}^1$ on the one hand, and $u_2, u_3 \in \mathcal{J}_{t_2}^2$, …, $u_{2m-2}, u_{2m-1} \in \mathcal{J}_{t_{2m-2}}^2$ on the other hand, for some t_1, …, t_{2m-1}. Then,

$$\begin{aligned} |j_2 - j_1| &\leq \sum_{p=1}^{2m-1} |\alpha_{1,u_p} - \alpha_{1,u_{p+1}}| \\ &\leq Cd^4(|\mathcal{J}_{t_1}^1|^2 + |\mathcal{J}_{t_2}^2|^2 + \cdots + |\mathcal{J}_{t_{2m-1}}^1|^2). \end{aligned}$$

We can assume without loss of generality that all the $\mathcal{J}_{t_p}^1$ are pairwise distinct, as well as the $\mathcal{J}_{t_p}^2$. Since the sum of the sizes of the $\mathcal{J}_t^1$, respectively of the $\mathcal{J}_t^2$, is bounded by k, and since the function $k \mapsto k^2$ is superadditive, $|\alpha_{1,j_2} - \alpha_{1,j_1}| \leq 2Cd^4k^2$. This means that we can merge all partitions built using the Newton polygons $\mathrm{Newt}_{1,i}(f)$ to get a new partition $\mathcal{I} = \mathcal{I}_1 \sqcup \cdots \sqcup \mathcal{I}_s$ such that for all t and $j_1, j_2 \in \mathcal{I}_t$, $|\alpha_{1,j_1} - \alpha_{1,j_2}| \leq \mathcal{O}(nd^4k^2)$.

This new partition has the property that if we define the normalized polynomials $f_t^\circ = f_{|\mathcal{I}_t}^\circ$ for all t, then $\mathrm{mult}_g(f) = \min_t(\mathrm{mult}_g(f_t^\circ))$ for all degree-d multidimensional polynomial depending on X_1. To include factors which do not depend on X_1, we simply have to ensure that two indices j_1 and j_2 such that $\alpha_{1,j_1} = \alpha_{1,j_2}$ belong to the same subset. This can be done by merging the partition $\mathcal{I}_1 \sqcup \cdots \sqcup \mathcal{I}_s$ with the partition induced by the equalities on $\alpha_{1,j}$. The bound on $|\alpha_{1,j_1} - \alpha_{1,j_2}|$ remains valid.

Now, we can replace X_1 by X_2 and refine the partition we have with the same algorithm, and so on with all variables. Let $\mathcal{I} = \mathcal{I}_1 \sqcup \cdots \mathcal{I}_s$ be the final partition and let f_t° be the normalization of $f_{|\mathcal{I}_t}$ for all t. The degree of f_t° is at most $\mathcal{O}(nd^4k^2)$ in each variable, and for any irreducible multidimensional polynomial g of degree at most d, $\text{mult}_g(f) = \min_t(\text{mult}_g(f_t^\circ))$. It only remains to factorize these low-degree polynomials.

References

[1] S. S. Abhyankar. *Algebraic Geometry for Scientists and Engineers*, volume 35 of *Mathematical surveys and monographs*. Am. Math. Soc., 1990.

[2] A. Bostan and P. Dumas. Wronskians and linear independence. *Am. Math. Mon.*, 117(8):722–727, 2010.

[3] A. Chattopadhyay, B. Grenet, P. Koiran, N. Portier, and Y. Strozecki. Computing the multilinear factors of lacunary polynomials without heights. Manuscript (submitted), 2013. arXiv:1311.5694.

[4] A. Chattopadhyay, B. Grenet, P. Koiran, N. Portier, and Y. Strozecki. Factoring bivariate lacunary polynomials without heights. In *Proc. ISSAC'13*, pages 141–158, 2013. arXiv:1206.4224.

[5] A. Chistov. Algorithm of polynomial complexity for factoring polynomials over local fields. *J. Math. Sci.*, 70(4):1912–1933, 1994.

[6] G. Chèze and A. Galligo. Four lectures on polynomial absolute factorization. In A. Dickenstein and I. Z. Emiris, editors, *Solving Polynomial Equations*, volume 14 of *Algorithms Comput. Math.*, pages 339–392. 2005.

[7] F. Cucker, P. Koiran, and S. Smale. A polynomial time algorithm for Diophantine equations in one variable. *J. Symb. Comput.*, 27(1):21–30, 1999.

[8] M. de Berg, M. van Kreveld, M. Overmars, and O. C. Schwarzkopf. *Computational Geometry*. Springer, 2000.

[9] E. Kaltofen. Polynomial-Time Reductions from Multivariate to Bi- and Univariate Integral Polynomial Factorization. *SIAM J. Comput.*, 14(2):469–489, 1985.

[10] E. Kaltofen. Factorization of polynomials given by straight-line programs. In S. Micali, editor, *Randomness and Computation*, volume 5 of *Advances in Computing Research*, pages 375–412. 1989.

[11] E. Kaltofen and P. Koiran. On the complexity of factoring bivariate supersparse (lacunary) polynomials. In *Proc. ISSAC'05*, pages 208–215. ACM, 2005.

[12] E. Kaltofen and P. Koiran. Finding small degree factors of multivariate supersparse (lacunary) polynomials over algebraic number fields. In *Proc. ISSAC'06*, pages 162–168. ACM, 2006.

[13] E. Kaltofen, J. P. May, Z. Yang, and L. Zhi. Approximate factorization of multivariate polynomials using singular value decomposition. *J. Symb. Comput.*, 43(5):359–376, 2008.

[14] E. Kaltofen and B. Trager. Computing with polynomials given by black boxes for their evaluations: Greatest common divisors, factorization, separation of numerators and denominators. *J. Symb. Comput.*, 9(3):301–320, 1990.

[15] P. Koiran, N. Portier, and S. Tavenas. A Wronskian approach to the real τ-conjecture. Manuscript (submitted), 2012. Presented at MEGA 2013. arXiv:1205.1015.

[16] H. Lenstra Jr. On the factorization of lacunary polynomials. In *Number theory in progress*, pages 277–291. De Gruyter, 1999.

[17] B. Sturmfels. Polynomial equations and convex polytopes. *Am. Math. Mon.*, 105(10):907–922, 1998.

[18] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Camb. U. Press, 2nd edition, 2003.