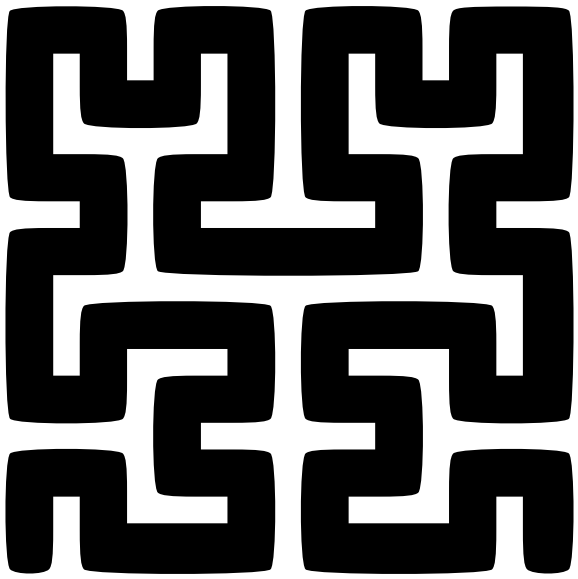

Polynesian Journal of Mathematics

Volume 2, Issue 4

Improvements of convex-dense factorization of bivariate polynomials

Martin Weimann

Received 12 Dec 2024
Revised 2 Jul 2025
Accepted 22 Jul 2025
Published 24 Jul 2025
Communicated by Roger Oyono

DOI: 10.69763/polyjmath.2.4

Polynesian Journal of Mathematics
Volume 2, Issue 4 (Jul. 2025)

Improvements of convex-dense factorization of bivariate polynomials

Martin Weimann

Laboratoire de mathématiques Nicolas Oresme
Université de Caen-Normandie, France

Abstract

We develop a new algorithm for factoring a bivariate polynomial $F \in \mathbb{K}[x, y]$ which takes full advantage of the geometry of the Newton polygon of F. Under some non degeneracy hypothesis, the complexity is $\tilde{O}(V r_0^{\omega-1})$ where V is the volume of the polygon and r_0 is its minimal lower lattice length. The integer r_0 reflects some combinatorial constraints imposed by the polygon, giving a reasonable and easy-to-compute upper bound for the number of non trivial indecomposable Minkowski summands. The proof is based on a new fast factorization algorithm in $\mathbb{K}[[x]][y]$ with respect to a slope valuation, a result which has its own interest.

1 Introduction

Factoring a bivariate polynomial $F \in \mathbb{K}[x, y]$ over a field $\mathbb{K}$ is a fundamental task of computer algebra which has received particular attention since the 1970s. We refer the reader to [10, Chapter III] and [6, 7, 11, 13] for a detailed historical account and an extended bibliography on the subject. For a dense polynomial of bidegree (d_x, d_y), the current complexity is $O(d_x d_y^\omega)$ plus one univariate factorization of degree d_y [11, 13]. Here, $2 \leqslant \omega \leqslant 3$ is so that we can multiply $n \times n$ matrices over $\mathbb{K}$ with $O(n^\omega)$ operations in $\mathbb{K}$. The current theoretical bound is $\omega \approx 2.371$ [25], although ω is in practice closer to 3 in most software implementations.

In this paper, we will focus on finer complexity indicators attached to the Newton polygon $N(F)$, convex hull of the set of exponents of F. The polynomial F is assumed to be represented by the list of its coefficients associated to the lattice points of $N(F)$, including zero coefficients. Following [2], we talk of *convex-dense* representation. Assuming $N(F)$ is two-dimensional, the size of F can also be measured as the Euclidean volume V of $N(F)$ by Pick's formula.

Various convex-dense factorization algorithms have been proposed in the last two decades, see e.g. [1, 2, 22, 23] and references therein. In [2], the authors compute in softly linear time a map $\tau \in \mathrm{Aut}(\mathbb{Z}^2)$ so that the volume of $\tau(N(F))$ is comparable to the volume of its bounding rectangle. Applying a classical dense algorithm on the resulting

polynomial $\tau(F)$, they get a complexity estimate $O(Vn^{\omega-1})$ where n is the width of the bounding rectangle, thus recovering the usual cost if F is a dense polynomial. However, this algorithm does not take advantage of the combinatorial constraints imposed by Ostrowski's theorem, namely

$$N(GH) = N(G) + N(H)$$

where + indicates Minkowski sum. Regarding this issue, we developed in [22, 23] some convex-dense algorithms based on toric geometry which take full advantage of Ostrowski's combinatorial constraints. Unfortunately, these algorithms only work in characteristic zero and the complexity is not optimal.

In this note, we intend to show that under some non degeneracy hypothesis, it is in fact possible to take into account both the volume and Ostrowski's constraints, including arbitrary characteristic. Our complexity improves [2], the gain being particularly significant when $N(F)$ has few Minkowski summands.

Complexity model. We work with computation trees [3, Section 4.4]. We use an algebraic RAM model, counting only the number of arithmetic operations in $\mathbb{K}$. We classically denote $O()$ and $\tilde{O}()$ to respectively hide constant and logarithmic factors in our complexity results ; see e.g. [10, Chapter 25, Section 7]. We use fast multiplication of polynomials, so that two polynomials in $\mathbb{K}[x]$ of degree at most d can be multiplied in softly linear time $\tilde{O}(d)$.

1.1 Fast convex-dense factorization

Let $P \subset \mathbb{R}^2$ be a lattice polygon. Let $\Lambda(P)$ be the *lower boundary* of P, union of edges whose inward normal vectors have strictly positive second coordinate. The *(lower) lattice length* of P is

$$r(P) := \mathrm{Card}\left(\Lambda(P) \cap \mathbb{Z}^2\right) - 1.$$

As $r(PQ) = r(P) + r(Q)$, this integer gives an easy-to-compute upper bound for the number of indecomposable Minkowski summands of P which are not a vertical segment (computing all Minkowski sum decompositions is NP-complete [9]).

Let $F = \sum c_{ij}x^j y^i \in \mathbb{K}[x^{\pm 1}, y^{\pm 1}]$. The support of F is the set of exponents $(i, j) \in \mathbb{Z}^2$ such that $c_{ij} \neq 0$. Notice that the exponents of y are represented by the horizontal axis. The Newton polygon $N(F)$ of F is the convex hull of its support and we denote for short $\Lambda(F)$ its lower boundary.

Definition 1.1. *We say that F is not degenerated if for all edge $E \subset \Lambda(F)$, the* edge polynomial $y^{-\operatorname{ord}_y(F_E)}F_E$ *is separable in y, where $F_E := \sum_{(i,j)\in E\cap\mathbb{Z}^2} c_{ij}x^j y^i$.*

Note that $F_E \in \mathbb{K}[x^{\pm 1}][y]$ is quasi-homogeneous, hence its factorization reduces to a univariate factorization of degree the lattice length of E.

Let us denote for short $V = \mathrm{Vol}(N(F))$ and $r = r(N(F))$. Note that $r \leqslant d_y$. Due to Ostrowski's theorem, r is an upper bound for the numbers of irreducible factors of F of positive y-degree. Our main result is the following.

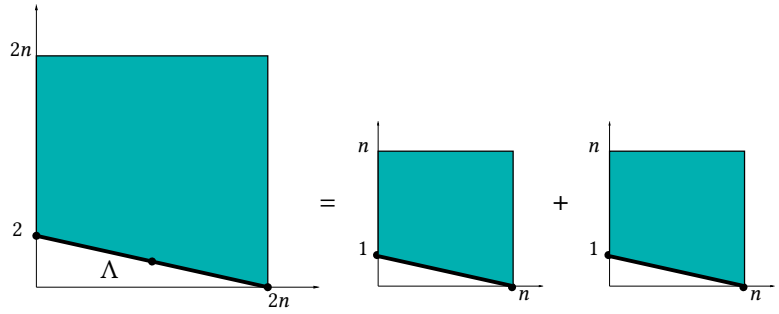

Figure 1: A Newton polygon which admits a unique Minkowski sum decomposition.

Theorem 1.2. *There exists a deterministic algorithm which given $F \in \mathbb{K}[x,y]$ non degenerated, computes the irreducible factorization of F over $\mathbb{K}$ with*

1. *$\tilde{O}(rV) + O(r^{\omega-1}V)$ operations in $\mathbb{K}$ if $p = 0$ or $p \geqslant 4V$, or*
2. *$\tilde{O}(kr^{\omega-1}V)$ operations in $\mathbb{F}_p$ if $\mathbb{K} = \mathbb{F}_{p^k}$,*

plus some univariate factorizations over $\mathbb{K}$ whose degree sum is r.

As in [2], we recover the usual complexity estimate $O(d_x d_y^\omega)$ when F is a dense polynomial. However, Theorem 1.2 may improve significantly [2] when F is non degenerated, as illustrated by the following example.

Example 1.3. Let F of bidegree $(2n, 2n)$, with Newton polygon

$$N(F) = \mathrm{Conv}\big((0,2),(2n,0),(0,2n),(2n,2n)\big).$$

The lower lattice length is $r = 2$, which is a very strong combinatorial constraint: there is a unique Minkowski sum decomposition whose summands have positive volume as shown by Figure 1.

As the bounding rectangle has size $O(V)$, the convex-dense approach of [2] boils down to the dense algorithm [13]. We get the following complexity estimates:

- Dense [13, 11] or convex-dense [2] algorithms: $O(n^{\omega+1})$ operations in $\mathbb{K}$ plus one univariate factorization of degree $2n$.
- Theorem 1.2 (assuming F non degenerated): $\tilde{O}(n^2)$ operations in $\mathbb{K}$ plus one univariate factorization of degree 2.

We get here a softly linear complexity. This is the most significant gain we can get, including the univariate factorization step.

A weakness of classical algorithms is to perform a shift $x \mapsto x + x_0$ to reduce to the case $F(0,y)$ separable, losing in such a way the combinatorial constraints offered by $N(F)$. Our approach avoids this shift.

1.2 Even faster

We can play with affine automorphisms $\tau \in \mathrm{Aut}(\mathbb{Z}^2)$ to minimize r while keeping V constant before applying Theorem 1.2. This leads to the concept of *minimal lattice length* of a lattice polygon P, defined as

$$r_0(P) := \min\left\{r(\tau(P)) \mid \tau \in \mathrm{Aut}(\mathbb{Z}^2)\right\}. \tag{1}$$

This integer is easy to compute (Lemma 3.13). Note that $r_0(N(F))$ can be reached by several τ, which can lead to various lower boundaries with lattice length r_0 (see Example 1.6 below). Let $\tau(F)$ be the image of F when applying τ to its monomial exponents.

Definition 1.4. *We say that F is* minimally non degenerated *if $\tau(F)$ is non degenerated for at least one transform τ reaching r_0.*

If F is minimally non degenerated, we may apply Theorem 1.2 to $\tau(F)$, with same volume V but with smaller r. The factorization of F is recovered for free from that of $\tau(F)$. We thus obtain:

Corollary 1.5. *Suppose that $F \in \mathbb{K}[x, y]$ is minimally non degenerated with minimal lattice length r_0. Then we can factorize F with*

1. *$\tilde{O}(r_0 V) + O(r_0^{\omega-1} V)$ operations in $\mathbb{K}$ if $p = 0$ or $p \geqslant 4V$, or*
2. *$O(k r_0^{\omega-1} V)$ operations in $\mathbb{F}_p$ if $\mathbb{K} = \mathbb{F}_{p^k}$,*

plus some univariate factorizations over $\mathbb{K}$ whose degree sum is r_0.

Notice that similar transforms $F \mapsto \tau(F)$ are used in [2], but the authors rather focus on minimizing the size of the bounding rectangle of $N(F)$, while we focus on minimizing r. The following examples illustrate the differences between these two approaches.

Example 1.6. Let $0 < m < n$ be two integers and suppose that

$$N(F) = \mathrm{Conv}\big((0,0), (m,0), (0,m), (n,n)\big),$$

as represented on the left side of Figure 2. The lower boundary $\Lambda(F)$ is the union of the yellow and red edges, with lattice length $r = m + \gcd(m, n)$. Applying the affine automorphism $\tau : (i, j) \mapsto (j, m - i + j)$, the resulting polygon $\tau(N(F))$ has red lower boundary, with minimal lattice length $r_0 = gcd(m, n)$. The bounding rectangle of $\tau(N(F))$ has volume $2mn = V/2$, so [2] would apply a dense algorithm on $\tau(F)$. We get the following estimates:

- Dense algorithm [11, 13]: $O(n^{\omega+1})$ operations and one univariate factorization of degree n.
- Convex-dense algorithm [2]: $O(nm^{\omega})$ operations and one univariate factorization of degree $2m$.

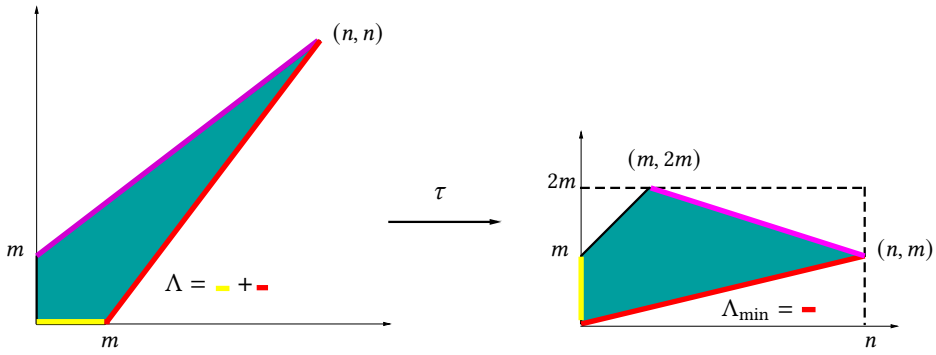

Figure 2: The affine automorphism $\tau : (i, j) \mapsto (j, m - i + j)$.

- Theorem 1.2 (assuming F non degenerate): $O(nm \gcd(n, m)^{\omega-1})$ operations and one univariate factorization of degree $\gcd(m, n)$.

Again, if $\gcd(m, n) \ll m$, our approach will be significantly faster than [2], including the univariate factorization step. Notice that by symmetry, r_0 is reached also by the transform τ' which maps the purple edge as the lower convex hull. Hence, even if F were "red-edge" degenerated, we would have a second chance that F is not "purple-edge" degenerated, allowing then to apply Corollary 1.5.

In the previous example, the image $\tau(F)$ reached simultaneously a minimal lower lattice length and a bounding rectangle of size $O(V)$. The next example illustrates that this is not always the case.

Example 1.7. Suppose that F has Newton polygon $N(F)$ as represented on the left side of Figure 3, depending on parameters k, n. The bounding rectangle of $N(F)$ has volume $O(kn^2) = O(V)$, so [2] applies a dense algorithm on F. Any black edge has lattice length n or $n + 2$ while the red edge has lattice length $r = 2$. We check that the affine automorphism $\tau(i, j) = (2i + j - 2n, -i + kn)$ sends $N(F)$ to the right hand polygon, leading to $r_0 = 2$. We get the complexity estimates:

- Dense [11, 13] or convex-dense algorithms [2]: $O(kn^{\omega+1})$ and one univariate factorization of degree $4n + 4$.
- Theorem 1.2 (assuming F minimally non degenerated): $\tilde{O}(kn^2)$ operations and one univariate factorization of degree 2.

Again, we get a softly linear complexity. This example illustrates the fact that minimizing the lower lattice length may increase significantly the volume of the bounding rectangle ($k^2n^2 \gg V$).

Classical fast factorization algorithms are based on a "lifting and recombination" scheme: factorize F in $\mathbb{K}[[x]][y]$ with x-adic precision $O(d_x)$ and recombine the analytic factors into global factors. Example 1.7 shows that we can't apply this strategy to our

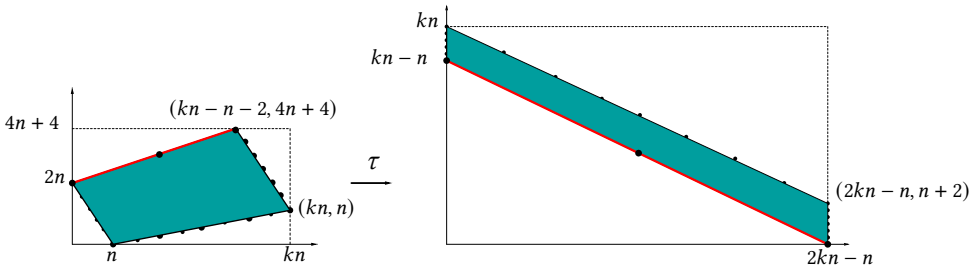

Figure 3: The affine automorphism $\tau : (i, j) \mapsto (2i + j - 2n, -i + kn)$.

target polynomial $\tau(F)$: the analytic factorization with precision $d_x = kn$ would have size $O(k^2n^2)$ which does not fit in our aimed bound. To remediate this, we will rather factorize $\tau(F)$ in $\mathbb{K}[[x]][y]$ with respect to another suitable valuation depending on the polygon. This is the second main result of our paper, that we explain now.

1.3 Fast valuated analytic factorization

Let $\lambda \in \mathbb{Q}$ and let v_λ stands for the valuation

$$v_\lambda : \mathbb{K}((x))[y] \to \mathbb{Q}, \qquad v_\lambda(\sum c_{ij}x^j y^i) := \min(j + i\lambda, c_{ij} \neq 0), \tag{2}$$

with convention $v_\lambda(0) = \infty$. If $F \in \mathbb{K}((x))[y]$, the lower convex hull $\Lambda = \Lambda(F)$ is well defined, and Definition 1.1 still makes sense in this larger ring. We denote

$$m_\lambda(F) = \max_{(i,j)\in\Lambda} (j + i\lambda) - v_\lambda(F). \tag{3}$$

Note that $m_\lambda(F) \geqslant 0$, with equality if and only if $\Lambda(F)$ is straight of slope $-\lambda$. We measure the quality of the v_λ-approximation of F by a polynomial G by the relative quantity $v_\lambda(F - G) - v_\lambda(F)$. We prove:

Theorem 1.8. *Let $F \in \mathbb{K}((x))[y]$ monic of degree d. Suppose that F is non degenerate, with monic irreducible factors $F_1^*, \ldots, F_s^*$. Given $\sigma \geqslant m_\lambda(F)$, we can compute $F_1, \ldots, F_s$ monic such that*

$$v_\lambda(F - F_1 \cdots F_s) - v_\lambda(F) > \sigma$$

with $\tilde{O}(d\sigma)$ operations in $\mathbb{K}$ plus some univariate factorizations over $\mathbb{K}$ whose degree sum is at most d. Moreover, each factor is approximated with a relative precision

$$v_\lambda(F_i - F_i^*) - v_\lambda(F_i^*) > \sigma - m_\lambda(F)$$

for all $i = 1, \ldots, s$.

To our knowledge, this result is new. It improves [17] and [18], which focus on the Gauss valuation v_0 and reach a quasi-optimal complexity only for $\sigma \geqslant dm_0(F)$ and characteristic of $\mathbb{K}$ zero or high enough. It turns out that we need to get rid of all these restrictions for our purpose. The proof of Theorem 1.8 is based on two main points:

- Fast arithmetic of sparse polynomials, leading to a softly linear v_λ-adic Hensel lifting (Proposition 2.18).
- A divide and conquer strategy based on a suitable choice of the various slopes λ' which will be used at each recursive call of Hensel lifting.

1.4 Main lines of the proof of Theorem 1.2

Except the choice of the valuation, the strategy for the proof of Theorem 1.2 mainly follows [13, 24]:

- We choose a suitable $\lambda \in \mathbb{Q}$ and we compute the factorization of F in $\mathbb{K}[[x]][y]$ with v_λ-adic precision $\sigma \in O(V/d_y)$, for a cost $\tilde{O}(V)$ by Theorem 1.8.
- Adapt the logarithmic derivative method of [13, 24] to reduce to linear algebra the problem of recombinations of the truncated analytic factors into factors in $\mathbb{K}[x, y]$. *A good choice of λ is a key point* to ensure that the v_λ-adic precision $O(V/d_y)$ is sufficient to solve recombinations.
- We are reduced to solve a linear system of at most r unknowns and $O(V)$ equations, which fits in the aimed bound. We build the underlying recombination matrix using a fast v_λ-adic Euclidean division by non monic polynomials (Proposition 3.6).

1.5 The case of degenerated polynomials

Note first that the non degeneracy hypothesis holds generically among all polynomials with prescribed polytope. If however F is degenerated, let us mention three options:

- We may compute nevertheless the v_λ-adic factorization of $F \in \mathbb{K}[[x]][y]$. We still expect a softly linear complexity using the recent algorithms [17, 18] combined with Theorem 1.8. The main drawback is that we might need a higher precision for solving recombinations, and this approach will be worthwhile only if the so-called "separability exponent" of F is not too big, see [24] for details in the x-adic case.
- We can also look for an other map τ such that $\tau(F)$ is not degenerated and has a lower lattice length which is "small enough," although not minimal.
- If none of these two options is worthwhile, then we can always use the convex-dense algorithm of [2]. Indeed, considering only the "bounding rectangle" allows to use a shift $x \mapsto x + x_0$ to reach the non degenerated case.

Remark 1.9. Let us mention too [5], where the authors develop a Hensel lifting with respect to a *Newton precision*, given by a convex piecewise affine function. It might be interesting to look if such an approach could be useful for our purpose, as it allows to take care of the shape of $\Lambda(F)$.

1.6 Organisation of the paper

Section 2 is dedicated to the proof of Theorem 1.8. In Section 3, we adapt the lifting and recombination scheme of [13, 11] in the v_λ-adic context, leading to the proof of Theorem 1.2 and Corollary 1.5.

2 Fast v_λ-adic factorization

In what follows, we fix $\lambda = m/q \in \mathbb{Q}$ with $q \geqslant 1$ and q, m coprime and we consider the valuation v_λ as defined in (2).

2.1 The ring $\mathbb{A}_\lambda$ and its fast arithmetic

Consider the classical Newton–Puiseux transformation

$$\tau_\lambda : \mathbb{K}((x))[y] \to \mathbb{K}((x))[y], \qquad F(x,y) \mapsto \widehat{F}(x,y) = F(x^q, x^m y). \tag{4}$$

This map is an injective $\mathbb{K}$-algebra endomorphism. Thus, its image

$$\mathbb{A}_\lambda := \mathbb{K}((x^q))[x^m y] \subset \mathbb{K}((x))[y]$$

is a subring isomorphic to $\mathbb{K}((x))[y]$. We denote

$$\mathbb{A}_\lambda^+ = \mathbb{A}_\lambda \cap \mathbb{K}[[x]][y] \qquad \text{and} \qquad \mathbb{B}_\lambda = \mathbb{A}_\lambda \cap \mathbb{K}[x,y].$$

Both sets are subrings of $\mathbb{A}_\lambda$. Note that the map τ_λ preserves the size of the support of a polynomial.

The valuation v_λ is related to the Gauss valuation v_0 by

$$v_0(\tau_\lambda(F)) = q v_\lambda(F). \tag{5}$$

Unfortunately, computing the v_0-adic factorization of $\tau_\lambda(F)$ which induces the v_λ-adic factorization of F with the recent softly linear algorithms [16] does not fit in the aimed bound due to the presence of the extra factor q in (5). To remediate this problem, we need take advantage of the sparsity of $\tau_\lambda(F)$, which is reflected in more details by the following lemma.

Lemma 2.1. *Let $F \in \mathbb{K}((x))[y]$. Then $F \in \mathbb{A}_\lambda$ if and only if*

$$F = \sum_k f_k(y^q) y^{\alpha_\lambda(k)} x^k, \quad f_k \in \mathbb{K}[y]$$

where $0 \leqslant \alpha_\lambda(k) < q$ is defined by $\alpha_\lambda(k) \equiv k m^{-1} \mod q$. Equivalently, we have

$$\mathbb{A}_\lambda = \bigoplus_{k=0}^{q-1} x^k y^{\alpha_\lambda(k)} \mathbb{K}((x^q))[y^q].$$

In particular, $\mathbb{A}_\lambda \cap \mathbb{K}((x)) = \mathbb{K}((x^q))$ and $\mathbb{A}_\lambda \cap \mathbb{K}[y] = \mathbb{K}[y^q]$.

Proof. By (4), we have $F \in \mathbb{A}_\lambda$ if and only if $F = \sum_{i,j} c_{ij} x^{mi+qj} y^i$ for some $c_{ij} \in \mathbb{K}$. For a fixed k, there exists i, j such that $mi + qj = k$ if and only $i \equiv km^{-1}[q]$. The proof follows straightforwardly. □

Corollary 2.2. $\mathbb{K}((x))[y] = \mathbb{A}_\lambda \oplus y\mathbb{A}_\lambda \oplus \cdots \oplus y^{q-1}\mathbb{A}_\lambda = \mathbb{A}_\lambda \oplus x\mathbb{A}_\lambda \oplus \cdots \oplus x^{q-1}\mathbb{A}_\lambda$.

Notice that if $q > 1$, neither x nor y belongs to $\mathbb{A}_\lambda$. Let us consider the union of all translated of $\mathbb{A}_\lambda$ and $\mathbb{B}_\lambda$ by a monomial:

$$\tilde{\mathbb{A}}_\lambda = \bigcup_{i\in\mathbb{Z}, j\in\mathbb{N}} x^i y^j \mathbb{A}_\lambda, \qquad \tilde{\mathbb{B}}_\lambda = \bigcup_{i\in\mathbb{Z}, j\in\mathbb{N}} x^i y^j \mathbb{B}_\lambda.$$

These sets are not stable by addition, but they both form a multiplicative monoid.

Corollary 2.3. *If $F \in \tilde{\mathbb{B}}_\lambda$ has bidegree (d, n), its support has size $O(dn/q)$.*

In what follows, we simply say precision for Gauss precision.

2.1.1 Fast multiplication in $\tilde{\mathbb{A}}_\lambda$

A key point for our purpose is that we have access to a faster multiplication in $\tilde{\mathbb{A}}_\lambda$ than in $\mathbb{K}((x))[y]$. Let us start with an easy lemma.

Lemma 2.4. *Let $G, H \in \mathbb{K}((x))[y]$ and let $N \in \mathbb{Z}$. The product $GH \mod x^N$ only depends on $G \mod x^{N-v_0(H)}$ and $H \mod x^{N-v_0(G)}$.*

Proof. Clear. □

Proposition 2.5. *Let $G, H \in \tilde{\mathbb{A}}_\lambda$ of degree at most d. Given $n > 0$, we can compute $F = GH$ with precision $n + v_0(F)$ with $\tilde{O}(dn/q)$ operations in $\mathbb{K}$.*

Proof. Thanks to the relation $v_0(F) = v_0(G) + v_0(H)$, Lemma 2.4 shows that it's enough to compute $F_0 = G_0 H_0$ with $G_0, H_0 \in \tilde{\mathbb{B}}_\lambda$ defined as

$$G_0 = G \mod x^{n+v_0(G)}, \quad H_0 = H \mod x^{n+v_0(H)}.$$

The supports of G_0, H_0 have size $O(dn/q)$. Since $\tilde{\mathbb{B}}_\lambda$ is a monoid, the support of $F_0 = G_0 H_0 \in \tilde{\mathbb{B}}_\lambda$ has also size $O(dn/q)$. It follows from [21, Proposition 6] or [20, Theorem 12] that F_0 can be computed in time $\tilde{O}(dn/q)$. □

We thus gain a factor q when compared to usual bivariate multiplication. Note that fast multiplication of polynomials with prescribed support is based on a sparse multivariate evaluation-interpolation strategy (see [20, 21] and references therein), the crucial point here being that $F = GH$ remains sparse thanks to the monoid structure of $\tilde{\mathbb{A}}_\lambda$.

2.1.2 Fast division in $\mathbb{A}_\lambda$

Since the map τ_λ preserves the degree in y, both rings $\mathbb{A}_\lambda, \mathbb{A}_\lambda^+$ are Euclidean rings when considering division with respect to y. Namely, given $F, G \in \mathbb{K}((x))[y]$, the Euclidean division $F = QG + R$, $\deg(R) < \deg(G)$ forces the Euclidean division of $\widehat{F}, \widehat{G}$ defined by (4) to be

$$\widehat{F} = \widehat{Q}\widehat{G} + \widehat{R}, \quad \widehat{Q}, \widehat{R} \in \mathbb{A}_\lambda, \quad \deg(\widehat{R}) < \deg(\widehat{G}).$$

Moreover, the next lemma ensures that if $\widehat{F}, \widehat{G} \in \mathbb{A}_\lambda^+$ (resp. $\mathbb{B}_\lambda$) with $\widehat{G}$ monic, then $\widehat{Q}, \widehat{R} \in \mathbb{A}_\lambda^+$ (resp. $\mathbb{B}_\lambda$).

Lemma 2.6. *Let $F, G \in \mathbb{K}((x))[y]$ with Euclidean division $F = QG + R$. Assume that the leading coefficient of G has valuation $v_0(G)$. Then*

$$v_0(Q) \geqslant v_0(F) - v_0(G) \quad \textit{and} \quad v_0(R) \geqslant v_0(F).$$

Proof. See e.g. [18] (a similar result holds for an arbitrary valuation). □

Given $F \in \mathbb{K}((x))[y]$ of degree d, let us denote by $\tilde{F} = y^d F(x, y^{-1})$ its reciprocal polynomial. We will need the following lemma.

Lemma 2.7. *Let $F \in \mathbb{A}_\lambda$ of degree d. Then $\tilde{F} \in y^r \mathbb{A}_{-\lambda}$ where $r = d \mod q$.*

Proof. Let $F \in \mathbb{A}_\lambda$ with expression as in Lemma 2.1. Then

$$\tilde{F} = \sum_k \tilde{f}_k(y^q) y^{d - q\deg(f_k) - \alpha_\lambda(k)} x^k, \quad f_k \in \mathbb{K}[y].$$

We have $d - q\deg(f_k) - \alpha_\lambda(k) \equiv r + \alpha_{-\lambda}(k) \mod q$ and the claim follows from Lemma 2.1 applied in the ring $\mathbb{A}_{-\lambda}$. □

Proposition 2.8. *Let $F, G \in \mathbb{A}_\lambda$ of degree at most d, and suppose that the leading coefficient of G has valuation $v_0(G)$. Given $n \geqslant 0$, we can compute $Q, R \in \mathbb{A}_\lambda$ with $\deg(Q) < \deg(G)$ such that*

$$F = QG + R \mod x^{v_0(F)+n}$$

with $\tilde{O}(dn/q)$ operations in $\mathbb{K}$.

Proof. Let $e = \deg(G)$ and $d = \deg(F)$. Assume $d > e$. Let us first reduce to the case where G is monic. We need to take care that multiplication by an arbitrary power of x is not allowed in $\mathbb{A}_\lambda$. We proceed as follows. Let $k = -v_0(G)$ and let $\alpha = \alpha_\lambda(k)$. By Lemma 2.1, we have $x^k y^\alpha \in \mathbb{A}_\lambda$ so the polynomials

$$G_0 = x^k y^\alpha G \quad \text{and} \quad F_0 = x^k y^\alpha F$$

belong to $\mathbb{A}_\lambda$, with now $v_0(G_0) = 0$. We are reduced to solve

$$F_0 = QG_0 + R_0 \mod x^{v_0(F_0)+n}$$

in $\mathbb{A}_\lambda$, recovering R for free from the relation $R_0 = x^k y^\alpha R$. By assumption the leading coefficient $u(x)$ of G_0 is invertible in $\mathbb{K}[[x]]$. Moreover, $\deg(G_0) = e + \alpha$ is divisible by q and it follows from Lemma 2.1 that $u \in \mathbb{K}[[x^q]] \subset \mathbb{A}_\lambda$. Hence so does u^{-1}. Thus u can be invert in $\mathbb{A}_\lambda^+$ with precision n in time $\tilde{O}(n/q)$, and we may suppose safely that G_0 is monic. Note that

$$\deg(F_0) - \deg(G_0) = \deg(F) - \deg(G) = d - e.$$

The classical fast Euclidean division $F_0 = QG_0 + R_0$ runs as follows:

1. Truncate F_0 at precision $n + v_0(F_0)$ and G_0 at precision n.
2. Compute $\tilde{H}_0 := \tilde{G}_0^{-1} \mod y^{d-e+1}$ with precision n.
3. Compute $\tilde{Q} = \tilde{F}_0 \tilde{H}_0 \mod y^{d-e+1}$ with precision $n + v_0(F)$.
4. Compute $Q = y^{d-e}\tilde{Q}(x, y^{-1})$.
5. Compute $R_0 = F_0 - QG_0$ with precision $n + v_0(F)$.

Note that Step 2 makes sense: since G_0 is monic, we have $\tilde{G}_0(0) = 1$ so $\tilde{G}_0$ can be invert in $\mathbb{K}[[x]][[y]]$. This algorithm returns the correct output $F = QG + R$ if we do not truncate, see e.g. [10, Theorem 9.6] and Lemma 2.6 and Lemma 2.4 ensure that truncations are correct to get $F_0 = QG_0 + R_0 \mod x^{n+v_0(F)}$. Using quadratic Newton iteration, the inversion of $\tilde{G}_0$ at Step 2 requires $O(\log(d))$ multiplications and additions in $\mathbb{K}[[x]][y]$ of degrees at most $d-e$ with precision n (see e.g. [10, Theorem 9.4]). Since q divides $\deg(G_0)$, Lemma 2.7 gives $\tilde{G}_0 \in \mathbb{A}_{-\lambda}$, which is a ring. Hence all additions and multiplications required by [10, Algorithm 9.3] take place in $\mathbb{A}_{-\lambda}$ and the cost of Step 2 fits in the aimed bound thanks to Proposition 2.5. Since $\tilde{H}_0, \tilde{F}_0 \in \tilde{\mathbb{A}}_{-\lambda}$ by Lemma 2.7, we compute $\tilde{Q}$ at Step 3 in time $\tilde{O}((d-e)n/q)$ by Proposition 2.5. Step 4 is free. At Step 5, the equation has degree $d + \alpha$ and vanish $\mod y^\alpha$, so its sparse size is $\tilde{O}(dn/q)$ and Step 5 fits too in the aimed bound since $F_0, Q, G_0 \in \mathbb{A}_\lambda$. □

2.1.3 Fast Hensel lifting in $\mathbb{A}_\lambda$

Proposition 2.9. *Let $F \in \mathbb{A}_\lambda^+$ of degree d and consider a coprime factorization $F(0, y) = f_0 \cdots f_r f_\infty$ in $\mathbb{A}_\lambda \cap \mathbb{K}[y] = \mathbb{K}[y^q]$ with f_i monic and $f_\infty = c \in \mathbb{K}^\times$. Then there exists uniquely determined polynomials $F_0, \ldots, F_r, F_\infty \in \mathbb{A}_\lambda^+$ such that*

$$F = F_0 \cdots F_r F_\infty, \quad F_i(0, y) = f_i(0, y) \quad i = 0, \ldots, k, \infty$$

with F_i monic of degree $\deg(f_i)$. We can compute the F_i's with precision n within $\tilde{O}(dn/q)$ operations in $\mathbb{K}$. Moreover, the truncated polynomials $F_i \mod x^n$ are uniquely determined by the equality $F \equiv F_0 \cdots F_r F_\infty \mod x^n$.

Proof. This is the classical fast multi-factor Hensel lifting, see e.g. [10, Algorithm 15.17]. The algorithm is based on multiplications and divisions of polynomials at precision n. The initial Bezout relations holds here in $\mathbb{K}[y^q] \subset \mathbb{A}_\lambda$, and it follows that at each Hensel

step, the input polynomials belong to the ring $\mathbb{A}_\lambda$. Moreover, all Euclidean divisions satisfy the hypothesis of Proposition 2.8. The claim thus follows from Proposition 2.5 and Proposition 2.8 together with [10, Theorem 15.18]. Unicity of the lifting mod x^n follows from [10, Theorem 15.14]. □

Remark 2.10. It is crucial to consider the factorization of $F(0, y)$ in the ring $\mathbb{A}_\lambda$. Typically, a polynomial of shape $y^q - 1$ should be considered irreducible. Otherwise, the complexity will be $\tilde{O}(dn)$ due to the loss of sparse arithmetic.

Remark 2.11. Propositions 2.8 and 2.9 appear also in [15, Propositions 11 and 12] under the assumption that $F \in \mathbb{B}_\lambda$ is monic. However, the proofs have not been published up to our knowledge.

2.2 Fast v_λ-adic Hensel lemma

By the isomorphism $\tau_\lambda : \mathbb{K}((x))[y] \to \mathbb{A}_\lambda$, the previous results translate in an obvious way in quasi-linear complexity estimates for v_λ-adic truncated multiplication and division in $\mathbb{K}((x))[y]$.

Corollary 2.12. *Let $\lambda \in \mathbb{Q}$ and let $G, H \in \mathbb{K}((x))[y]$ of degree at most d. We can compute $F = GH$ at λ-precision $v_\lambda(F) + \sigma$ with $\tilde{O}(d\sigma)$ operations in $\mathbb{K}$.*

Proof. Follows from (5) together with Proposition 2.5. □

Corollary 2.13. *We can multiply arbitrary polynomials $G, H \in \mathbb{K}[x, y]$ in quasi-linear time with respect to the λ-size of the output.*

Remark 2.14. We could have used directly a sparse multivariate evaluation-interpolation strategy on the input polynomials G, H. However, we believe that using fast arithmetic in the ring $\mathbb{A}_\lambda$ is more convenient and offers more applications.

Definition 2.15. *We say that $G \in \mathbb{K}((x))[y]$ is λ-monic if its leading monomial uy^e satisfies $v_\lambda(uy^e) = v_\lambda(G)$.*

Proposition 2.16. *Let $F, G \in \mathbb{K}((x))[y]$ of degrees at most d with G λ-monic. We can compute $Q, R \in \mathbb{K}((x))[y]$ such that*

$$v_\lambda(F - (QG + R)) \geqslant v_\lambda(F) + \sigma$$

within $\tilde{O}(d\sigma)$ operations.

Proof. We apply Proposition 2.8 to the polynomials $\widehat{F} = \tau_\lambda(F)$ and $\widehat{G} = \tau_\lambda(G)$. We are reduced to compute $\widehat{Q}, \widehat{R}$ such that

$$v_0\big(\widehat{F} - (\widehat{Q}\widehat{G} + \widehat{R})\big) \geqslant v_0(\widehat{F}) + q\sigma. \tag{6}$$

Since G is assumed to be λ-monic, the leading coefficient $u(x)$ of $\widehat{G}$ has valuation $v_0(\widehat{G})$ and we conclude thanks to Proposition 2.8. □

We get finally a fast Hensel lifting with respect to the valuation v_λ.

Definition 2.17. *Let* $F = \sum c_{ij}x^iy^j \in \mathbb{K}((x))[y]$ *and* $\sigma \in \frac{1}{q}\mathbb{Z}$. *The* λ-homogeneous component *of* F *of degree* σ *is*

$$F_\sigma = \sum_{i+j\lambda=\sigma} c_{ij}x^iy^j \in \mathbb{K}[x^{\pm 1}][y].$$

The λ-initial part *of* F *is the* λ*-homogeneous component of* F *of lowest degree* $v_\lambda(F)$*, denoted by* $\mathrm{in}_\lambda(F)$.

Let $F \in \mathbb{K}((x))[y]$. The irreducible factorization of the λ-initial part of F can be written in a unique way (up to permutation) as

$$\mathrm{in}_\lambda(F) = p_0p_1\cdots p_kp_\infty \in \mathbb{K}[x^{\pm 1}][y] \tag{7}$$

where $p_0 = y^n$ with $n \in \mathbb{N}$, $p_\infty = ux^a$ with $a \in \mathbb{Z}$, $u \in \mathbb{K}^\times$ and where $p_1, \ldots, p_k$ are coprime powers of irreducible λ-homogeneous monic polynomials, not divisible by y. The following result is well known (see e.g. [4, Chapter VI]).

Proposition 2.18. *There exists unique polynomials* $P_i^* \in \mathbb{K}((x))[y]$ *such that*

$$F = P_0^*\cdots P_k^*P_\infty^* \in \mathbb{K}((x))[y], \qquad \mathrm{in}_\lambda(P_i^*) = p_i,$$

with P_i^* λ*-monic of* $\deg(P_i^*) = \deg(p_i)$ *for* $i = 0, \ldots, k$. *Moreover,* P_i^* *is irreducible if* p_i *is irreducible.*

We get the following complexity result.

Proposition 2.19. *Given* $\sigma \in \mathbb{Q}^+$*, and given the irreducible factorization* (7) *we can compute* $P_0, \ldots, P_\infty \in \mathbb{K}[x^{\pm 1}][y]$ *such that*

$$v_\lambda(P_i^* - P_i) > v_\lambda(P_i^*) + \sigma \quad \forall i = 0, \ldots, k, \infty$$

in time $\tilde{O}(d\sigma)$. *We have then*

$$v_\lambda(F - P_0\cdots P_\infty) > v_\lambda(F) + \sigma.$$

Proof. Even if it means multiplying F by a suitable monomial x^iy^α with $0 \leqslant \alpha < q$, we may assume that $v_\lambda(F) = 0$. Then we apply Proposition 2.9 to the polynomial $\widehat{F} = \tau_\lambda(F)$, starting from the factorization of $\widehat{F}(0, y)$ induced by the factorization of $\mathrm{in}_\lambda(F)$, and with a suitable Gauss precision in order to recover the desired λ-precision. The cost and the unicity of the truncated polynomial P_i follow from Proposition 2.9. □

Remark 2.20. This result improves [18, Corollary 2] which gives the complexity estimate $\tilde{O}(d(\sigma + v_\lambda(F))$. Proposition 2.19 has a significant impact when a lifting precision closed to $v_\lambda(F)$ is needed. This is precisely the case for our application to bivariate factorization.

Definition 2.21. *We denote* `PartialFacto`(F, λ, σ) *the algorithm which computes the factorization* (7) *of* $\mathrm{in}_\lambda(F)$ *and returns the truncated factors* $P_0, \ldots, P_\infty$ *following Proposition 2.19.*

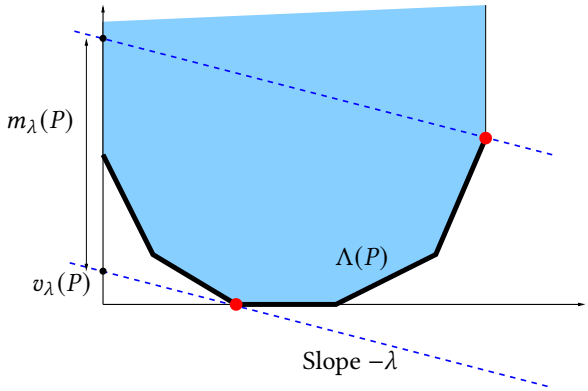

Figure 4: Example of λ-defect of straightness where $m_\lambda(P) = b_\lambda(P) > a_\lambda(P)$.

2.3 Fast v_λ-adic factorization

We want now to compute the complete irreducible factorization of F in $\mathbb{K}((x))[y]$. Although our target precision is measured in terms of the valuation v_λ, we will perform recursive calls of `PartialFacto` with various valuations $v_{\lambda'}$. The integer $m_\lambda(F)$ introduced in (3) will play a key role.

2.3.1 The λ-defect of straightness

Definition 2.22. *Given $P = \sum_{i=s}^{n} p_i y^i \in \mathbb{K}((x))[y]$ with $p_s, p_n \neq 0$, we denote $\mathrm{in}_y(P) = p_s y^s$ the initial term of P and $\mathrm{lt}_y(P) = p_n y^n$ the leading term of P. We define*

$$a_\lambda(P) = v_\lambda(\mathrm{in}_y(P)) - v_\lambda(P) \quad \text{and} \quad b_\lambda(P) = v_\lambda(\mathrm{lt}_y(P)) - v_\lambda(P).$$

The λ-defect of straightness *of P is $m_\lambda(P) = \max(a_\lambda(P), b_\lambda(P))$.*

See Figure 4 for an example in the case that $m_\lambda(P) = b_\lambda(P) > a_\lambda(P)$.

Recall from the introduction that $\Lambda(P)$ is the lower convex hull of the set of points $(i, v_0(p_i))$, $i = s, \ldots, n$, where $v_0(p_i)$ is the x-adic valuation. By convexity, $v_0(p_i) + i\lambda$ takes its maximal value at $i = s$ or $i = n$, hence the definition of $m_\lambda(P)$ coincides with (3). The terminology for m_λ is justified by the following fact.

Lemma 2.23. *The following properties hold:*

1. $a_\lambda(P) \geqslant 0$.
2. $b_\lambda(P) \geqslant 0$ *with equality if and only if P is λ-monic.*
3. $m_\lambda(P) \geqslant 0$ *with equality if and only if $\Lambda(P)$ is one-sided of slope $-\lambda$.*

Proof. This follows from the equality $v_\lambda(P) = \min\{v_\lambda(p_i y^i) \mid i = s, \ldots, n\}$. □

Corollary 2.24. *Let $P, Q \in \mathbb{K}((x))[y]$. We have $\Lambda(PQ) = \Lambda(P) + \Lambda(Q)$ and*

$$m_\lambda(PQ) \geqslant \max(m_\lambda(P), m_\lambda(Q))$$

with equality if $\Lambda(Q)$ or $\Lambda(P)$ is one-sided of slope $-\lambda$.

Proof. First equality is a well known variant of Ostrowski's theorem. Since in_y and lt_y are multiplicative operators and v_λ is a valuation, we get

$$a_\lambda(PQ) = a_\lambda(P) + a_\lambda(Q) \quad \text{and} \quad b_\lambda(PQ) = b_\lambda(P) + b_\lambda(Q).$$

The inequality for m_λ follows straightforwardly. If $\Lambda(Q)$ or $\Lambda(P)$ is one-sided of slope $-\lambda$, the equality follows from point (3) of Lemma 2.23. □

2.3.2 Comparisons between various valuations

Lemma 2.25. *Let $\lambda' \geqslant \lambda$ and let $P \in \mathbb{K}((x))[y]$ of degree n. Then*

$$v_\lambda(P) \leqslant v_{\lambda'}(P) \leqslant v_\lambda(P) + n(\lambda' - \lambda)$$

Proof. Since $i + j\lambda \leqslant i + j\lambda'$ we get immediately $v_\lambda \leqslant v_{\lambda'}$. Let (i_0, j_0) in the support of P such that $v_\lambda(P) = i_0 + j_0\lambda$. We get

$$v_{\lambda'}(P) \leqslant i_0 + j_0\lambda' = i_0 + j_0\lambda + j_0(\lambda' - \lambda) = v_\lambda(P) + j_0(\lambda' - \lambda)$$

and we conclude thanks to $j_0 \leqslant n$. □

Definition 2.26. *Let $P_0, P \in \mathbb{K}((x))[y]$. We say that P_0 approximates P with relative λ-precision σ if $v_\lambda(P - P_0) > v_\lambda(P) + \sigma$. We say that P is known with relative λ-precision σ if we know such an approximant P_0.*

Corollary 2.27. *Let $P \in \mathbb{K}((x))[y]$ of degree n, let $\lambda, \lambda' \in \mathbb{Q}$ and let $\sigma \geqslant 0$. If P is known with relative λ'-precision*

$$\sigma' = \sigma'(\lambda, \lambda', \sigma, P) := \begin{cases} \sigma + v_\lambda(P) - v_{\lambda'}(P) + n(\lambda' - \lambda) & \text{if } \lambda' \geqslant \lambda \\ \sigma + v_\lambda(P) - v_{\lambda'}(P) & \text{if } \lambda' \leqslant \lambda \end{cases} \tag{8}$$

then P is known with relative λ-precision σ.

Proof. This follows from the second inequality in Lemma 2.25 for the case $\lambda' \geqslant \lambda$ and from the first inequality in Lemma 2.25 for the case $\lambda' \leqslant \lambda$. □

Lemma 2.28. *We keep notations of Corollary 2.27.*

- *If $\lambda' \geqslant \lambda$, then $m_\lambda(P) + v_\lambda(P) - n\lambda \geqslant m_{\lambda'}(P) + v_{\lambda'}(P) - n\lambda'$.*
- *If $\lambda' \leqslant \lambda$, then $m_\lambda(P) + v_\lambda(P) \geqslant m_{\lambda'}(P) + v_{\lambda'}(P)$.*

Proof. Denoting $P = \sum_{i=s}^{n} p_i y^i$, the first inequality is equivalent to

$$\max\big(v_0(p_s) - (n-s)\lambda, v_0(p_n)\big) \geqslant \max\big(v_0(p_s) - (n-s)\lambda', v_0(p_n)\big),$$

which follows from the assumption $\lambda \leqslant \lambda'$. The second inequality is equivalent to

$$\max\big(v_0(p_s) + s\lambda, v_0(p_n) + n\lambda\big) \geqslant \max\big(v_0(p_s) + s\lambda', v_0(p_n) + n\lambda'\big),$$

which follows from the assumption $\lambda \geqslant \lambda'$. □

Corollary 2.29. *We have $\sigma' - \sigma \geqslant m_{\lambda'}(P) - m_\lambda(P)$.*

Proof. Combining (8) and Lemma 2.28 leads to the desired inequality. □

We will need also an upper bound for σ' in terms of σ.

Lemma 2.30. *We keep notations of Corollary 2.27. Suppose that $b_\lambda(P) = 0$ if $\lambda' \geqslant \lambda$ and that $a_\lambda(P) = 0$ and P not divisible by y if $\lambda' \leqslant \lambda$. Then $\sigma' \leqslant \sigma + m_{\lambda'}(P)$.*

Proof. Suppose $\lambda' \geqslant \lambda$. By (8), the inequality is equivalent to

$$m_{\lambda'}(P) \geqslant v_\lambda(P) - v_{\lambda'}(P) + n(\lambda' - \lambda).$$

Both sides are invariant when dividing P by some element of $\mathbb{K}((x))$, hence we can safely suppose P monic in y. The hypothesis $b_\lambda(P) = 0$ is still true and we get $v_\lambda(P) = v_\lambda(y^n) = n\lambda$. We are reduced to show that $m_{\lambda'}(P) \geqslant n\lambda' - v_{\lambda'}(P) = b_{\lambda'}(P)$, which follows from Definition 2.22. Suppose now $\lambda' \leqslant \lambda$. By (8), we need to show that $m_{\lambda'}(P) \geqslant v_\lambda(P) - v_{\lambda'}(P)$. By hypothesis, we have $v_\lambda(P) = v_\lambda(p_0) = v_{\lambda'}(p_0)$. We are reduced to show that $m_{\lambda'}(P) \geqslant v_{\lambda'}(p_0) - v_{\lambda'}(P) = a_{\lambda'}(P)$, which follows from Definition 2.22. □

2.3.3 Recursive calls

Let $F \in \mathbb{K}((x))[y]$. We fix $\lambda \in \mathbb{Q}$ and a relative λ-precision $\sigma \geqslant 0$. Following Definition 2.21, let us consider

$$L = [P_0, P_1, \ldots, P_k, P_\infty] = \texttt{PartialFacto}(F, \lambda, \sigma).$$

Assuming F non degenerated, we know thanks to Proposition 2.18 that $P_0, \ldots, P_\infty$ approximate some coprime factors $P_0^*, P_1^*, \ldots, P_k^*, P_\infty^*$ of F with relative λ-precision σ. Moreover, the polynomials $P_1^*, \ldots, P_k^*$ and their approximant are irreducible. There remains to factorize (if required) the polynomials P_0^* and P_∞^*. We denote for short

$$(G^*, H^*) = (P_0^*, P_\infty^*) \quad \text{and} \quad (G, H) = (P_0, P_\infty).$$

Lemma 2.31. *The polynomials G and G^* are monic of same degree and H and H^* are not divisible by y. Moreover:*

- *If P divides G, then $m_\lambda(P) = a_\lambda(P)$ and $b_\lambda(P) = 0$.*

- *If P divides H, then $m_\lambda(P) = b_\lambda(P)$ and $a_\lambda(P) = 0$.*

Proof. The first claim follows from Proposition 2.19, Proposition 2.18 and (7). More precisely, denoting $G = c_0 + \cdots + c_n y^n$ and $H = h_0 + \cdots + h_m y^m$ with $c_n, h_m \neq 0$, we have

$$\mathrm{in}_\lambda(G) = \mathrm{in}_\lambda(G^*) = \mathrm{in}_\lambda(y^n) \quad \text{and} \quad \mathrm{in}_\lambda(H) = \mathrm{in}_\lambda(H^*) = \mathrm{in}_\lambda(h_0).$$

As $v_\lambda(G - \mathrm{in}_\lambda(G)) > v_\lambda(G)$ we deduce $b_\lambda(G) = 0$ and $m_\lambda(G) = a_\lambda(G)$. In the same way, we get $a_\lambda(H) = 0$ and $m_\lambda(H) = b_\lambda(H)$. If P divides G, we have $b_\lambda(P) \leqslant b_\lambda(G)$ by multiplicativity of b_λ. As $b_\lambda(P) \geqslant 0$, this forces $b_\lambda(P) = 0$, and thus $m_\lambda(P) = a_\lambda(P)$. If P divides H, then $0 \leqslant a_\lambda(P) \leqslant a_\lambda(H)$ forces now $a_\lambda(P) = 0$ and $m_\lambda(P) = b_\lambda(P)$. □

We need a lower bound on σ which ensures that we can detect the irreducible factors of G^* and H^* on their approximants G and H.

Lemma 2.32. *Suppose that $\sigma \geqslant m_\lambda(F)$. Then $\Lambda(G) = \Lambda(G^*)$ and the restriction of G and G^* to their lower convex hull coincide. The same assertion is true for H and H^*. In particular, $m_\lambda(G) = m_\lambda(G^*)$ and $m_\lambda(H) = m_\lambda(H^*)$.*

Proof. By Lemma 2.31, we have $G^* = c_s^* y^s + \cdots + y^n$ with $c_s^* \neq 0$ and $G = c_s y^s + \cdots + y^n$ (we might have *a priori* $c_s = 0$). By a convexity argument, we are reduced to show that $c_s, c_s^* \in \mathbb{K}((x))$ have same x-adic initial term. We have

$$v_\lambda(c_s^* y^s - c_s y^s) \geqslant v_\lambda(G - G^*) > v_\lambda(G^*) + \sigma \geqslant v_\lambda(G^*) + m_\lambda(G^*) = v_\lambda(c_s^* y^s),$$

the first inequality by definition of v_λ, the second inequality by Proposition 2.18, the last inequality by hypothesis $\sigma \geqslant m_\lambda(F)$ combined with Corollary 2.24, and the last equality by Lemma 2.23 since G^* is λ-monic (Proposition 2.18). We deduce that $\mathrm{in}_\lambda(c_s^* y^s) = \mathrm{in}_\lambda(c_s y^s)$, from which it follows that $c_s, c_s^* \in \mathbb{K}((x))$ have same initial term as required. The assertion for H is proved in the same way, focusing now on the leading term of H. □

Assuming F non degenerated, its irreducible factorization in $\mathbb{K}((x))[y]$ is deduced from the irreducible factorization of its lower edges polynomials. Hence, Lemma 2.32 ensures that knowing G and H at precision $\sigma \geqslant m_\lambda(F)$ is sufficient to detect all remaining *irreducible* factors of F.

2.3.4 Divide and conquer

We apply now recursively `PartialFacto` to G and H with respect to some well chosen slopes λ_G and λ_H.

Definition 2.33. *Let $n > s$. The* average slope *of $P = \sum_{i=s}^n p_i y^i \in \mathbb{K}((x))[y]$ is*

$$\lambda_P := -\frac{v_0(p_n) - v_0(p_s)}{n - s} \in \mathbb{Q}.$$

In other words, $-\lambda_P$ is the slope of the segment joining the two extremities of the lower boundary $\Lambda(P)$.

This slope is chosen so that the λ_P-valuation of the leading term and the initial term of P coincide. Equivalently, it satisfies

$$m_{\lambda_P}(P) = a_{\lambda_P}(P) = b_{\lambda_P}(P). \tag{9}$$

We deduce:

Proposition 2.34. *Let λ and G, H as above, and suppose G, H of positive y-degree.*

- *If P divides G then $\lambda_P \geqslant \lambda$.*
- *If P divides H then $\lambda_P \leqslant \lambda$.*

In both cases, we have $m_{\lambda_P}(P) \leqslant m_\lambda(P)$.

Proof. Let $P = a_s y^s + \cdots + a_n y^n$ with $a_s, a_n \neq 0$ and $n > s$. If P divides G, Lemma 2.31 implies $v_\lambda(a_s y^s) \geqslant v_\lambda(a_n y^n)$, which implies $\lambda_P \geqslant \lambda$ by Definition 2.33. On the other hand, (9) implies that

$$m_{\lambda_P}(P) = a_{\lambda_P}(P) = v_{\lambda_P}(a_s y^s) - v_{\lambda_P}(P).$$

Let $i \geqslant s$ such that $v_{\lambda_P}(P) = v_{\lambda_P}(a_i y^i)$. We get

$$v_{\lambda_P}(a_s y^s) - v_{\lambda_P}(a_i y^i) \leqslant v_\lambda(a_s y^s) - v_\lambda(a_i y^i) \leqslant v_\lambda(a_s y^s) - v_\lambda(P) = m_\lambda(P),$$

the first inequality since $(s-i)\lambda_P \leqslant (s-i)\lambda$, the second inequality by Definition of $v_\lambda(P)$, and the last equality by Lemma 2.31. It follows that $m_{\lambda_P}(P) \leqslant m_\lambda(P)$, as required. If P divides H, Lemma 2.31 forces $s = 0$ and $v_\lambda(P) = v_\lambda(a_0) \leqslant v_\lambda(a_n y^n)$, hence $\lambda_P \leqslant \lambda$. By (9), we get

$$m_{\lambda_P}(P) = b_{\lambda_P}(P) = v_{\lambda_P}(a_n y^n) - v_{\lambda_P}(P).$$

On the one hand, we have $v_{\lambda_P}(P) \geqslant v_0(a_0) = v_\lambda(P)$. On the other hand $\lambda_P \leqslant \lambda$ implies $v_{\lambda_P}(a_n y^n) \leqslant v_\lambda(a_n y^n)$. We get

$$m_{\lambda_P}(P) \leqslant v_\lambda(a_n y^n) - v_\lambda(P) = b_\lambda(P) = m_\lambda(P),$$

the two equalities by Lemma 2.31. □

Definition 2.35. *Let λ be fixed and let $P \in \mathbb{K}((x))[y]$. Given a λ-precision σ, we denote $\sigma_P = \sigma'(\lambda, \lambda_P, \sigma, P)$ the precision induced by* (8) *with $\lambda' = \lambda_P$.*

We deduce the following key uniform upper bound for σ_P.

Proposition 2.36. *Suppose that $\sigma \geqslant m_\lambda(F)$. If P divides G or H, then*

$$m_{\lambda_P}(P) \leqslant \sigma_P \leqslant \sigma + m_\lambda(F).$$

Proof. The inequality $m_{\lambda_P}(P) \leqslant \sigma_P$ follows from Corollary 2.29. By Lemma 2.30, we get $\sigma_P \leqslant \sigma + m_{\lambda_P}(P)$. By Proposition 2.34, we get $m_{\lambda_P}(P) \leqslant m_\lambda(P)$. Since P divides G, we have $m_\lambda(P) \leqslant m_\lambda(G)$ by Corollary 2.24. By Lemma 2.32, we have $m_\lambda(G) = m_\lambda(G^*)$, and Corollary 2.24 again gives $m_\lambda(G^*) \leqslant m_\lambda(F)$. The claim follows. □

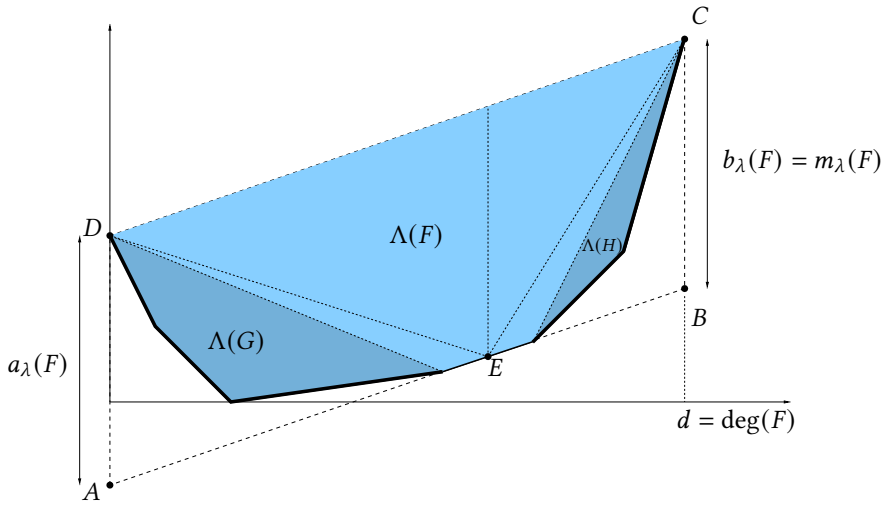

Figure 5: Illustrated proof of Proposition 2.37.

The last key result ensures that using the slopes λ_G and λ_H lead to a divide and conquer strategy. Given $P \in \mathbb{K}((x))[y]$, we denote in what follows by V_P the Euclidean volume of the convex hull of $\Lambda(P)$.

Proposition 2.37. *Let $F \in \mathbb{K}((x))[y]$ and suppose that $\lambda = \lambda_F$ (as will be the case at the recursive calls). Let $G, H \in \mathbb{K}((x))[y]$ as defined above.*

1. $dm_{\lambda_F}(F)/2 \leqslant V_F \leqslant dm_{\lambda_F}(F)$.
2. $V_F = 0$ *if and only if* $\Lambda(F)$ *is one-sided, in which case its slope is* λ_F.
3. *We have* $(V_G + V_H) \leqslant V_F/2$.

Proof. We still denote $\Lambda(F)$ the convex hull of the lower boundary $\Lambda(F)$. Let $ABCD$ be the smallest parallelogram with two vertical sides containing $\Lambda(F)$ such that C and D are respectively the right end point and the left end point of $\Lambda(F)$ and (AD) and (BC) are vertical. See Figure 5.

Denote $\lambda = \lambda_F$ for short. The line (AB) has equation $i + j\lambda = v_\lambda(F)$, and the segments $[AD]$ and $[BC]$ have both length $a_\lambda(F) = b_\lambda(F) = m_\lambda(F)$ (Definition 2.22) by choice of the average slope. Hence $ABCD$ has volume $dm_\lambda(F)$, which gives $V_F \leqslant dm_\lambda(F)$. By construction, there exists $E \in [AB] \cap \Lambda(F)$ and by convexity, the triangle CDE is contained in $\Lambda(F)$. Since $\mathrm{Vol}(CDE) = \frac{1}{2}\mathrm{Vol}(ABCD)$, the inequality $dm_{\lambda_F}(F)/2 \leqslant V_F$ follows, proving first point. The second item is immediate. Since $\Lambda(G)$ is the Minkowski summand of $\Lambda(F)$ whose all minus slopes are strictly greater than λ (Proposition 2.34), we may suppose that (up to translation) $\Lambda(G) \subset \Lambda(F)$ with left end point D and right end point $I \in [AE]$. By convexity, $\Lambda(G) \subset AED$ and $V_G \leqslant \mathrm{Vol}(AED)$. In the same way, we find $V_H \leqslant \mathrm{Vol}(BCE)$. On the other hand, we have $V_F \geqslant V_G + V_H + \mathrm{Vol}(CDE)$. We conclude thanks to the relation $\mathrm{Vol}(CDE) = \mathrm{Vol}(AED) + \mathrm{Vol}(BCE)$. □

Remark 2.38. The partial factorization of F with respect to λ_F is $F = G^*Q^*H^*$ where $Q^* = P_1^* \cdots P_k^*$ has a one-sided lower boundary slope λ_F (which is $[AB] \cap \Lambda(F)$ on Figure 5). However, although we use the terminology "slope," the rational λ_F is generally not a slope of $\Lambda(F)$. In such a case, the intersection $[AB] \cap \Lambda(F)$ is reduced to a point and the partial λ_F-factorization of F is simply $F = G^*H^*$. An important point is that G^* and H^* are not trivial factors as soon as $\Lambda(F)$ has several slopes.

2.3.5 Proving Theorem 1.8

In the following algorithm, σ_G, σ_H are defined by Definition 2.35, in terms of the input (λ, σ) and the current slopes λ_G, λ_H.

Algorithm 1: `Facto`(F, λ, σ)

Input: $F \in \mathbb{K}((x))[y]$ monic non degenerated, $\lambda \in \mathbb{Q}$ and $\sigma \geqslant m_\lambda(F)$
Output: The irreducible factors F with relative λ-precision $\sigma - m_\lambda(F)$

1 **if** $\deg(F) \leqslant 1$ **then return** $[F]$;
2 $[P_0, P_1, \ldots, P_k, P_\infty] \leftarrow$ `PartialFacto`(F, λ, σ);
3 $G \leftarrow P_0$, $H \leftarrow P_\infty$;
4 **if** $\deg(G) = 0$ **then** $L_G \leftarrow [\,]$ **else** $L_G \leftarrow$ `Facto`(G, λ_G, σ_G);
5 **if** $\deg(H) = 0$ **then** $L_H \leftarrow [\,]$ **else** $L_H \leftarrow$ `Facto`(H, λ_H, σ_H);
6 **return** $[P_1, \ldots, P_k] \cup L_G \cup L_H$

Theorem 2.39. *Given $F \in \mathbb{K}((x))[y]$ non degenerate with irreducible factors $F_1^*, \ldots, F_s^*$ and given $\sigma \geqslant m_\lambda(F)$, running* `Facto`$(F, \lambda, \sigma)$ *returns a list of irreducible monic coprime polynomials $F_1, \ldots, F_s \in \mathbb{K}((x))[y]$ such that*

$$v_\lambda(F - F_1 \cdots F_s) - v_\lambda(F) > \sigma$$

within $\tilde{O}(d\sigma)$ operations in $\mathbb{K}$. Moreover,

$$v_\lambda(F_i - F_i^*) - v_\lambda(F_i^*) > \sigma - m_\lambda(F)$$

for all $i = 1, \ldots, s$.

We will need the following lemma.

Lemma 2.40. *If $v_\lambda(A^* - A) > v_\lambda(A) + \sigma$ and $v_\lambda(B - B^*) > v_\lambda(B) + \sigma$, then $v_\lambda(A^*B^* - AB) > v_\lambda(AB) + \sigma$.*

Proof. Follows from $A^*B^* - AB = A^*(B^* - B) + B(A^* - A)$ together with $v_\lambda(A) = v_\lambda(A^*)$ and $v_\lambda(B) = v_\lambda(B^*)$. □

Proof of Theorem 2.39.
Correctness. By induction on the number of recursive calls. If the algorithm stops at Step 2, then the result follows from Proposition 2.19. Else, we know that G and H are not degenerated (Lemma 2.32) and approximate G^* and H^* with relative λ-precision σ (Proposition 2.19). As $\sigma_G \geqslant m_{\lambda_G}(G)$ (Proposition 2.36), we deduce by induction that $\texttt{Facto}(G, \lambda_G, \sigma_G)$ returns some approximants $G_1, \ldots, G_t$ of the irreducible factors $G_1^*, \ldots, G_t^*$ of G such that

$$v_{\lambda_G}(G - G_1 \cdots G_t) - v_{\lambda_G}(G) > \sigma_G, \tag{10}$$

with moreover

$$v_{\lambda_G}(G_i - G_i^*) - v_{\lambda_G}(G_i^*) > \sigma_G - m_{\lambda_G}(G) \quad \forall i = 1, \ldots, t. \tag{11}$$

Corollary 2.27 and (10) forces

$$v_\lambda(G - G_1 \cdots G_t) - v_\lambda(G) > \sigma.$$

Since $v_\lambda(G) = v_\lambda(G^*)$ and $v_\lambda(G - G^*) - v_\lambda(G^*) > \sigma$, we deduce

$$v_\lambda(G^* - G_1 \cdots G_r) - v_\lambda(G^*) > \sigma.$$

In the same way, the induction hypothesis ensures that $\texttt{Facto}(H, \lambda_H, \sigma_H)$ computes an approximate irreducible factorization of H^* such that

$$v_\lambda(H^* - H_1 \cdots H_u) - v_\lambda(H^*) > \sigma.$$

We have $F = G^* P_1^* \cdots P_k^* H^*$ and we have too $v_\lambda(P_i^* - P_i) - v_\lambda(P_i^*) > \sigma$ (Proposition 2.19). The polynomials $(F_1, \ldots, F_s) = (P_1, \ldots, P_k, G_1, \ldots, G_t, H_1, \ldots, H_u)$ approximate the irreducible factors of F and Lemma 2.40 implies

$$v_\lambda(F - F_1 \cdots F_r) - v_\lambda(F) > \sigma$$

as required. There remains to show that $v_\lambda(F_i - F_i^*) - v_\lambda(F_i^*) > \sigma - m_\lambda(F)$ for all i. This is true for the factors P_j by Proposition 2.18. Let us consider a factor $A = G_i$. As $\lambda_G \geqslant \lambda$, (11) combined with (8) gives

$$v_{\lambda_G}(A - A^*) > v_{\lambda_G}(A) + \sigma + v_\lambda(G) - v_{\lambda_G}(G) + d_G(\lambda_G - \lambda).$$

Denote B the (truncated) cofactor of A in G. Using $v_\lambda(A - A^*) + d_A(\lambda_G - \lambda) \geqslant v_{\lambda_G}(A - A^*)$ (Lemma 2.25) together with $d_G = d_A + d_B$, $v_\lambda(G) = v_\lambda(A) + v_\lambda(B)$, $v_{\lambda_G}(G) = v_{\lambda_G}(A) + v_{\lambda_G}(B)$ and Lemma 2.28, the previous inequality implies that

$$v_\lambda(A - A^*) > v_\lambda(A) + \sigma + m_{\lambda_G}(B) - m_\lambda(B).$$

As $m_{\lambda_G}(B) \geqslant 0$ and $m_\lambda(B) \leqslant m_\lambda(F)$ (Corollary 2.24), we get the desired inequality

$$v_\lambda(A - A^*) - v_\lambda(A) > \sigma - m_\lambda(F).$$

We prove in a similar way the analogous assertion if $A = H_i$ is a factor of H.

Complexity. There is at most $1 + \lceil \log_2(V_F) \rceil = O(\log_2(dm_\lambda(F))) = O(\log_2(d\sigma))$ recursive calls thanks to Proposition 2.37 (the +1 due to the fact that the initial slope λ can take any value). At each level of the tree of recursive calls, the procedure `PartialFacto` is called on a set of polynomials P dividing G or H and whose degree sum is at most $d_G + d_H \leqslant d$, and with λ_P-precision σ_P for each P. By Proposition 2.36, $\sigma_P \leqslant \sigma + m_\lambda(F) \leqslant 2\sigma$ for all P, and we conclude thanks to Proposition 2.19. □

Proof of Theorem 1.8. Theorem 1.8 follows straightforwardly from Theorem 2.39, taking into account the cost of the factorizations (7) of the various quasi-homogeneous initial components. These factorizations are not trivial only when λ is a slope of $\Lambda(F)$, in which case the degree of the underlying univariate factorization corresponds to the lattice length of the edge of slope λ. □

3 Application to convex-dense bivariate factorization

This section is dedicated to derive from Theorem 1.8 a fast algorithm for factoring a bivariate polynomial $F \in \mathbb{K}[x, y]$. We follow closely [24], which generalizes the usual factorization algorithm of [11, 13] to the case $F(0, y)$ non separable. To be consistent with [11, 24], we denote from now on by $\mathcal{F}_i$ the factors of F in $\mathbb{K}((x))[y]$ and by F_j the factors of F in $\mathbb{K}[x, y]$.

3.1 The recombination problem

In all what follows, we assume that the input $F \in \mathbb{K}[x, y]$ is primitive and separable with respect to y (see [12] for fast separable factorization). We let $d := \deg_y(F)$. We normalize F by requiring that its coefficient attached to the right end point of $\Lambda(F)$ equals 1. Up to permutation, F admits a unique factorization

$$F = F_1 \cdots F_\rho \in \mathbb{K}[x, y], \tag{12}$$

where each $F_j \in \mathbb{K}[x, y]$ is irreducible and normalized. Also, F admits a unique analytic factorization of shape

$$F = u\mathcal{F}_1 \cdots \mathcal{F}_s \in \mathbb{K}[[x]][y], \tag{13}$$

with $\mathcal{F}_i \in \mathbb{K}[[x]][y]$ irreducible with leading coefficient x^{n_i}, $n_i \in \mathbb{N}$ and $u \in \mathbb{K}[x]$, $u(0) \neq 0$. We thus have

$$F_j = c_j \mathcal{F}_1^{v_{j1}} \cdots \mathcal{F}_s^{v_{js}}, \quad j = 1, \ldots, \rho, \tag{14}$$

for some unique $v_{ji} \in \{0, 1\}$, and with $c_j \in \mathbb{K}[x]$, $c_j(0) = 1$. The recombination problem consists to compute the exponent vectors

$$v_j = (v_{j1}, \ldots, v_{js}) \in \{0, 1\}^s$$

for all $j = 1, \dots, \rho$. Then, the computation of the F_j's follows easily. Since F is separable by hypothesis, the vectors v_j form a partition of $(1, \dots, 1)$ of length ρ. In particular, they form up to reordering the reduced echelon basis of the vector subspace

$$V := \langle v_1, \dots, v_\rho \rangle \subset \mathbb{K}^s$$

that they generate over $\mathbb{K}$ (in fact over any field). Hence, solving recombinations mainly reduces to find a system of $\mathbb{K}$-linear equations that determine $V \subset \mathbb{K}^s$.

Let $\mu = (\mu_1, \dots, \mu_s) \in \mathbb{K}^s$. Applying the logarithmic derivative with respect to y to (14) and multiplying by F we get

$$\mu \in V \iff \exists \alpha_1, \dots, \alpha_\rho \in \mathbb{K} \quad | \quad \sum_{i=1}^{s} \mu_i \widehat{\mathcal{F}_i} \partial_y \mathcal{F}_i = \sum_{j=1}^{\rho} \alpha_j \widehat{F_j} \partial_y F_j, \tag{15}$$

with notations $\widehat{F_j} = F/F_j$ and $\widehat{\mathcal{F}_i} = F/\mathcal{F}_i$. The reverse implication holds since the F_j's are supposed to be separable [13, Lemma 1]. In [11], the author shows how to derive from (15) a finite system of linear equations for V that depends only on the $\mathcal{F}_i$'s truncated with x-adic precision $d_x + 1$, *assuming* $F(0, y)$ *separable of degree* d. For our purpose, we will rather consider v_λ-adic truncation of the $\mathcal{F}_i$'s for a suitable λ, under the weaker hypothesis that F is non degenerated.

3.2 Residues and recombinations

In what follows, we fix $\lambda = m/q \in \mathbb{Q}$. Given $G \in \mathbb{K}((x))[y]$ and $\sigma \in \mathbb{Q}$, the v_λ-truncation of G with precision σ is

$$[G]_\lambda^\sigma := \sum_{j+i\lambda \leqslant \sigma} g_{ij} x^j y^i \in \mathbb{K}[x^{\pm 1}][y].$$

If $\lambda = 0$, this is the classical Gauss (or x-adic) truncation $[G]_0^\sigma = G \mod x^{\sigma+1}$. If $G \in \mathbb{K}[x, y]$, we can define the λ-degree of G,

$$d_\lambda(G) := \max(j + i\lambda, g_{ij} \neq 0). \tag{16}$$

Note that $G = [G]_\lambda^{d_\lambda(G)}$. Moreover, we have

$$d_\lambda(GH) = d_\lambda(G) + d_\lambda(H) \quad \text{and} \quad d_\lambda(G + H) \leqslant \max(d_\lambda(G), d_\lambda(H)).$$

Let $\mu \in \mathbb{F}^r$. Given the factorization (13), we let

$$G_\mu := \sum_{i=1}^{s} \mu_i \big[\widehat{\mathcal{F}_i} \partial_y \mathcal{F}_i\big]_\lambda^{d_\lambda(F)} \in \mathbb{K}[x, y]. \tag{17}$$

Recall the notation $d = \deg_y(F)$. We denote $y_1, \dots, y_d \in \overline{\mathbb{K}(x)}$ the roots of F. We denote by $\rho_k = \rho_k(\mu)$ the residues of G_μ/F at y_k, that is

$$\rho_k := \frac{G_\mu(x, y_k)}{\partial_y F(x, y_k)} \in \overline{\mathbb{K}(x)}, \qquad k = 1, \dots, d.$$

These residues are well defined since F is separable. The next key result is mainly a consequence of [24, Proposition 8.7].

Proposition 3.1. *Suppose $\lambda \geqslant 0$ and F not degenerated. Then $\mu \in V$ if and only if $\rho_k \in \overline{\mathbb{K}}$ for all $k = 1, \dots, d$.*

Proof. The direct implication follows from (15). Let us prove the converse, assuming that the residues ρ_k are constant. Let $\tau = \tau_\lambda$ as defined by (4). Given $Q \in \mathbb{K}((x))[y]$, we denote for short

$$\tau_0(Q) = x^{-v_0(\tau(Q))}\tau(Q) \in \mathbb{K}[[x]][y]. \tag{18}$$

Hence $\tau_0(F) \in \mathbb{K}[x, y]$ is a primitive polynomial with primitive factors $\tau_0(F_j)$ in $\mathbb{K}[x, y]$ and $\tau_0(\mathcal{F}_i)$ in $\mathbb{K}[[x]][y]$. Following (5), we get

$$n := \deg_x(\tau_0(F)) = q(d_\lambda(F) - v_\lambda(F)).$$

Recall the notation $\lambda = m/q$. Let us consider

$$G_\mu^0 := \sum_{i=1}^{r} \mu_i \big[\tau_0\widehat{(\mathcal{F}_i)}\partial_y\tau_0(\mathcal{F}_i)\big]_0^{n+m}. \tag{19}$$

In other words, G_μ^0 coincides with the polynomial defined by (17) when considering the recombinations of the analytic factors $\tau_0(\mathcal{F}_i)$ of $\tau_0(F)$ using the Gauss valuation v_0, except that the precision is (for now) $n + m$ instead of n. Let $\phi_k(x) := x^{-m}y_k(x^q)$. We have $\tau(F)(x, \phi_k(x)) = F(x^q, y_k(x^q)) = 0$ for all k so $\tau_0(F)$ has roots $\phi_1, \dots, \phi_d$.

Claim (see below for the proof)**.** *We have*

$$\frac{G_\mu^0}{\partial_y\tau_0(F)} = \tau\Big(\frac{G_\mu}{\partial_y F}\Big).$$

Assuming that, the residues of $G_\mu^0/\tau_0(F)$ at the roots of $\tau_0(F)$ are

$$\frac{G_\mu^0(x, \phi_k(x))}{\partial_y\tau_0(F)(x, \phi_k(x))} = \frac{G_\mu(x^q, y_k(x^q))}{\partial_y F(x^q, y_k(x^q))} = \rho_k(x^q) \in \overline{\mathbb{K}},$$

hence are constant by assumption. It follows from [24, Lemma 3.8] that G_μ^0 is a $\overline{\mathbb{K}}$-linear combinations of $E_j\partial_y E_j$, where the polynomials $E_j \in \overline{\mathbb{K}}[x, y]$ are the irreducible factors of $\tau_0(F)$ over $\overline{\mathbb{K}}$. Since $\deg_x E_j\partial_y E_j \leqslant \deg_x \tau_0(F) \leqslant n$, this implies

$$G_\mu^0 = [G_\mu^0]_0^n = \sum_{i=1}^{r} \mu_i \big[\tau_0\widehat{(\mathcal{F}_i)}\partial_y\tau_0(\mathcal{F}_i)\big]_0^n,$$

the second equality by (19), since $m \geqslant 0$ by assumption. Since F is separable and not degenerated, so is $\tau_0(F)$. Thus, we can apply [24, Proposition 8.7] to $\tau_0(F)$ and we deduce that G_μ^0 is a $\mathbb{K}$-linear combination of the polynomials $\tau_0\widehat{(F_j)}\partial_y\tau_0(F_j)$, which in turns implies that G_μ is a $\mathbb{K}$-linear combination of the polynomials $\widehat{F_j}\partial_y F_j$. Hence $\mu \in V$ thanks to (15), as required. □

Remark 3.2. The assumption F not degenerated is crucial to solve recombinations with v_λ-precision $d_\lambda(F)$. Otherwise, we might need to compute the $\mathcal{F}_i$'s with a higher precision. We refer the reader to [24] for various options to solve the recombination problem for degenerated polynomials in the x-adic case.

Proof of the claim. As $v_\lambda(y) = \lambda$, we deduce from (17) that

$$yG_\mu = \sum_{i=1}^{s} \mu_i \big[\widehat{\mathcal{F}}_i y \partial_y \mathcal{F}_i\big]_\lambda^{d_\lambda(F)+\lambda}. \tag{20}$$

Now, using (4) and (16), we leave the reader to check that for all $H \in \mathbb{K}(x)[y]$ and all $a \in \mathbb{Q}$, we have:

$$\tau(y\partial_y H) = y\partial_y(\tau(H)) \qquad \text{and} \qquad \tau([H]_\lambda^a) = [\tau(H)]_0^{qa}. \tag{21}$$

Combined with (20), we get

$$\tau(yG_\mu) = \sum_i \mu_i \big[\tau(\widehat{\mathcal{F}}_i) y \partial_y \tau(\mathcal{F}_i)\big]_0^{qd_\lambda(F)+m}.$$

Using now (18) together with $v_0(\tau(\widehat{\mathcal{F}}_i)) + v_0(\tau(\mathcal{F}_i)) = v_0(\tau(F)) = qv_\lambda(F)$, we get

$$\tau(yG_\mu) = yx^{v_0(\tau(F))} \sum_i \mu_i \big[\tau_0(\widehat{\mathcal{F}}_i)\partial_y \tau_0(\mathcal{F}_i)\big]_0^{qd_\lambda(F)-qv_\lambda(F)+m} = yx^{v_0(\tau(F))} G_\mu^0.$$

It follows that

$$\frac{G_\mu^0}{\partial_y \tau_0(F)} = \frac{yx^{v_0(\tau(F))}G_\mu^0}{y\partial_y \tau(F)} = \frac{\tau(yG_\mu)}{\tau(y\partial_y F)} = \tau\Big(\frac{G_\mu}{\partial_y F}\Big),$$

the first equality by (18) and the second equality using (21) again. □

3.3 Computing equations for V

Since F is separable, ρ_k belongs to the separable closure of $\mathbb{K}(x)$ and we can talk about the derivative of ρ_k. Hence, an obvious necessary condition for that $\rho_k \in \overline{\mathbb{K}}$ is that its derivative vanishes. More precisely, we have the following lemma.

Lemma 3.3. *Let $p \geqslant 0$ be the characteristic of $\mathbb{K}$. If $\rho_k' = 0$ then $\rho_k \in \overline{\mathbb{K}(x^p)}$. If moreover $p = 0$ or $p \geqslant 2d(d_\lambda(F) - v_\lambda(F))$, then $\rho_k \in \overline{\mathbb{K}}$.*

Proof. If $\rho_k' = 0$, then clearly $\rho_k \in \overline{\mathbb{K}(x^p)}$. If $p = 0$, the claim follows. If $p > 0$, we consider the polynomial $\tau_0(F)$ defined above, of x-degree $n = q(d_\lambda(F) - v_\lambda(F))$. Its residue is $\rho_k(x^q)$ which thus lives in $\overline{\mathbb{K}(x^{pq})}$. Hence, it's straightforward to check that we can divide the bound $p \geqslant 2dn$ of [8, Lemma 2.4] by q in this context. □

Let us consider the $\mathbb{K}$-linear operator

$$\mathrm{D} : \begin{cases} \mathbb{K}(x)[y] \longrightarrow \mathbb{K}(x)[y] \\ \qquad G \longmapsto \Big(G_x F_y - G_y F_x\Big)F_y - \Big(F_{xy}F_y - F_{yy}F_x\Big)G \end{cases} \tag{22}$$

with the standard notations F_y, F_{xy}, etc. for the partial derivatives.

Lemma 3.4. *We have $\rho_k' = 0$ for all $k = 1, \dots, d$ if and only if F divides $\mathrm{D}(G_\mu)$ in the ring $\mathbb{K}(x)[y]$.*

Proof. Combining $\rho_k(x) = \frac{G_\mu(x,y_k)}{F_y(x,y_k)}$ and $y_k'(x) = -\frac{F_x(x,y_k)}{F_y(x,y_k)}$, we get

$$\rho_k'(x) = \frac{\mathrm{D}(G_\mu)(x,y_k)}{F_y^3(x,y_k)}.$$

Thus $\rho_k' = 0$ if and only if $\mathrm{D}(G_\mu)$ vanishes at all roots of F, seen as a polynomial in y. The result follows since F is separable. □

Let us denote $D_\mu := \mathrm{D}(G_\mu)$ for short. We will need the following lemma.

Lemma 3.5. *Suppose $\lambda \geqslant 0$. Then $3v_\lambda(F) \leqslant v_\lambda(D_\mu)$ and $d_\lambda(D_\mu) \leqslant 3d_\lambda(F)$.*

Proof. For any $Q \in \mathbb{K}[x,y]$, the support of xQ_x and yQ_y is contained in the support of Q. Hence $v_\lambda(Q) \leqslant v_\lambda(xQ_x)$ and $v_\lambda(Q) \leqslant v_\lambda(yQ_y)$ while $d_\lambda(Q) \geqslant d_\lambda(xQ_x)$ and $d_\lambda(Q) \geqslant d_\lambda(yQ_y)$. As $v_\lambda(x) = d_\lambda(x) = 1$ and $d_\lambda(y) = v_\lambda(y) = \lambda \geqslant 0$, we get

$$v_\lambda(Q_x), v_\lambda(Q_y) \geqslant v_\lambda(Q) \quad \text{and} \quad d_\lambda(Q_x), d_\lambda(Q_y) \leqslant d_\lambda(Q). \tag{23}$$

In particular, we get from (17) that $v_\lambda(G_\mu) \geqslant v_\lambda(F)$. On the other hand we have $d_\lambda(G_\mu) \leqslant d_\lambda(F)$ by the very definition (17). The claim then follows from (22), using moreover that v_λ and $-d_\lambda$ are valuations. □

Lemma 3.4 suggests to compute the v_λ-adic Euclidean division of D_μ by F up to a sufficient precision to test divisibility in $\mathbb{K}(x)[y]$. A difficulty is that F is not necessarily λ-monic, hence we do not have access to Proposition 2.16. To solve this issue, we adapt [24, Section 5] to our context. We get:

Proposition 3.6. *Suppose $\lambda \geqslant 0$. Given $\mathcal{F}_1, \dots, \mathcal{F}_s$ with relative λ-precision $d_\lambda(F) - v_\lambda(F)$, we can compute a linear map*

$$\phi : \mathbb{K}^s \to \mathbb{K}^N, \quad N \in O\big(d\big(d_\lambda(F) - v_\lambda(F)\big)\big)$$

such that $\mu \in \ker(\phi)$ if and only if $F | D_\mu$, using at most $\tilde{O}(sN)$ operations in $\mathbb{K}$.

Proof. Note first that G_μ only depends on the $\mathcal{F}_i$'s with relative λ-precision $d_\lambda(F) - v_\lambda(F)$ by (23) and Lemma 2.40. Let $0 \leqslant \alpha < q$ and $k \in \mathbb{Z}$ be the unique integers such that $\tilde{F} := \tau(x^k y^\alpha F)$ satisfies

$$q | \deg_y(\tilde{F}) = d + \alpha \quad \text{and} \quad 0 \leqslant v_0(\tilde{F}) < q. \tag{24}$$

These conditions ensure that both $\tilde{F}$ and its leading coefficient $c := \mathrm{lc}_y(\tilde{F})$ lie in the subring $\mathbb{B}_\lambda \subset \mathbb{K}[x,y]$ (Lemma 2.1). Let $k' = k - 2v_\lambda(F)$ and $\tilde{D}_\mu := \tau(x^{k'} y^\alpha D_\mu)$. Note that F divides D_μ in $\mathbb{K}(x)[y]$ if and only if $\tilde{F}$ divides $\tilde{D}_\mu$ in $\mathbb{K}(x)[y]$. Moreover, k' does not depend on μ so the map $\mu \mapsto \tilde{D}_\mu$ is $\mathbb{K}$-linear.

Claim. *We have* $v_0(\tilde{D}_\mu) \geqslant v_0(\tilde{F})$ *and* $\deg_x(\tilde{D}_\mu) \leqslant 3\deg_x(\tilde{F})$.

Proof of the claim. As $\lambda \geqslant 0$, Lemma 3.5 and $v_0(\tau(Q)) = qv_\lambda(Q)$ give

$$v_0(\tilde{D}_\mu) = q\big(k' + \alpha\lambda + v_\lambda(D_\mu)\big) \geqslant q\big(k + \alpha\lambda + v_\lambda(F)\big) = v_0(\tilde{F}) \geqslant 0.$$

In a similar way, using now $\deg_x(\tau(Q)) = qd_\lambda(Q)$, we get

$$\begin{aligned}\deg_x(\tilde{D}_\mu) &\leqslant q\big(k' + \alpha\lambda + 3d_\lambda(F)\big)\\ &= 2q\big(d_\lambda(F) - v_\lambda(F)\big) + q\big(k + \alpha\lambda + v_\lambda(F)\big)\\ &= 2\big(\deg_x(\tilde{F}) - v_0(\tilde{F})\big) + \deg_x(\tilde{F})\\ &\leqslant 3\deg_x(\tilde{F}).\end{aligned}$$

□

As $0 \leqslant v_0(\tilde{F}) \leqslant v_0(\tilde{D}_\mu)$, $\tilde{F}$ divides $\tilde{D}_\mu$ in $\mathbb{K}(x)[y]$ if and only if it divides $\tilde{D}_\mu$ in $\mathbb{K}[x][y]$ by Gauss' lemma. To reduce to the monic case, we localize $\mathbb{K}[x]$ at some prime $a \in \mathbb{K}[x^q]$ coprime to $c := \mathrm{lc}_y(\tilde{F})$. The Euclidean division

$$\tilde{D}_\mu = Q_\mu\tilde{F} + R_\mu \in \mathbb{K}[x]_{(a)}[y] \tag{25}$$

is now well defined. Any $Q \in \mathbb{B}_\lambda \subset \mathbb{K}[x,y]$ has a unique a-adic expansion

$$Q = \sum_{i=0}^{\lfloor \deg(Q)/\deg(a)\rfloor} q_i(x,y)a(x)^i \quad \text{with} \quad q_i \in \mathbb{B}_\lambda \quad \text{and} \quad \deg_x q_i < \deg a. \tag{26}$$

Note that $q_i \in \mathbb{B}_\lambda$ since $a \in \mathbb{B}_\lambda$. Let $\{Q\}_m^n = \sum_{i=m}^n q_i a^i$ and $\{Q\}^n = \{Q\}_0^n$. Since $\deg_x(\tilde{D}_\mu) \leqslant 3\deg_x(\tilde{F})$, we deduce from (the proof of) [24, Lemma 5.2] that $\tilde{F}$ divides $\tilde{D}_\mu$ if and only if

$$\{Q_\mu\}_m^n = \{R_\mu\}^n = 0, \quad \text{with} \quad m := \left\lfloor \frac{2d_x}{\deg a} \right\rfloor + 1 \quad \text{and} \quad n := \left\lceil \frac{3d_x}{\deg a} \right\rceil,$$

where $d_x = \deg_x(\tilde{F})$. We have $d_x = q(d_\lambda(F) - v_\lambda(F))$ by (24). Since both polynomials $\{Q_\mu\}_m^n$ and $\{R_\mu\}^n$ live in $\mathbb{B}_\lambda$, we deduce from Corollary 2.3 that their supports have size $O(d(d_\lambda(F) - v_\lambda(F)))$. The linear map

$$\phi(\mu) := \big(\{Q_\mu\}_m^n/a^m, \{R_\mu\}^n\big)$$

thus satisfies the conditions of Proposition 3.6. Let us look at complexity issues. If $Q_1, Q_2 \in \mathbb{B}_\lambda$ have x-degrees $O(d_x)$ and relative y-degrees $\deg_y(Q_i) - v_y(Q_i) \in O(d)$, we compute $\{Q_1\}^n$, $\{Q_2\}^n$ and $\{Q_1Q_2\}^n$ in time $\tilde{O}(dd_x/q)$ thanks to Proposition 2.5 since all operations in (26) take place in $\mathbb{B}_\lambda$. We have $c \in \mathbb{K}[x^q] \subset \mathbb{B}_\lambda$ invertible modulo a, and computing $\{c^{-1}\}^n$ costs $\tilde{O}(d_x/q)$. Then, adapting the proof of Proposition 2.8 in the a-adic case, we compute (25) with a-adic precision n and thus $\phi(\mu)$ in time $\tilde{O}(dd_x/q) = \tilde{O}(d(d_\lambda(F) - v_\lambda(F)))$. To build the matrix of ϕ, we compute $\phi(\mu_i)$ where the μ_i's run over the canonical basis of $\mathbb{K}^s$. Given the $\mathcal{F}_i$'s with relative λ-precision $d_\lambda(F) - v_\lambda(F)$, computing $G_{\mu_i} = \big[\widehat{\mathcal{F}_i}\partial_y\mathcal{F}_i\big]_\lambda^{d_\lambda(F)}$ costs $\tilde{O}(d(d_\lambda(F) - v_\lambda(F))$ thanks to Corollary 2.13. Summing over all $i = 1, \ldots, s$, we get the result. □

Remark 3.7. We need to compute $a \in \mathbb{K}[x^q]$ coprime to c. As $a = a_0(x^q)$ and $c = c_0(x^q)$, we look for a_0 coprime to c_0. We have $\deg_x(c_0) \leqslant d_x/q = d_\lambda(F) - v_\lambda(F)$. If $\mathrm{Card}(\mathbb{K}) \geqslant d_\lambda(F) - v_\lambda(F)$, we use multipoint evaluation of c_0 at $\deg_x(c_0)$ distinct elements of $\mathbb{K}$ to find $z \in \mathbb{K}$ such that $c_0(z) \neq 0$, and we take $a(x) = x^q - z$. Otherwise, we follow a similar strategy in a finite extension of $\mathbb{K}$ of degree $O(\log(d_\lambda(F) - v_\lambda(F))$, considering now $a = a_0(x^q)$, with a_0 the minimal polynomial of z over $\mathbb{K}$. The cost fits in the aimed bound.

Corollary 3.8. *If $\mathbb{K}$ has characteristic zero or greater than $2d(d_\lambda(F) - v_\lambda(F))$ and F is non degenerated, then $(v_1, \ldots, v_\rho)$ is the reduced echelon basis of* $\ker(\phi)$.

Proof. Follows from Proposition 3.1, Lemma 3.3, Lemma 3.4 and Proposition 3.6. □

If $\mathbb{K}$ has small characteristic p, we need extra conditions to ensure $\rho_k \in \overline{\mathbb{K}}$. These conditions rely on linear algebra over the prime field $\mathbb{F}_p$ of $\mathbb{K}$. They are based on Niederreiter's operator, which was originally introduced for univariate factorization over finite fields [14], and used then for bivariate factorization in [11]. We deliberately do not go into the details here. We assume $\lambda \geqslant 0$. We introduce the $\mathbb{F}_p$-linear map

$$\psi : \begin{cases} \ker(\phi_{|\mathbb{F}_p^s}) \longrightarrow \mathbb{K}[x^p, y^p]_{pd_\lambda, p(d-1)} \\ \qquad\qquad \mu \longmapsto G_\mu^p - \partial_y^{p-1}(G_\mu F^{p-1}). \end{cases}$$

In contrast to [11], the subscripts indicate the λ-degree and the y-degree.

Proposition 3.9. *The map ψ is well-defined and $(v_1, \ldots, v_\rho)$ is the reduced echelon basis of* $\ker(\psi)$.

Proof. We have $\partial_y G_\mu^p = 0$ in characteristic p. Moreover, $\partial_y^p(y^i) = 0 \mod p$ for all $i \geqslant 0$. We deduce that $\partial_y(\psi(\mu)) = 0$, so $\psi(\mu)$ is a polynomial in y^p of y-degree $p(d-1)$. Since $d_\lambda(Q_y) \leqslant d_\lambda(Q)$ (proof of Lemma 3.5), $\psi(\mu)$ has λ-degree at most pd_λ. Since moreover $\mu \in \ker(\phi)$, we have $\rho_k' = 0$ by Proposition 3.6 and Lemma 3.5, hence $\rho_k \in \overline{\mathbb{K}(x^p)}$ by Lemma 3.3. This forces $\psi(\mu)$ to be a polynomial in x^p (see [11, Lemma 4]). Hence ψ is well-defined. The second claim follows from [11, Proposition 11] together with Proposition 3.1. □

Proposition 3.10. *Let $\lambda \geqslant 0$ and $N = d(d_\lambda(F) - v_\lambda(F))$. Assume $F \in \mathbb{K}[x, y]$ non degenerated. Given the factors $\mathcal{F}_1, \ldots, \mathcal{F}_s \in \mathbb{K}[[x]][y]$ of F with relative λ-precision $d_\lambda(F) - v_\lambda(F)$, we can solve the recombination problem with*

1. *$\tilde{O}(sN) + O(s^{\omega-1}N)$ operations in $\mathbb{K}$ if $p = 0$ or $p \geqslant 2N$,*
2. *$O(ks^{\omega-1}N)$ operations in $\mathbb{F}_p$ if $\mathbb{K} = \mathbb{F}_{p^k}$.*

Proof sketch. We can compute the reduced echelon basis of the kernel of a matrix of size $s \times N$ with coefficient in a field $\mathbb{L}$ with $O(s^{\omega-1}N)$ operations in $\mathbb{L}$ [19, Theorem 2.10]. Hence, the first point follows from Proposition 3.1 and Corollary 3.8. Suppose that $\mathbb{K} = \mathbb{F}_{p^k}$. Thus $\mathbb{K}$ is an $\mathbb{F}_p$-vector space of dimension k and it follows again from Proposition 3.1 that we can build the matrix of $\phi_{|\mathbb{F}_p^s}$ and compute a basis of its kernel

over $\mathbb{F}_p$ in the aimed cost. To build the matrix of ψ we use again the operator τ to reduce to the case $\lambda = 0$. We apply then [11, Proposition 13], using again that the complexity can be divided by q since we work in the sparse subring $\mathbb{B}_\lambda \subset \mathbb{K}[x, y]$ (in the non monic case, we localize at some $a \in \mathbb{K}[x^q]$ as in the proof of Proposition 3.1). The resulting complexity fits in the aimed cost. The matrix of ψ having size at most $s \times kN$ over $\mathbb{F}_p$, we conclude. □

3.4 Proving Theorem 1.2 and Corollary 1.5

The key point is to choose a good slope λ before applying Proposition 3.10. Let $F \in \mathbb{K}[x, y]$ of y-degree d with Newton polygon $N(F)$ and lower convex hull $\Lambda(F)$. Let $V = \mathrm{Vol}(N(F))$.

Lemma 3.11. *Let $\lambda := \lambda_F$ be the average slope of $\Lambda(F)$ (Definition 2.33). Assume that y does not divide F. Then*

$$V \leqslant d(d_\lambda(F) - v_\lambda(F)) \leqslant 2V.$$

Proof. It is a similar proof as that of Proposition 2.36. Consider the bounding parallelogram $ABCD$ of $N(F)$ with two vertical sides and two sides of slope $-\lambda$. See Figure 6. We have $\mathrm{Vol}(ABCD) = d(d_\lambda(F) - v_\lambda(F))$ which gives the first inequality. Consider I and J the left and right end points of $\Lambda(F)$ and let $K \in [BC] \cap N(F)$ and $L \in [AD] \cap N(F)$. Then

$$V \geqslant \mathrm{Vol}(IJK) + \mathrm{Vol}(IJL) = \frac{\mathrm{Vol}(IBCJ)}{2} + \frac{\mathrm{Vol}(IADJ)}{2} = \frac{\mathrm{Vol}(ABCD)}{2},$$

the inequality since IJK and IJL are contained in $N(F)$, and the first equality since (IJ) is parallel to (AD) and (CD) by choice of $\lambda = \lambda_F$. The result follows. □

Previous results lead to algorithm `Factorization` below.

Algorithm 2: `Factorization`(F)

Input: $F \in \mathbb{K}[x, y]$ primitive, separable in y, non degenerated, with $\lambda_F \geqslant 0$.
Output: The irreducible factorization of F over $\mathbb{K}$

1 **if** y *divides* F **then** $L = [y]$ and $F \leftarrow F/y$ **else** $L \leftarrow [\,]$;
2 $\lambda \leftarrow \lambda_F$ and $\sigma \leftarrow d_\lambda(F) - v_\lambda(F) + m_\lambda(F)$;
3 $[\mathcal{F}_1, \ldots, \mathcal{F}_s] \leftarrow \mathtt{Facto}(F, \lambda, \sigma)$;
4 **if** $s = 1$ **then return** $L \cup [\mathcal{F}_1]$;
5 Compute the reduced echelon basis $(v_1, \ldots, v_\rho)$ of V using Proposition 3.10;
6 **for** $j = 1, \ldots, \rho$ **do**
7 Compute $\tilde{F}_j := [\mathrm{lc}_y(F) \prod_{i=1}^{s} \mathcal{F}_i^{v_{ji}}]_\lambda^{d_\lambda(F)}$;
8 Compute the primitive part F_j of $\tilde{F}_j$ with respect to y
9 **return** $L \cup [F_1, \ldots, F_\rho]$;

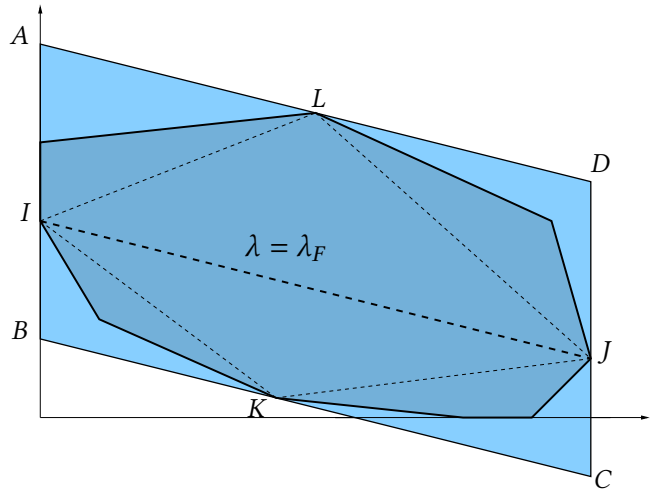

Figure 6: Proof of Lemma 3.11. In dark blue, the polygon $N(F)$; in light blue, its bounding parallelogram of slope λ_F.

Proposition 3.12. *Algorithm* `Factorization` *is correct. Up to the cost of univariate factorizations, it takes at most*

1. $\tilde{O}(sV) + O(s^{\omega-1}V)$ *operations in* $\mathbb{K}$ *if* $p = 0$ *or* $p \geqslant 4V$,
2. $O(ks^{\omega-1}V)$ *operations in* $\mathbb{F}_p$ *if* $\mathbb{K} = \mathbb{F}_{p^k}$.

Proof. By Theorem 2.39, Step 3 computes the $\mathcal{F}_i$'s with relative λ-precision at least $d_\lambda(F) - v_\lambda(F)$. Since $\lambda_F \geqslant 0$ by assumption, Proposition 3.10 and Lemma 3.11 ensure that the v_j's at Step 5 are solutions to the recombination problem (11). Since F is primitive, so are the F_j's. Since $d_\lambda(\mathrm{lc}(F)/\mathrm{lc}(F_j)) + d_\lambda(F_j) \leqslant d_\lambda(F)$ we have $\tilde{F}_j = \frac{\mathrm{lc}(F)}{\mathrm{lc}(F_j)} F_j$ so F_j is the primitive part of $\tilde{F}_j$. Hence the algorithm returns a correct answer. Since $m_\lambda(F) \leqslant d_\lambda(F) - v_\lambda(F)$ (Definition 2.22), we have $\sigma \leqslant 4V/d$ by Lemma 3.11. Hence Step 3 costs $\tilde{O}(V)$ by Theorem 2.39. Step 5 fits in the aimed bound by Proposition 3.10 and Lemma 3.11. Using technique of subproduct trees, Step 7 costs $\tilde{O}(\deg(F_j)(d_\lambda(F) - v_\lambda(F))$ by Corollary 2.13, and computing primitive parts at Step 8 has the same cost. Summing over $j = 1, \ldots, \rho$ the overall cost of Step 6 is $\tilde{O}(d(d_\lambda(F) - v_\lambda(F)) = \tilde{O}(V)$. This concludes the proof. □

Proof of Theorem 1.2. If $\lambda_F \geqslant 0$, Theorem 1.2 follows immediately from Proposition 3.12 since s is smaller or equal to the lower lattice length r of $N(F)$. Note that testing non degeneracy amounts to test squarefreeness of some univariate polynomials whose degree sum is r, hence costs only $\tilde{O}(r)$ operations in $\mathbb{K}$. If $\lambda_F < 0$, we apply algorithm `Factorization` to the reciprocal polynomial $\tilde{F}$ of F, which satisfies now $\lambda_{\tilde{F}} \geqslant 0$. We

recover the factors of F as the reciprocal factors of $\tilde{F}$. Since $\tilde{F}$ has same V, same s and same r than F, the cost remains unchanged. □

Proof of Corollary 1.5. The corollary follows straightforwardly from Theorem 1.2. However, let us explain for the sake of completeness how to compute quickly the minimal lower lattice length $r_0(F) = r_0(N(F))$. Recall from (1) that, for a lattice polygon P,

$$r_0(P) = \min\left\{r(\tau(P)) \mid \tau \in \mathrm{Aut}(\mathbb{Z}^2)\right\},$$

where $r(\tau(P))$ stands for the lattice length of the lower convex hull $\Lambda(\tau(P))$ and $\mathrm{Aut}(\mathbb{Z}^2)$ stands for the group of affine automorphisms. □

Lemma 3.13. *Let P be a lattice polygon, with edges $E_1, \dots, E_n$. Denote $w_i \in \mathbb{Z}^2$ the inward orthogonal primitive vector of E_i. There exist $\tau_i, \tau_i' \in GL_2(\mathbb{Z})$ with $\det(\tau_i) = 1$ and $\det(\tau_i') = -1$ and such that $\tau_i(w_i) = \tau_i'(w_i) = (1, 0)$. Then*

$$r_0(P) = \min\left(r(\tau_1(P)), r(\tau_1'(P)), \dots, r(\tau_n(P)), r(\tau_n'(P))\right).$$

Geometrically, the maps τ_i and τ_i' simply send E_i to a vertical left hand edge. Such maps are straightforward to compute (note that they are not unique).

Proof. Since the lower lattice length is invariant by translation, it's sufficient to look for a map $\tau \in GL_2(\mathbb{Z})$ that reaches r_0. Let us first consider $\tau \in GL_2(\mathbb{R})$. Consider the set $I_\tau = \{j \mid \tau(E_j) \subset \Lambda(\tau(P)\}$ of the indices of the lower edges of $\tau(P)$. Denoting $d_j(\tau) = \det((1, 0), \tau(w_j))$, we have

$$j \in I_\tau \iff d_j(\tau) > 0.$$

The maps $\tau \mapsto d_j(\tau)$ being continuous, we deduce that $I_\tau \subset I_{\tau'}$ for all $\tau' \in GL_2(\mathbb{R})$ close enough to τ, and with equality $I_\tau = I_{\tau'}$ if $d_j(\tau) \neq 0$ for all $j = 1, \dots, n$. Now, if $\tau, \tau' \in GL_2(\mathbb{Z})$ then $I_\tau \subset I_{\tau'}$ implies $r(\tau(P)) \leqslant r(\tau'(P))$ and equality $I_\tau = I_{\tau'}$ implies equality of the lower lattice lengths. It follows that r_0 is reached at $\tau \in GL_2(\mathbb{Z})$ such that $d_i(\tau) = 0$ for some i (such a τ exists for each i). This forces $\tau(w_i) = \pm(1, 0)$ and we may suppose $\tau(w_i) = (1, 0)$ since the lower lattice length is invariant by vertical axis symmetry. But if $\tau' \in GL_2(\mathbb{Z})$ is another map such that $\tau'(w_i) = (1, 0)$ and which satisfies moreover $\det(\tau) = \det(\tau')$, then

$$d_j(\tau') = \det\left(\tau'(w_i), \tau'(w_j)\right) = \det(\tau')\det(w_i, w_j) = \det(\tau)\det(w_i, w_j) = d_j(\tau)$$

for all $j = 1, \dots, n$, from which it follows that $I_\tau = I_{\tau'}$, hence $r(\tau(P)) = r(\tau'(P))$. The lemma follows. □

Acknowledgements

We thank the reviewers for their careful reading and for their valuable suggestions which helped to improve the presentation of this article.

References

[1] Fatima Abu Salem, Shuhong Gao, and Alan G.B. Lauder. Factoring polynomials via polytopes. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation — ISSAC'04*, pages 4–11. Association for Computing Machinery, 2004. DOI: 10.1145/1005285.1005289.

[2] Jérémy Berthomieu and Grégoire Lecerf. Reduction of bivariate polynomials from convex-dense to dense, with application to factorizations. *Mathematics of Computation*, 81(279):1799–1821, 2012. DOI: 10.1090/S0025-5718-2011-02562-7.

[3] Peter Bürgisser, Michael Clausen, and Mohammad Amin Shokrollahi. *Algebraic Complexity Theory*. Volume 315 of *Grundlehren der mathematischen Wissenschaften*. Springer, 1997. ISBN: 9783540605829.

[4] Antonio Campillo. *Algebroid Curves in Positive Characteristic*. Volume 813 of *Lecture Notes in Mathematics*. Springer, 1980. ISBN: 9783540100225.

[5] Xavier Caruso, David Roe, and Tristan Vaccon. Division and slope factorization of p-adic polynomials. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation — ISSAC'16*, pages 159–166. Association for Computing Machinery, 2016. DOI: 10.1145/2930889.293089

[6] Guillaume Chèze and André Galligo. Four lectures on polynomial absolute factorization. In *Solving Polynomial Equations: Foundations, Algorithms, and Applications*, volume 14 of *Algorithms and Computation in Mathematics*, pages 339–392. Springer, 2005. DOI: 10.1007/3-540-27357-3_9

[7] Guillaume Chèze and Grégoire Lecerf. Lifting and recombination techniques for absolute factorization. *Journal of Complexity*, 23(3):380–420, 2007. DOI: 10.1016/j.jco.2007.01.008

[8] Shuhong Gao. Factoring multivariate polynomials via partial differential equations. *Mathematics of Computation*, 72(242):801–822, 2003. DOI: 10.1090/S0025-5718-02-01428-X.

[9] Shuhong Gao and Alan G.B. Lauder. Decomposition of polytopes and polynomials. *Discrete and Computational Geometry*, 26:89–104, 2001. DOI: 10.1007/s00454-001-0024-0.

[10] Joachim von zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, 2013. ISBN: 9781139856065.

[11] Grégoire Lecerf. Sharp precision in Hensel lifting for bivariate polynomial factorization. *Mathematics of Computation*, 75(254):921–933, 2006. DOI: 10.1090/S0025-5718-06-01810-2.

[12] Grégoire Lecerf. Fast separable factorization and applications. *Applicable Algebra in Engineering, Communication and Computing*, 19(2):135–160, 2008. DOI: 10.1007/s00200-008-0062-4.

[13] Grégoire Lecerf. New recombination algorithms for bivariate polynomial factorization based on Hensel lifting. *Applicable Algebra in Engineering, Communication and Computing*, 21(2):151–176, 2010. DOI: 10.1007/s00200-010-0121-5.

[14] Harald Niederreiter. Factorization of polynomials and some linear-algebra problems over finite fields. *Linear Algebra and its Applications*, 192:301–328, 1993. DOI: 10.1016/0024-3795(93)90247-L.

[15] Adrien Poteaux and Marc Rybowicz. Improving complexity bounds for the computation of Puiseux series over finite fields. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation — ISSAC'15*, pages 299–306. Association for Computing Machinery, 2015. DOI: 10.1145/2755996.275665.

[16] Adrien Poteaux and Martin Weimann. Computing the equisingularity type of a pseudo-irreducible polynomial. *Applicable Algebra in Engineering, Communication and Computing*, 31(5):435–460, 2020. DOI: 10.1007/s00200-020-00451-x.

[17] Adrien Poteaux and Martin Weimann. Computing Puiseux series: a fast divide and conquer algorithm. *Annales Henri Lebesgue*, 4:1061–1102, 2021. DOI: 10.5802/ahl.97.

[18] Adrien Poteaux and Martin Weimann. Local polynomial factorisation: improving the Montes algorithm. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation — ISSAC'22*, pages 149–157. Association for Computing Machinery, 2022. DOI: 10.1145/3476446.3535487.

[19] Arne Storjohann. *Algorithms for matrix canonical forms*. Thesis 13922, ETH Zürich, 2000. DOI: 10.3929/ethz-a-004141007.

[20] Joris van der Hoeven, Romain Lebreton, and Éric Schost. Structured FFT and TFT: symmetric and lattice polynomials. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation — ISSAC'13*, page 355–362. Association for Computing Machinery, 2013. DOI: 10.1145/2465506.2465526.

[21] Joris van der Hoeven and Grégoire Lecerf. On the bit-complexity of sparse polynomial and series multiplication. *Journal of Symbolic Computation*, 50:227–254, 2013. DOI: 10.1016/j.jsc.2012.06.004.

[22] Martin Weimann. A lifting and recombination algorithm for rational factorization of sparse polynomials. *Journal of Complexity*, 26(6):608–628, 2010. DOI: 10.1016/j.jco.2010.06.005.

[23] Martin Weimann. Algebraic osculation and application to factorization of sparse polynomials. *Foundations of Computational Mathematics*, 12(2):173–201, 2012. DOI: 10.1007/s10208-012-9114-z

[24] Martin Weimann. Bivariate factorization using a critical fiber. *Foundations of Computational Mathematics*, 17(5):1219–1263, 2017. DOI: 10.1007/s10208-016-9318-8.

[25] Virginia Vassilevska Williams, Yinzhan Xu, Zixuan Xu, and Renfei Zhou. New bounds for matrix multiplication: from alpha to omega. In *Proceedings of the Symposium on Discrete Algorithms — SODA'24*, page 3792–3835. Society for Industrial and Applied Mathematics, 2024. DOI: 10.1137/1.9781611977912.134.