Applicable Algebra in Engineering, Communication and Computing manuscript No.
(will be inserted by the editor)

Complexity Bounds for the rational Newton-Puiseux Algorithm over Finite Fields

Adrien Poteaux · Marc Rybowicz

Received: date / Accepted: date

Abstract We carefully study the number of arithmetic operations required to compute rational Puiseux expansions of a bivariate polynomial F over a finite field. Our approach is based on the rational Newton-Puiseux algorithm introduced by D. Duval. In particular, we prove that coefficients of F may be significantly truncated and that the complexity of parts of the computation may be bounded in terms of the output size. These preliminary results lead to a more efficient version of the algorithm with a complexity upper bound that improves previously published results. This algorithm could easily be implemented in a computer algebra system; the only asymptotically "fast" subalgorithm required to stay within our bound is the FFT-based multiplication of univariate polynomials with coefficients in a finite field.

Keywords Puiseux series · Complexity · Algebraic functions

1 Introduction

1.1 Statement of the problem and motivations

Let K be a field and F be a polynomial of $K[X,Y]$ with degrees $d_X > 0$, $d_Y > 0$ and total degree d. We assume that F is squarefree and has no factor in $K[X]$; in other words, F is primitive with respect to Y. We denote by R_F the resultant of F and F_Y with respect to Y, where F_Y is the derivative of F with respect to Y. The roots of $R_F \in K[X]$ are called the *affine critical*

Adrien Poteaux
E-mail: adrien.poteaux@sophia.inria.fr

Marc Rybowicz
XLIM UMR 6172
Université de Limoges/CNRS
E-mail: marc.rybowicz@unilim.fr

points of F. The point at infinity is critical if $X = 0$ is a critical point of $F(1/X, Y)X^{d_X}$. Non critical points are called *regular points*.

Let x_0 be a critical point. If K has characteristic 0 or $p > d_Y$, each root of F, viewed as a univariate polynomial in Y, can be formally represented by a Laurent series S in $(X - x_0)^{1/e}$, called a *Puiseux series*, for some well-chosen positive integer e. This integer is the *ramification index* of S and coefficients of S belong to a finite algebraic extension of K; see Section 2.

Puiseux series are ubiquitous in the theory and practice of algebraic curves: For instance, they may be used to determine the genus of the curve $F(X, Y) = 0$ via Hurwitz formula (20), to compute integral bases of the function field $K(X)[Y]/(F)$ (23), to determine bases of $\mathcal{L}(D)$ spaces for divisors D by means of Dedekind-Weber's algorithm (18; 2), to approximate values of algebraic functions (when $K \subset \mathbb{C}$), etc.

A Puiseux series can be decomposed into two parts: the singular part, that captures important information such as ramification indices or Puiseux pairs, and the regular part; see Section 2. Let K' denote the field generated over K by the Puiseux series coefficients under consideration. Kung and Traub (29) showed that, once the singular part is known, the regular part can be computed using quadratic Newton iterations in $O(d_Y M(N))$ arithmetic operations in K', where N is the number of terms required and $M(N)$ is the number of operations in K' necessary to multiply two polynomials of degree at most N in $K'[X]$. Since their purpose is to estimate the asymptotic complexity as a function of N, they do not study the complexity of the singular part computation, which is a constant independent of N. In practice, however, the singular part may already be a bottleneck.

In this paper, we focus on the complexity of the singular part, expressed as a function of the input size (namely, d_X, d_Y and the size of the coefficients of F), but also in terms of the output size.

From now on, assume that K is a field such that there exist algorithms for factorizing elements of $K[X]$. Singular parts of Puiseux series may be computed with the classical Newton-Puiseux algorithm (see (42)) or the rational Newton-Puiseux algorithm introduced by Duval in (19). The latter allows to take into account conjugacy over K and to restrict computations to residue fields of places, thus minimizing the required algebraic extensions of K and giving useful arithmetic information. It is therefore the method of choice for computer algebra systems; implementations are available in Maple, when K is an algebraic number field, ((31), see the `algcurves[puiseux]` command due to Mark Van Hoeij (23)) and Magma ((3), command `RationalPuiseux`) for instance.

We became interested in the finite field case for the following reason: If K is an algebraic number field, both variants suffer from a dramatic coefficient swell that handicap their practical utility. Moreover they cannot be applied directly with floating point coefficients; see (36; 35; 33) for examples and details. To overcome these problems, we have introduced a symbolic-numeric method:

- Important exact information, called *polygon trees*, is first obtained by means of Duval's algorithm applied modulo a well chosen prime number p. We gave a good reduction criterion for choosing a prime p such that F and $\bar{F} = F \bmod p$ have the same polygon tree. Proofs and details are given in (36); see also the preprint (35) or (33). We had to slightly modify classical Newton polygons to deal with cancellation modulo p of non essential coefficients. To this end, we introduced *generic Newton polygons*, that we will also use in this paper, since they have other advantages; Section 3. For instance, they provide *regularity indices* directly and they are relevant herein to simplify proofs.
- Then, polygon trees are used to guide floating point Puiseux series computations. A method was sketched in (32) and significant improvements, based on Singular Value Decompositions, were introduced in (33).

The rational Newton-Puiseux algorithm over finite fields is crucial in our strategy to obtain floating point Puiseux series; it is therefore important to understand its asymptotic behaviour.

Moreover, polygon trees encode ramification indices, Puiseux pairs, intersection multiplicity of branches, and more. Complexity results for Puiseux series over finite fields, combined with our good reduction criterion and bounds for good primes p, lead to bit-complexity results for problems over algebraic number fields. We illustrate this approach in the conclusion, but we will not elaborate further on this topic in the present work.

1.2 Contributions and contents

Let L be a finite field with $[L : \mathbb{F}_p] = t_0$, where p is a prime satisfying $p > d_Y$.

In Section 2 we recall classical facts about rational Puiseux expansions and Puiseux series.

Our variant of the rational Newton-Puiseux algorithm, based on generic Newton polygons and appropriate truncations of powers of X, is described in Section 3.

Section 4 contains our contributions:

- First of all, we study truncation orders in terms of the output size (Propositions 1 and 2). We also show that truncations orders throughout the algorithm form a decreasing sequence. This justifies our variant of the algorithm; see Propositions 3 and 4.
- Then, we study the coefficient field representation. Since we need to change this representation along the course of the algorithm, we bound the number of required operations in L for this task. Again, we give results in terms of the ouput size (Proposition 5).
- The total running time is usually dominated by changes of variables; this is precisely stated in Proposition 6.
- Unlike Duval (16), who relied the D5 system (16) to avoid factorization, we have chosen to factorize characteristic polynomials because efficient

factorization algorithms exist over finite fields[1]. Moreover, we obtain more precise arithmetic information. The total cost is bounded in Proposition 7.
- In Proposition 8, we prove that the output size is at most the valuation in X of the resultant of F and its derivative in Y.
- Finally, we prove our **main result**: rational Puiseux expansions above the critical point 0 of a polynomial $F \in L[X,Y]$ may be computed with an expected number of :

$$\tilde{O}\left(d_Y^3 d_X^2 + d_Y^2 d_X t_0 \log p\right)$$

operation in L, where the notation $\tilde{O}$ hides logarithmic factors; see Theorem 3. This result was presented without proof at the ISSAC '08 conference (34) and improves those of Duval, namely $O(d_Y^6 d_X^2)$ for a monic F and $O(d_Y^8 d_X^2)$ for the general case. It is worth noting that non monic cases require some care, since a bruteforce change of variable to make polynomials monic do not yield the expected complexity.
- For expansions above *all* critical points of F, we have obtained a remarkably similar estimate, indicating that "most of the complexity" may be located above a single critical point; see Theorem 4. It also lead to the following result: Provided that $p > d_Y$, the number of operations in L necessary to **compute the genus of the algebraic curve** defined by $F(X,Y) = 0$ belongs to:

$$\tilde{O}\left(d_Y^3 d_X^2 t_0 \log p\right).$$

Surprisingly, we have not found complexity bounds for this problem in the literature.

The only asymptotically "fast"method we use is for multiplying univariate polynomials. Hence, our algorithms can be effectively implemented and our results should be applicable for reasonable size entries. We emphasize that we have attempted to parametrize our bounds in terms of subalgorithm complexity (e.g. factorization, multiplication of univariate polynomials,...) and in term of the output size. Hence, it should be reasonably easy to update our results to take into account other algorithms for subproblems or any a priori information about the output size.

1.3 Related works

Chistov (8) was first to show that Newton-Puiseux algorithm had polynomial bit-complexity, but did not provide explicit exponents.

Other methods to compute Puiseux expansions have been proposed: (17), following an idea of (11) and (37). We have explained in (36) why the Newton-Puiseux approach seems preferable. No complexity results for these methods is known.

[1] In particular, we have shown in (34; 36) that the size of L can be kept small when probabilistic versions of our symbolic-numeric method are used.

Merle and Henry (22) studied the arithmetic complexity of the resolution of the singularity at the origin defined by $F(X,Y) = 0$, assuming that F is irreducible in $K[[X,Y]]$. Using *lazy evaluation*, a programming technique that delays a calculation until the result is necessary to proceed further, they obtained an arithmetic complexity similar to Duval's, that is $O(d^8)$. Teitelbaum (40) considered a more general situation and gave a factorization-free algorithm that computes a representation for the resolution of a singularity defined by a bivariate power series. The arithmetic complexity, expressed in terms of a classical local invariant c_F called the *conductor degree* is $O(c_F^6)$. In general, c_F is not known in advance and would rather be a by-product of the resolution process. In our context, if F happens to be irreducible in $K[[X,Y]]$, c_F is smaller than the X-valuation of the discriminant of F with respect to Y (44), which in turn is bounded by $(2d_Y - 1)d_X$; this yields an upper bound for the arithmetic complexity in $O(d^{12})$.

Finally, note that Puiseux expansions may be computed efficiently in terms of the truncation order if a linear differential equation satisfied by the functions is known (12; 9; 10; 24; 25). But again, the singular part complexity has not been detailed. The computation of the differential equation may be a bottleneck since it usually has coefficients with fairly large degrees: a bound in $O(d_X d_X^4)$ was given in (14) and reduced to $O(d_X d_Y^3)$ in (4), but only when K has characteristic 0[2].

1.4 Notations

- If L is a field, $\overline{L}$ will denote an algebraic closure of L.
- If $H \in L[X,Y]$, then H_X and H_Y are the formal partial derivatives of H. We denote by $d_X(H)$ (resp. $d_Y(H)$, $d(H)$) the degree of H with respect to X (resp. Y, resp. the total degree of H). The leading coefficient of H with respect to the variable, say Y, is represented by $\mathrm{lc}_Y(H)$.
- For each positive integer e, ζ_e is a primitive e-th root of unity in $\overline{L}$. Primitive roots are chosen so that $\zeta_{ab}^b = \zeta_a$.
- v_X denotes the X-adic valuation of the fractional power series field $L((X^{1/e}))$, normalized with $v_X(X) = 1$. If $S \in L((X^{1/e}))$, we denote by $\mathrm{tc}(S)$ the trailing coefficient of S, namely $S = \mathrm{tc}(S)X^{v_X(S)} + \text{terms of higher order}$.
- If $S = \sum_k \alpha_k X^{k/e}$ is a fractional power series in $L((X^{1/e}))$ and r is a rational number, $\widetilde{S}^r$ denotes the truncated power series $\widetilde{S}^r = \sum_{k \le N} \alpha_k X^{k/e}$ where $N = \max\{k \in \mathbb{N} \mid \frac{k}{e} \le r\}$. We generalize this notation to elements of $L((X^{1/e}))[Y]$ by applying it coefficient-wise. In particular, if $H \in L[[X]][Y]$ is defined as $H = \sum_i (\sum_{k \ge 0} \alpha_{ik} X^k) Y^i$, then $\widetilde{H}^r = \sum_i (\sum_{k=0}^{\lfloor r \rfloor} \alpha_{ik} X^k) Y^i$.
- If U is a univariate polynomial, then Δ_U denotes the discriminant of U and R_U denotes the resutant of U and its derivative. If U is a multivariate polynomial, the context will always allow to identify the variable.

[2] Complications occur in characteristic $p > 0$ since the derivative of X^p is 0

2 Rational Puiseux Expansions

The purpose of this section is to recall facts about Puiseux series and rational Puiseux expansions. For short, Classical Puiseux series and rational Puiseux expansion will respectively be called CPS and RPE.

Let L be a field of characteristic $p \geq 0$ and H be a squarefree polynomial in $L[X,Y]$, primitive in Y. We assume that L *and* H *satisfy the characteristic condition*:

$$p = 0 \quad \text{or} \quad p > d_Y(H) \tag{1}$$

Up to a change of variable $X \leftarrow X + x_0$ or $X \leftarrow 1/X$, we suppose that $X = 0$ is a critical point and we will consider Puiseux series and rational Puiseux expansions above 0.

Definition 1 Let H be a polynomial in $L[X,Y]$ with $d_Y(H) > 0$. A *parametrization* $R(T)$ of H is a pair of non constant power series $R(T) = (X(T), Y(T)) \in \overline{L}((T))^2$ such that $H(X(T), Y(T)) = 0$ in $\overline{L}((T))$. The parametrization is *irreducible* if there is no integer $u > 1$ such that $R(T) \in \overline{L}((T^u))^2$. The *coefficient field* of $R(T)$ is the extension of L generated by the coefficients of $X(T)$ and $Y(T)$.

Definition 2 (Rational Puiseux expansions - RPEs)

- Assume that H is irreducible in $L[X,Y]$ ($H \neq Y$) and let $\mathcal{K} = L(X)[Y]/(H)$ be the algebraic function field defined by H. A *system of L-RPEs above 0 of H* is a set of irreducible parametrizations $\{R_i\}_{1 \leq i \leq r}$ of the form:

$$R_i(T) = (X_i(T), Y_i(T)) = \left(\gamma_i T^{e_i}, \sum_{k=n_i}^{\infty} \beta_{ik} T^k \right) \in \overline{L}((T))^2$$

  with $e_i > 0, n_i \in \mathbb{Z}$ and $\beta_{in_i} \neq 0$, and such that:

  (i) There exists a canonical one-to-one correspondence between the $\{R_i\}_{1 \leq i \leq r}$ and the places $\{\mathfrak{P}_i\}_{1 \leq i \leq r}$ of $\mathcal{K}$ dividing X; see (36; 35) or (33) for more details. Places are considered in the sense of (7).

  (ii) The coefficient field of R_i is isomorphic to the residue field k_i of $\mathfrak{P}_i$.

  If $H = Y$, then $R_1 = (T, 0)$ is the only rational Puiseux expansion[3] of H above 0 and we set $n_1 = 0$. In this case, $e_1 = 1$ and $k_1 = L$.
- Assume that H is squarefree. A *system of L-RPEs above 0 of H* is the union of systems of L-RPEs for the irreducible factors of H in $L[X,Y]$.

The integer e_i is called the *ramification index* of R_i.

This concept was introduced by Duval (18). A slightly different definition appeared in (19) and (43), that corresponds to $\overline{L}$-RPE in the above sense. In the sequel, we will identify k_i with the coefficient field of R_i. Since L and H satisfy the characteristic condition (1), a system of L-RPEs exists; it is however not unique since T may be replaced in R_i by γT for any γ in k_i.

[3] This case could have been avoided, but our symbolic-numeric method may cause an expansion to vanish modulo a prime p and we have preferred to provide a treatment for vanishing roots as well.

Definition 3 We say that R_i is *defined at* $T = 0$ if $Y_i \in \overline{L}[[T]]$. In this case, the *center* of R_i is the pair $(X_i(0), Y_i(0)) \in k_i^2$.

The following result is classical (7, Chapter 4, Section 1):

Theorem 1 *Set* $f_i = [k_i : L]$.

$$\sum_{i=1}^{r} e_i \, f_i = d_Y(H)$$

It is well-known (42) that the roots of H, viewed as a univariate polynomial in Y, may be expressed as fractional power series in X called the *Puiseux series* of H above 0:

Theorem 2 (Puiseux) *There exist positive integers* $e_1, \ldots, e_s$ *with* $\sum_{i=1}^{s} e_i = d_Y(H)$ *and* $d_Y(H)$ *distinct series:*

$$S_{ij}(X) = \sum_{k=n_i}^{\infty} \alpha_{ik} \, \zeta_{e_i}^{jk} \, X^{\frac{k}{e_i}}$$

where $1 \leq i \leq s$, $0 \leq j \leq e_i - 1$, $n_i \in \mathbb{Z}$ *and* $\alpha_{in_i} \neq 0$ *if* $S_{ij} \neq 0$, *such that:*

$$H(X, S_{ij}(X)) = 0 \quad \text{in} \quad \overline{L}((X^{1/e_i})).$$

If $S_{ij} = 0$ *is a root of* H, *we set* $n_i = 0$ *and* $e_i = 1$.

From a system of RPEs, classical Puiseux series can readily be computed with the following process:

1. R_i has exactly f_i conjugates over L, that we denote R_i^σ $(1 \leq \sigma \leq f_i)$.

$$R_i^\sigma(T) = (X_i^\sigma(T), Y_i^\sigma(T)) = \left(\gamma_i^\sigma T^{e_i}, \sum_{k=n_i}^{\infty} \beta_{ik}^\sigma T^k \right)$$

2. Each R_i^σ yields a Puiseux series $S_i = Y_i^\sigma((X/\gamma_i^\sigma)^{1/e_i})$.
3. The d_Y CPS are finally obtained using the action of $\mathbb{G}_{e_i}$ on S_i, where $\mathbb{G}_{e_i}$ is the cyclic group generated by the automorphism $X^{1/e_i} \mapsto \zeta_{e_i} X^{1/e_i}$ of $\overline{L}((X^{1/e_i}))$.

Definition 4 Define $s_i = \min\{0, n_i\}$. The *regularity index* r_{ij} *of* S_{ij} *in* H is the least integer $N \geq s_i$ such that $\widetilde{S_{ij}}^{\frac{N}{e_i}} = \widetilde{S_{uv}}^{\frac{N}{e_i}}$ implies $(u, v) = (i, j)$; $\widetilde{S_{ij}}^{\frac{r_{ij}}{e_i}}$ is called the *singular part of* S_{ij} *in* H.

Roughly speaking, the regularity index is the number of terms necessary to "separate" a CPS from all the others. Since regularity indices of all Puiseux series corresponding to the same RPE are equal, we define the *singular part* of a RPE R_i to be the pair:

$$\left(\gamma_i T^{e_i}, \sum_{k=s_i}^{r_i} \beta_{ik} T^k\right)$$

where r_i is the regularity index of any Puiseux series associated to R_i.

We insist on the dependance on H of the regularity index: Indeed, the regularity index depends not only on the root considered, but also on the the initial terms of the other roots. For instance, the regularity index of the root 0 in $H = Y(Y - X)$ (resp. $H = Y(Y - X^3)$), $H = Y(X^3Y - 1)$ is 1 (resp. 3, 0). The regularity indices of the second roots are respectively 1, 3 and -3. Observe also that the singular part of a non zero root may be null: The regularity index of the root X^2 in $H = (Y - X^2)(Y - X)$ is 1 and the corresponding singular part is $0 + 0X = 0$. But the regularity index of the very same root in $H = (Y - X^2)(Y - X^3)$ is 2, yielding the singular part X^2. Finally, consider the example $H = (Y - 1 - X)(X^2Y - 1)$. The regularity index of $1 + X$ is 0 and the regularity index of $1/X^2$ is -2; the singular parts are 1 and $1/X^2$. The above examples justify the introduction of the quantity s_i, that will also enter our complexity estimates.

Our goal is to compute singular parts of RPEs, since higher order terms of the series may be computed by means of asymptotically fast methods (29).

3 Rational Newton-Puiseux algorithm

In order to compute singular part of RPEs of H above 0, we present a modified version of Duval's rational Newton-Puiseux algorithm. In our approach, classical Newton polygons are replaced with *generic Newton polygons*; see (36; 35) or (33). Generic polygons guarantee that polygon trees obtained with modular computations are the same as polygon trees that would have been computed in characteristic 0. Unlike classical Newton polygons, they also allow to compute exactly regularity indices, even when one of the expansion is null.

Assume that $H = \sum_{i,j} a_{ij} X^j Y^i$ satisfies the same hypotheses as in the previous section; in particular, $H(0, Y) \neq 0$.

Definition 5 For each pair (i, j) of $\mathrm{Supp}(H) = \{(i, j) \in \mathbb{N}^2 \,|\, a_{ij} \neq 0\}$, define $Q_{ij} = \{(i', j') \in \mathbb{R}^2 \mid i' \geq i \text{ and } j' \geq j\}$. Then the *Newton Polygon* $\mathcal{N}(H)$ of H is the set of finite edges of the convex hull $\mathcal{H}$ of $Q(H) = \cup_{(i,j)\in \mathrm{Supp}(H)} Q_{ij}$. The *generic Newton polygon* $\mathcal{GN}(H)$ is obtained by restricting $\mathcal{N}(H)$ to edges with slope no less than -1 and by joining the leftmost remaining point to the vertical axis with an edge of slope -1. Namely, we add a fictitious point $(0, j_0)$ to $\mathrm{Supp}(H)$ so as to mask edges with slope less than -1.

The algorithm first stage requires a special treatment and we introduce *exceptional polygons* for this purpose.

Definition 6 The exceptional Newton polygon $\mathcal{EN}(H)$ is the lower part of the convex hull of $\mathrm{Supp}(H) \cup \{(0,0)\}$. In other words, it consists of the edge $[(0,0),(\mathcal{I}(H),0)]$, followed by a sequence of edges with positive slopes that join $(\mathcal{I}(H),0)$ to $(d_Y(H), v_X(\mathrm{lc}_Y(H)))$. In particular, $\mathcal{EN}(H) = [(0,0),(d_Y(H),0)]$ if H is monic.

Details, examples and figures are given in (36; 35).

Each edge Δ of $\mathcal{GN}(H)$ (resp. $\mathcal{N}(H)$, $\mathcal{EN}(H)$) corresponds to three integers q, m and l with $q > 0$, q and m coprime, such that Δ is on the line $q\,j + m\,i = l$. If Δ is the horizontal edge of $\mathcal{EN}(H)$, $m = l = 0$ and we choose $q = 1$.

Definition 7 We define the *characteristic polynomial* ϕ_Δ:

$$\phi_\Delta(T) = \sum_{(i,j)\in\Delta} a_{ij} T^{\frac{i-i_0}{q}}$$

where i_0 is the smallest value of i such that (i,j) belongs to Δ.

The rational Newton-Puiseux algorithm below performs successive changes of variable, determined by (q,m,l) and the roots of ϕ_Δ. It returns a set of triplets $\{(G_i(X,Y), P_i(X), Q_i(X,Y))\}_i$ such that:

- $G_i \in \overline{L}[X,Y]$,
- $P_i(X)$ is a monomial of the form $\lambda_i X^{e_i}$,
- $Q_i(X,Y) = Q_{i0}(X) + c_i Y\, X^{r_i}$, where r_i is the regularity index of the expansion, $(P_i(T), Q_{i0}(T))$ is the singular part of a parametrization of F and $c_i \in k_i$.
- There exist nonnegative integers L_i such that $G_i(X,Y) = F(P_i(X), Q_i(X,Y))/X^{L_i}$, $G_i(0,0) = 0$ and $G_{iY}(0,0) \neq 0$.

By the formal Implicit Function Theorem, the latter conditions ensure that there exists a unique power series S such that $G_i(X, S(X)) = 0$ and $S(0) = 0$. The corresponding parametrization of F is therefore $R_i(T) = (P_i(T), Q_i(T, S(T)))$. Hence, we will consider that such a triplet represents a Puiseux series or a RPE.

We first give specifications for two sub-algorithms:

`Factor`(L,ϕ)

`Input:`

L: *a field*

ϕ: *a univariate polynomial in* $L[T]$.

`Output:`

A set $\{(\phi_i, M_i)\}_i$ *so that* ϕ_i *is irreducible in* $L[T]$ *and* $\phi = c\prod_i \phi_i^{M_i}$, *with* $c \in L$.

`Bézout`(q,m)

`Input:`

q: *a positive integer*

m: *an integer*

`Output:`

A pair of integers (a, b) *so that* $a\,q - b\,m = 1$ *and* $0 \leq b < q$.

The rational Newton-Puiseux algorithm reads as follow. The algorithm is presented in a recursive setting and we suppose that a mechanism to distinguish initial calls from recursive calls exists; for instance, an additional parameter could be used. Note that the main transformation is performed modulo a power of X. We will discuss in Section 4 how this parameter should initially be chosen and updated.

`RNPuiseux(`L`,`H`,`N`)`

`Input:`

L: A field.

H: A squarefree element of $L[X, Y]$, with $d_Y(H) \geq 1$ and $H(0, Y) \neq 0$.

N: a positive integer (truncation order).

`Output:`

A set of triplets $\{[G_i, P_i, Q_i]\}_i$, which form a set of representatives for:

- *L-RPEs of H above 0 for the initial call,*
- *L-RPEs of H centered at $(0, 0)$ for recursive calls.*

```
Begin
  If in a recursive call then
    𝒩 ← 𝒢𝒩(H)
    If 𝓘(H) = 1 then Return {[H, X, Y]} End
    H̃ ← H
  else
    𝒩 ← ℰ𝒩(H)
    v ← v_X(lc_Y(H))
    H̃ ← H modulo X^{N+v+1}
  End
  ℛ ← {}
  For each side Δ of 𝒩 do
    Compute q, m, l and φ_Δ
    (u, v) ← Bézout(q, m)
    For each (f, k) in Factor(L, φ_Δ) do
      ξ ← Any root of f
      N̂ ← N/[L(ξ) : L]
      (a, b) ← Bézout(q, m)
      Ĥ(X, Y) ← H̃(ξ^b X^q, X^m(ξ^a + Y))/X^l modulo X^{N̂+1}
      For each [G, P, Q] in RNPuiseux(L(ξ), Ĥ, N̂) do
        ℛ ← ℛ ∪ {[G, ξ^b P^q, P^m(ξ^a + Q)]}
      End
    End
  End
  Return ℛ
End.
```

Example 1 Consider $F \in \mathbb{F}_{13}[X,Y]$:

$$F(X,Y) = \left(16\,X^3 - Y^2 + 2\,Y - 1\right)\left(-2\,X^2 + Y^2 - 2\,Y + 1\right)\left(XY^3 - 2\right).$$

The output of the algorithm is:

$$\begin{aligned}(P_1, Q_1) &= (9\,X^2,\ 1 + X^2(0 + 9\,X(1+Y)))\\ (P_2, Q_2) &= (X,\ 1 + X\left(\sqrt{2} + Y\right))\\ (P_3, Q_3) &= (2\,X^3,\ \frac{1}{X}(1+Y))\end{aligned}$$

where $\sqrt{2}$ stands for any root of $T^2 - 2$. Null coefficients corresponds to fictitious edges of generic polygons. Ramifications indices are respectively 2, 1 and 3. Regularity indices are respectively 3, 1 and -1.

Polygons trees keep track of combinatorial information encountered in the execution of the algorithm: Namely, generic Newton polygons and multiplicity structures for the roots or characteristic polynomials. We refer the reader to (34, Section 3.3) or (33, Section 2.1.5) for a formal definition and to (32; 33) for methods to guide floating point computation.

4 Arithmetic complexity of `RNPuiseux` over finite fields

In this section, L denotes a finite field and $F = \sum_{i=0}^{d_Y} A_i(X)Y^i$ is a squarefree polynomial in $L[X,Y]$, primitive with respect to Y, with degrees $d_Y = d_Y(F)$, $d_X = d_X(F)$ and $d = d(F)$. We denote by $p > d_Y$ the characteristic of L and define $t_0 = [L : \mathbb{F}_p]$. The section is devoted to the proof of the following theorems:

Theorem 3 *Assuming that FFT-based polynomial multiplication over finite fields is used, the* `RNPuiseux` *algorithm can compute the singular parts of a system of RPEs above* 0 *of* F *with an expected number of* $\tilde{O}(d_Y^3 d_X^2 + d_Y^2 d_X t_0 \log p)$ *field operations in* L.

As usual, the notation $\tilde{O}$ hides logarithmic factors. This result was presented in (34) without proof and only when F is a monic polynomial in Y.

Randomization is necessary for factorization steps and for the computation of primitive elements; see Section 4.2 and 4.4. Therefore, we give bounds for average numbers of operations.

This result improves those of (19), who used the D5 system to avoid factorization. She obtained $O(d_Y^6 d_X^2)$ operations in L for the monic case, and $O(d_Y^8 d_X^2)$ in general. If the cardinal of L is large, it may be preferable to use the D5 approach in order to remove the dependence on q and t_0; see (15) for complexity results.

To deal with *all* conjugacy classes over L of affine critical points, we proceed as follow: Let R_F be the resultant of F and F_Y with respect to Y. If $R_F =$

$\prod_i \Phi_i^{h_i}$ is a factorization of R_F into a product of irreducible polynomials $\Phi_i \in L[X]$, RPEs above roots of Φ_i are conjugated over L. Therefore, it is sufficient to compute a system of RPEs above one root c_i of Φ_i for each i. We obtain a remarkably similar result:

Theorem 4 *Assuming that FFT-based polynomial multiplication over finite fields is used, the* `RNPuiseux` *algorithm can compute the singular parts of systems of RPEs above all conjugacy classes over L of critical points of F, including the point at infinity, with an expected number of $\tilde{O}(d_Y^3 d_X^2 t_0 \log p)$ field operations in L.*

We first introduce notations and make some assumptions:

- $\{R_i\}_{1\leq i\leq \rho}$ with $R_i(T) = (X_i(T), Y_i(T))$ stands for singular parts of a system of RPEs above 0, with coefficient fields k_i.
- (G_i, P_i, Q_i) is the output of `RNPuiseux` corresponding to R_i.
- $(n_i, s_i, r_i, e_i, f_i)$, $1 \leq i \leq \rho$ are the integers associated with R_i in Section 2.
- The $e_i f_i$ CPS associated with R_i are denoted $S_{ijk}(X)$, $0 \leq k \leq e_i - 1, 1 \leq j \leq f_i$, and Y_{ijk} represents the singular part of S_{ijk}.
- The following quantities enter our estimates:

$$\delta_F = \sum_{i=1}^{\rho} (r_i - s_i) f_i \qquad \eta_F = \sum_{i=1}^{\rho} (r_i - s_i + 1) f_i$$

In particular, if F is monic, all the s_i are null and the definition of δ_F is identical to that of (34; 35; 33). This modification allows to deal with the non monic case. Considering Theorem 1, remark that:

$$\delta_F \leq \eta_F = \delta_F + \sum_{i=1}^{\rho} f_i \leq \delta_F + d_Y.$$

Note also, that unlike δ_F, η_F cannot be null.

Remark 1 The integer η_F is exactly the number of elements of L necessary to represent the Y_i. If $n_i \geq 0$, assume that a vector of coefficients in k_i, indexed from 0 to r_i, is used to represent the singular part of $Y_i(T)$. Such a representation is legitimate since generic Newton polygon may introduce null coefficients. If $n_i < 0$, then Y_i may be represented by a vector of $r_i - n_i + 1$ coefficient in k_i. Since the elements of k_i may in turn be represented by vectors of f_i elements of L, the total size of `RNPuiseux` output is η_F.

We split the proof of theorems 3 and 4 into several results.

4.1 Truncating powers of X

Lemma 1 *Define $v = v_X(A_{d_Y})$. Then, $v = -\sum_{i=1}^{\rho} s_i f_i$.*

Proof The minimal polynomial of S_{ijk} over $\overline{L}((X))$ is $M_{ij} = \prod_{k=0}^{e_i-1}(Y - S_{ijk})$. The content of M_{ij} with respect to Y is X^{s_i}. Setting $M = \prod_{ij} M_{ij}$, it is easily seen that M and $X^{-v}F$ are equal, up to a factor that is a unit in $\overline{L}[[X]]$. By Gauss Lemma, the content of M is $X^{\sum_{i=1}^{\rho} s_i f_i}$, while the content of $X^{-v}F$ is X^{-v}.

Proposition 1 *RPEs of $\widetilde{F}^{\delta_F+v}$ and F above 0 have the same singular parts.*

Proof Since F and $\widetilde{F}^{\delta_F+v}$ have the same degree in Y, they have the same number of CPSs above 0; those are denoted by $\{U_m\}_{1\leq m\leq d_Y}$. We order the R_i so that:

$$\frac{r_1 - s_1}{e_1} \leq \frac{r_2 - s_2}{e_2} \leq \cdots \leq \frac{r_\rho - s_\rho}{e_\rho}. \tag{2}$$

It is sufficient to show that for each triplet (i, j, k) there exists an integer $l(i, j, k)$ such that $v_X(S_{ijk} - U_{l(i,j,k)}) > r_i/e_i$. In other words, $U_{l(i,j,k)} = Y_{ijk} +$ *higher order terms*. This will prove in particular that $\widetilde{F}^{\delta_F+v}$ is squarefree and that l defines a one-to-one map.

Assume that Proposition 1 is wrong. Let i_0, $1 \leq i_0 \leq \rho$, be the smallest integer such that there exist j_0 and k_0 satisfying: For each m, $v_X(S_{i_0j_0k_0} - U_m) \leq r_{i_0}/e_{i_0}$.

Set $m_0 = \sum_{i=1}^{i_0-1} e_i f_i$. For $i < i_0$ and all relevant j and k, there exists pairwise distinct $l(i, j, k)$ with $v_X(S_{ijk} - U_{l(i,j,k)}) > r_i/e_i$; we assume that the U_m are ordered so that $l(i, j, k) < m_0$.

Write $F = \widetilde{F}^{\delta_F+v} + X^{\delta_F+v+1}V$, where $V \in L[X, Y]$. Computing valuations in $\widetilde{F}^{\delta_F+v}(X, S_{i_0j_0k_0}) = -X^{\delta_F+v+1}V(X, S_{i_0j_0k_0})$, we obtain:

$$v + \delta_F + 1 + v_X(V(X, S_{i_0j_0k_0})) = v + \sum_{m=1}^{d_Y} v_X(S_{i_0j_0k_0} - U_m). \tag{3}$$

Since $v_X(V(X, S_{i_0j_0k_0})) \geq d_Y \frac{s_{i_0}}{e_{i_0}}$, using the definition of i_0, we get:

$$d_Y \frac{s_{i_0}}{e_{i_0}} + \delta_F + 1 \leq \sum_{m=1}^{m_0} v_X(S_{i_0j_0k_0} - U_m) + (d_Y - m_0)\frac{r_{i_0}}{e_{i_0}}.$$

Assumption (2) implies $\sum_{i=i_0}^{\rho}(r_i - s_i)f_i \geq (d_Y - m_0)(r_{i_0} - s_{i_0})/e_{i_0}$. Hence:

$$m_0 \frac{s_{i_0}}{e_{i_0}} + \sum_{i=1}^{i_0-1}(r_i - s_i)f_i + 1 \leq \sum_{m=1}^{m_0} v_X(S_{i_0j_0k_0} - U_m) \tag{4}$$

Next, we define $I_<$ (resp. $I_>$, $I_=$) to be the subset of integers in $[1, i_0 - 1]$ such that $\frac{s_i}{e_i} < \frac{s_{i_0}}{e_{i_0}}$ (resp. $\frac{s_i}{e_i} > \frac{s_{i_0}}{e_{i_0}}$, $\frac{s_i}{e_i} = \frac{s_{i_0}}{e_{i_0}}$). We note that, if $i \in I_=$, then $v_X(S_{i_0j_0k_0} - U_{l(i,j,k)}) \leq r_i/e_i$; otherwise, $S_{i_0j_0k_0}$ and S_{ijk} would coincide up to an order greater than r_i/e_i, contradicting the definition of r_i. Splitting the sum over m in inequality (4) and using this remark, we obtain:

$$m_0 \frac{s_{i_0}}{e_{i_0}} + \sum_{i=1}^{i_0-1}(r_i - s_i)f_i + 1 \leq \sum_{i\in I_<} s_i f_i + \frac{s_{i_0}}{e_{i_0}} \sum_{i\in I_>} e_i f_i + \sum_{i\in I_=} r_i f_i.$$

Splitting $m_0 = \sum_{i\in I_<} e_i f_i + \sum_{i\in I_>} e_i f_i + \sum_{i\in I_=} e_i f_i$ gives:

$$\sum_{i=1}^{i_0-1}(r_i - s_i)f_i + 1 \leq \sum_{i\in I_<}\left(\frac{s_i}{e_i} - \frac{s_{i_0}}{e_{i_0}}\right)e_i f_i + \sum_{i\in I_=}\left(\frac{r_i}{e_i} - \frac{s_{i_0}}{e_{i_0}}\right)e_i f_i.$$

But in the right hand side, the first sum is negative and $s_i/e_i = s_{i_0}/e_{i_0}$ in the second sum. Hence:

$$\sum_{i=1}^{i_0-1}(r_i - s_i)f_i + 1 \leq \sum_{i\in I_=}(r_i - s_i)f_i.$$

This is clearly impossible since all terms are non negative and $I_= \subset [1, i_0 - 1]$; the proof of Proposition 1 is complete.

The initial term of a RPE centered at $(0,0)$ corresponds to an edge with negative slope of the classical Newton polygon of F. We assume for a moment that the $\{R_i\}_{1\leq i\leq \rho'}$, $1 \leq \rho' \leq \rho$, are exactly the RPEs of F centered at $(0,0)$ and we introduce the following quantity:

$$\theta_F = \sum_{i=1}^{\rho'}(r_i - s_i)f_i = \sum_{i=1}^{\rho'} r_i f_i.$$

We recall that S_{ijk} vanishes at $X = 0$ if $1 \leq i \leq \rho'$.

Proposition 2 *RPEs centered at* $(0,0)$ *of* $\widetilde{F}^{\theta_F}$ *and* F *have the same singular parts.*

Proof Although this result is similar to Proposition 1, we have not found a reduction to Proposition 1, or vice versa.

Assume $\theta_F > 0$ (otherwise, there is nothing to prove) and define $d'_Y = d_Y(\widetilde{F}^{\theta_F}) \leq d_Y$ and $v' = v_X(\mathrm{lc}_Y(\widetilde{F}^{\theta_F}))$. Define:

$$d_0 = v_Y(F(0,Y)) = v_Y(\widetilde{F}^{\theta_F}(0,Y)) = \sum_{i=1}^{\rho'} e_i f_i;$$

d_0 is positive and represents the number of CPSs of F and $\widetilde{F}^{\theta_F}$ vanishing at $X = 0$. Denote by $\{U_m\}_{1\leq m\leq d'_Y}$ the Puiseux series of $\widetilde{F}^{\theta_F}$ and suppose that U_m vanishes at $X = 0$ for $1 \leq m \leq d_0$.

If, for each triplet (i,j,k) with $1 \leq i \leq \rho'$, there exists an integer $l(i,j,k)$ with $1 \leq l(i,j,k) \leq d_0$ and $v_x(S_{ijk} - U_{l(i,j,k)}) > r_i/e_i$, we are done. Suppose that i_0 is the smallest integer such that there exist j_0 and k_0 satisfying: $v_x(S_{i_0 j_0 k_0} - U_m) \leq r_{i_0}/e_{i_0}$ for all m, $1 \leq m \leq d_0$. Set $m_0 = \sum_{i=1}^{i_0-1} e_i f_i$. For $i < i_0$ and all relevant j and k, there exists pairwise distinct $l(i,j,k)$ with $v_X(S_{ijk} - U_{l(i,j,k)}) > r_i/e_i$; we assume that the U_m are ordered so that $l(i,j,k) \leq m_0$ for $i < i_0$.

Evaluating the expression $F = \widetilde{F}^{\theta_F} + X^{\theta_F+1}V$ at $Y = S_{i_0j_0k_0}$, we now obtain:

$$\theta_F + 1 + v_X(V(X, S_{i_0j_0k_0})) = v' + \sum_{m=1}^{d'_Y} v_X(S_{i_0j_0k_0} - U_m).$$

For $m > d_0$, $v_X(S_{i_0j_0k_0} - U_m) = v_X(U_m) \leq 0$ and Lemma 1 applied to $\widetilde{F}^{\theta}$ gives $v' = -\sum_{m=d_0+1}^{d'_Y} v_X(U_m)$. Since $v_X(V(X, S_{i_0j_0k_0})) \geq 0$, we get:

$$\theta_F + 1 \leq \sum_{m=1}^{d_0} v_X(S_{i_0j_0k_0} - U_m)$$

We have $v_X(S_{i_0j_0k_0} - U_m) \leq r_i/e_i$ if $m = l(i,j,k) \leq m_0$ for some $i < i_0$ and $v_X(S_{i_0j_0k_0} - U_m) \leq r_{i_0}/e_{i_0}$ for $m > m_0$. We deduce:

$$\theta_F + 1 \leq \sum_{i=1}^{i_0-1} r_i f_i + (d_0 - m_0)\frac{r_{i_0}}{e_{i_0}} \Longleftrightarrow \sum_{i=i_0}^{\rho'} r_i f_i + 1 \leq (d_0 - m_0)\frac{r_{i_0}}{e_{i_0}}.$$

The latter inequality is impossible if the positive rational numbers $\{r_i/e_i\}_{1\leq i\leq \rho'}$ form a non decreasing sequence; Proposition 2 is proved.

Consider the following examples:

- Set $H = X^v \prod_{ijk}(Y - Y_{ijk})$ and suppose that the coefficients of X^{r_i/e_i} are non zero for all i.
  - If $v = 0$, it is easily seen that $d_X(H) = \theta_H$. If $\theta_H > 0$, then $\widetilde{H}^{\theta_H - 1}$ has a Puiseux series equal to 0: The truncation order θ_H is optimal for H.
  - If $v > 0$, the truncation order $\delta_H + v$ is not equal to $d_X(H)$. For general F, define $\gamma_F = \sum_{i=1}^{\rho}(\max\{r_i, 0\} - \min\{n_i, 0\})f_i$. It can be shown that $d_X(H) = \gamma_F \leq \delta_F + v$. Truncating at order γ_F is not always sufficient, as demonstrated by the following example:

    $$F = (X^3 + 9\,X^4)\,Y^6 - 3\,Y^4X^2 + (3\,X - 6\,X^2)\,Y^2 - X^2 - 2\,X - 1.$$

    The singular part of the unique RPE is $(T^6, \frac{1}{T^3} - \frac{1}{2}\frac{1}{T})$ and $\gamma_F = 0 - (-3) = 3$. But the RPE of $\widetilde{F}^3$ is precisely $(T^6, \frac{1}{T^3} + \frac{1}{T})$.
- If $F_m = X^{2m}Y^2 - 1$, the two Puiseux series are $\pm 1/X^m$. This time, $\delta_{F_m} = 2(-m + m) = 0$, but $\delta_{F_m} + v = 2\,m$. The bound $\delta_{F_m} + v$ is optimal for this case.
- Define $F_m = (Y - X^m)(X^mY - 1)$. We have $\theta_{F_m} = 0$; $\widetilde{F_m}^0 = Y$ gives the correct singular part for the positive order expansion, namely, 0. But to get both singular parts, order $\delta_{F_m} + v = 0 + (-m - (-m)) + m = m$ is required.

Although bounds of Proposition 1 and 2 are attained for families of examples, they are usually not optimal.

For the next step, we introduce more notations. If H is a polynomial as specified in `RNPuiseux`, let Δ be an edge of the Newton polygon, ξ be a root of ϕ_Δ with multiplicity $M_{\Delta,\xi}$ and $(m_\Delta, q_\Delta, l_\Delta, a_\Delta, b_\Delta)$ be the integers associated with Δ. The main transformation in the rational Newton-Puiseux algorithm is the computation of $H_{\Delta,\xi}$:

$$H_{\Delta,\xi} = \frac{(\xi^{b_\Delta} X^{q_\Delta}, X^{m_\Delta}(\xi^{a_\Delta} + Y))}{X^{l_\Delta}}.$$

Proposition 3

$$\theta_H = \sum_{(\Delta,\xi)\in \mathcal{GN}(H)} [L(\xi) : L] \left(\theta_{H_{\Delta,\xi}} + m_\Delta M_{\Delta,\xi}\right) \tag{5}$$

$$\delta_H = \sum_{(\Delta,\xi)\in \mathcal{EN}(H)} [L(\xi) : L]\, \theta_{H_{\Delta,\xi}} \tag{6}$$

Proof Let R_i be a RPE of H and $\widehat{R}_i$ be the corresponding RPE of $H_{\Delta,\xi}$. If r_i and e_i (resp. $\hat{r}_i$ and $\hat{e}_i$) are the regularity and ramification indices of R_i (resp. $\widehat{R}_i$), we have: $\hat{r}_i = r_i - m_\Delta/q_\Delta e_i = r_i - m_\Delta \hat{e}_i$. Moreover, if f_i (resp. $\hat{f}_i$) is the the degree of the residue field of R_i (resp. $\widehat{R}_i$), then $f_i = [L(\xi) : L]\hat{f}_i$ because the ground field of $H_{\Delta,\xi}$ is $L(\xi)$. Denote by $E_{\Delta,\xi}$ the set of indices i such that R_i corresponds to (Δ, ξ). Then:

$$\begin{aligned}
\theta_H &= \sum_{(\Delta,\xi)\in \mathcal{GN}(H)} \sum_{i\,\in E_{\Delta,\xi}} r_i f_i \\
&= \sum_{(\Delta,\xi)\in \mathcal{GN}(H)} \sum_{i\,\in E_{\Delta,\xi}} (\hat{r}_i + m_\Delta \hat{e}_i)\hat{f}_i [L(\xi) : L] \\
&= \sum_{(\Delta,\xi)\in \mathcal{GN}(H)} [L(\xi) : L] \left(\sum_{i\in E_{\Delta,\xi}} \hat{r}_i \hat{f}_i + m_\Delta \sum_{i\in E_{\Delta,\xi}} \hat{e}_i \hat{f}_i \right)
\end{aligned}$$

The latter sum represents the number of roots of $H_{\Delta,\xi}$ that vanish at $X = 0$, and this is precisely the multiplicity of ξ as a root of Φ_Δ; see (6) Section 8.3 or (42) for instance. This gives equality (5). For the second equality, the same argument applied to the exceptional Newton polygon gives:

$$\sum_{i=1}^{\rho} r_i f_i = \sum_{(\Delta,\xi)\in \mathcal{EN}(H)} [L(\xi) : L] \left(\theta_{H_{\Delta,\xi}} + m_\Delta M_{\Delta,\xi}\right).$$

For $\Delta \in \mathcal{EN}(H)$, we always have $m_\Delta \leq 0$. From $\frac{m_\Delta}{q_\Delta} = \frac{s_i}{e_i}$, we get:

$$\sum_{(\Delta,\xi)\in \mathcal{EN}(H)} [L(\xi) : L] m_\Delta M_{\Delta,\xi} = \sum_{(\Delta,\xi)\in \mathcal{EN}(H)} \frac{m_\Delta}{q_\Delta} \sum_{i\in E_{\Delta,\xi}} e_i f_i = \sum_{i=1}^{\rho} s_i f_i.$$

Proposition 3 is proved.

The main result of this section is now:

Proposition 4 *The function call* `RNPuiseux`*$(L,\ F,\ \delta_F)$ returns a system of L-RPEs of F above 0.*

Proof We consider a single expansion R and drop indices to simplify notations, so that (G, P, Q) are the associated quantities. $F = H_1, H_2, \ldots, H_{h+1} = G$ be the sequence of input polynomials in the corresponding branch of the function call tree. Propositions 1 and 2 show that it is sufficient to truncate F modulo X^{δ_F+v+1} and to compute H_i modulo $X^{\theta_{H_i}+1}$ for $i > 1$. But recurrence relation coefficients in Proposition 3 are positive; therefore, $\delta_F/[L(\xi_1, \ldots, \xi_i) : L] \geq \theta_{H_{i+1}}$ for $1 \leq i \leq h$, where the ξ_i are the succesive roots of the characteristic polynomials. Hence, `RNPuiseux` returns the expected output.

In practice, δ_F is usually unknown in advance and we will give bounds in Section 4.5.

Remark 2 Proposition 3 allows to truncate further powers of X. Consider the following example: Assume that L is a field that contains no square roots of 2 and let F be the irreducible polynomial of $L[X, Y]$ corresponding to the RPE $(T^6, T^2 + \sqrt{2}T^4 + T^9)$, with residue field of degree 2 over L. We skip the first step, because the exceptional polygon has a unique slope with a unique root, namely, 0. Since F is monic, $\delta_F + v = \theta_{H_1}$. Each polygon has a unique slope that we call Δ_i. Starting from $\theta_{H_1} = 2 \times 9 + 0 = 18$ and applying relation (5), we obtain $\theta_{H_2} = \theta_{H_1} - M_{\Delta_1,1} = 18 - 4 = 14$, $\theta_{H_3} = \theta_{H_2}/[L(\sqrt{2}) : L] - M_{\Delta_2,\sqrt{2}} = 14/2 - 2 = 5$ (it can be shown that the optimal truncation orders are 7, 9 and 5; see remark 3). Hence, if there is a unique RPE, the θ_{H_i}, $i > 1$, can be deduced from $\delta_F + v$. If there are several RPEs, information obtained from recursive calls for some branches can also be used to reduce truncation orders for other branches.

Remark 3 The proof of Proposition 4 given in (33; 35) is significantly different: Therein, a formula for the optimal truncation order is determined and it is shown that it is bounded by δ_F (F is assumed monic). Note that the proof of Lemma 16 of (33) is wrong: It is implicitely assumed that the sequence of optimal truncation orders is decreasing, but this is false. However, the final result is correct because one can show that the optimal truncation orders are bounded by terms of the decreasing sequence $\delta_F + v \geq \theta_{H_2} \geq \cdots \geq \theta_{H_h}$, as above.

To conclude this section, we show that there are families of examples such that the sequence of truncation orders given by Proposition 1 and 2 is optimal: This occurs when the ramification is introduced at the last step of the computation. Consider the irreducible polynomial F that vanishes at $(T^6, 1 + T^6 + T^{12} + T^{13})$. There is a unique RPE above 0 and $\mathcal{EN}(F)$ has a unique horizontal slope. The unique root of the characteristic polynomial is 1. Therefore, $\delta_F + v = \theta_{H_1} = \theta_{H_2} = 13$, $\theta_{H_3} = 7$, $\theta_{H_4} = 1$. Truncating the H_i at orders less than 13, 7 or 1 yields an incorrect output: Indeed, $\widetilde{F}^{12}$, $\widetilde{H_2}^{12}$, $\widetilde{H_3}^{6}$ and $\widetilde{H_4}^{0}$ are not even squarefree.

4.2 Representation of residue fields and arithmetic in finite fields

`RNPuiseux` constructs residue fields k_i step by step by adding characteristic polynomial roots to the ground field. It would therefore be convenient to represent them as multiple extensions. More precisely:

$$k_i \simeq L[T_1, \ldots, T_{m_i}]/(M_1(T_1), M_2(T_1, T_2), \ldots, M_{m_i}(T_1, \ldots, T_{m_i})),$$

for some integer m_i, where M_j is the minimal polynomial (with $\deg_{T_j}(M_j) \geq 2$) of a root over the previous coefficient field:

$$k_{i,j-1} \simeq L[T_1, \ldots, T_{j-1}]/(M_1(T_1), \ldots, M_{j-1}(T_1, \ldots, T_{j-1})),$$

with $k_{i,0} = L$ and $k_{i,m_i} = k_i$.

With such a triangular representation, it is shown in (30) that the number of operations in L necessary to perform an operation in k_i is bounded by $C4^{m_i} f_i \log^3(f_i)$, for some universal constant C. Since $m_i \leq \log_2 f_i$, we obtain $C f_i^3 \log^3(f_i)$; with such a bound, we could not reach our goal, namely Theorem 3. Moreover, taming the "exponential" factor 4^{m_i} seems to be difficult (see (30)) although recent results (28) may prove useful (Eric Schost, personal communication).

Therefore, we turn to *primitive representations* for the fields $k_{i,j}$: $k_{i,j} \simeq L[T]/(P_j(T))$, where P_j is the minimal polynomial over L of a primitive element α_j, i.e. $k_{i,j} \simeq L(\alpha_j)$ for $1 \leq j \leq m_i$. The elements of $k_{i,j}$ are encoded as polynomials of degree less than $[k_{i,j} : L]$ with coefficients in L. Whenever the root $\xi = \xi_j$ of a characteristic polynomial is not in the coefficient field, the following computations are required:

($\mathcal{A}$) Compute a primitive element α_j of $k_{i,j} = L(\xi_j, \alpha_{j-1})$ and its minimal polynomial P_j over L.
($\mathcal{B}$) Express ξ_j and α_{j-1} as polynomials in α_j of degree less than $[k_{i,j} : L]$ with coefficients in L.
($\mathcal{C}$) Rewrite the coefficients of $H(X, Y) \in L(\alpha_{j-1})[X, Y]$ in terms of α_j.

There are obviously at most $\log_2(f_i)$ such transformations to perform.

Moreover, when a recursive call to `RNPuiseux` return, note that the representation of ξ_j is different from the coefficient representation of the returned triplet $[G, P, Q]$. In order to form a meaningful result $[G, \xi_j^b P^q, P^m(\xi_j^a + Q)]$ we still need to:

($\mathcal{D}$) Rewrite ξ_j^a and ξ_j^b in terms of α_{m_i}.

These "backward" transformations must be executed each times the function returns.

Since changes of representation are not required if multiple extension are used, for the sake of simplicity, we have not included them in our description of `RNPuiseux`.

Let us introduce some notations and recall a few facts:

- L_t denotes a *simple algebraic extension* of degree t of L, as above.
- By an L-operation, we mean a field operation in L: addition, multiplication or division.
- For a sufficiently large integer N, we define $\mathcal{L}(N) = \log N \log\log N$. In the sequel, log always stands for a logarithm with base greater than 1.
- $\mathcal{M}(N)$ is a bound for the number of field operations needed to compute the product of two polynomials of degree no larger than N with coefficients in a finite field. We recall that we can choose $\mathcal{M}(N) = C_0 N^2$ for classical arithmetic and $\mathcal{M}(N) = C_1 N \mathcal{L}(N) \in O\tilde{}(N)$ if FFT-based multiplication is used, where C_0 and C_1 are constants; see (21, Corollary 8.22), for instance. We assume $\mathcal{M}(N) \geq N$ and the following property. If a and b are positive integers, then there exists a constant C such that:

$$\mathcal{M}(a)\mathcal{M}(b) \leq C\mathcal{M}(ab)\log(ab)\,\mathcal{L}(ab). \tag{7}$$

  This property is verified by the two functions above: since $\log a \log b \leq \frac{1}{2}\log^2 ab$, we deduce $(\log\log a)(\log\log b) \leq 2(\log\log ab)^2$ and the inequality follows. Moreover, we assume that $\mathcal{M}$ is superadditive:

$$\mathcal{M}(a) + \mathcal{M}(b) \leq \mathcal{M}(a+b).$$

  Again, this property is satisfied by the above functions.
- It will be convenient to introduce $\widetilde{\mathcal{M}}(N) = \mathcal{M}(N)/N$, so that $\widetilde{\mathcal{M}}(N) \in O(N)$ or $\widetilde{\mathcal{M}}(N) \in O(\mathcal{L}(N))$.
- Multiplication (resp. addition, division) of two elements of L_t can be done in $O(\mathcal{M}(t))$ (resp. $O(t)$, $O(\mathcal{M}(t)\log t)$) L-operations; see (21, Corollary 11.8) for instance.
- ω is a real number such that two square matrix of size n can be multiplied using $O(n^\omega)$ operations in their coefficient field. We assume $2 \leq \omega \leq 3$. The best known exponent is $\omega = 2.376$; see (13)
- Let $d_j = [k_{i,j} : k_{i,j-1}]$ and $D_j = d_1 \cdots d_j$ for $1 \leq j \leq m_i$.

4.2.1 Step $(\mathcal{A})$

We choose a random linear combination of monomials in $\{\xi_j^l \alpha_{j-1}^m \mid 0 \leq l < d_j,\ 0 \leq m < D_{j-1}\}$ with coefficients in L and compute its minimal polynomial over L. If this polynomial has degree D_j, we are done; otherwise we repeat the process. It is well-known that the ratio of primitive elements of $k_{i;j}$ is bounded from below by a constant; see (38, Fact 5.8) for instance. Hence, the expected number of trials is in $O(1)$. To compute a minimal polynomial, we suggest two methods, both due to Shoup:

$(\mathcal{A}_1)$ In (38), it is shown that such a minimal polynomial can be computed with $O(D_j^{(\omega+1)/2})$ L-operations; see Theorem 3.4 and the subsequent remark therein. Writing $D_j^{(\omega+1)/2} \leq f_i d_Y^{(\omega-1)/2}$, multiplying by $(r_i - s_i)$ and summing over i, we obtain a total cost in $O(\delta_F d_Y^{(\omega-1)/2})$. The method is based on the (theoretical) application of Tellegen's transposition principle. If fast

matrix multiplication is used, the method is superior to the next one. But so far, fast matrix multiplication methods have demonstrated little practical value. Athough this approach may give the best asymptotic behaviour, it is not recommanded in practice.

($\mathcal{A}_2$) More recently, Shoup has given an explicit version of Tellegen's principle for "power projections" that allows to compute minimal polynomials with $O(\mathcal{M}(D_j)D_j^{1/2}+D_j^2)$ L-operations; see (39). The algorithm has been implemented by Shoup, who reported an experimental behaviour in accordance with the theory. Writing $\mathcal{M}(D_j)D_j^{1/2}+D_j^2 \leq f_i\widetilde{\mathcal{M}}(D_j)\mathcal{D}_j \leq f_i\mathcal{M}(d_Y)$, the total cost is $O(\delta_F M(d_Y))$.

4.2.2 Step ($\mathcal{B}$)

Again, we suggest two methods based on the work of Shoup:

($\mathcal{B}_1$) A method in $O(D_j^{(\omega+1)/2})$, leading to a total cost of $O(\delta_F d_Y^{(\omega-1)/2})$, see (38, Theorem 3.5 and the subsequent remark therein). This algorithm is based on the following ingredients: The resolution of a Toeplitz system in $O(D_j\mathcal{L}(D_j)\log D_j)$ L-operations, the resolution of two power projection problems, and the evaluation of a univariate polynomial at an element of k_{ij}. The same comments apply as for ($\mathcal{A}_1$).

($\mathcal{B}_2$) By (39), power projections and evaluation in ($\mathcal{B}_1$) can be achieved by an explicit algorithm requiring $O(\mathcal{M}(D_j)D_j^{1/2} + D_j^2)$. This gives again $O(\delta_F M(d_Y))$.

4.2.3 Step ($\mathcal{C}$)

The number of coefficients of H that require a change of representation is bounded by $d_Y\delta_F/D_{j-1}$: Indeed, at the first function call, there is no need for a representation change, but for following calls, the truncation order is δ_F divided by the degree over L of the coefficient field. Each coefficient $C(\alpha_{j-1})$ can be viewed as a polynomial of degree less than D_{j-1} that must be evaluated at an element of k_{ij}. Three evaluation methods are considered. The first two ones are based on (5); the last one is a naive approach. It is interesting to note that they all yield the same asymptotic bound. For practical purposes, the last two ones are preferable since they are the simplest ones.

($\mathcal{C}_1$) From (38, Fact 3.1), each evaluation can be done at a cost of

$$O\left(D_{j-1}^{\omega/2}\left[D_j/D_{j-1}^{1/2}+1\right]+D_{j-1}^{1/2}\mathcal{M}(D_j)\right)$$

L-operations. Multiplying by $d_Y\delta_F/D_{j-1}$, simple calculations and inequalities lead to $O(\delta_F^2 M(d_Y))$. Note that classical matrix arithmetic may be used.

($\mathcal{C}_2$) Using the evaluation algorithm of (39), we get $O(\mathcal{M}(D_j)D_{j-1}^{1/2}+D_jD_{j-1})$, hence $O(\delta_F^2 M(d_Y))$ again.

($\mathcal{C}_3$) Finally, Horner scheme gives an evaluation cost of $D_{j-1}\mathcal{M}(D_j)$. Again, this gives $O(\delta_F^2 M(d_Y))$.

We could have multiplied by $\log_2 f_i$ instead of $(r_i - s_i) \leq \log_2 f_i$ and obtained $O(d_Y \delta_F \mathcal{M}(d_Y))$. However, we prefer to insist on the dependence on δ_F; the result $O(\delta_F^2 M(d_Y))$ cannot be deduced from the latter estimate since d_Y may be greater than δ_F, as demonstrated by the example $F(X,Y) = Y^n - X$ for $n > 2$.

4.2.4 Step ($\mathcal{D}$)

For $2 \leq j \leq m_i$, let g_j be a univariate polynomial of degree less than D_j such that $\alpha_{j-1} = g_j(\alpha_j)$. Such polynomials have been computed at steps ($\mathcal{B}$). Using a backward induction argument, we show that a polynomial u_j such that $\alpha_j = u_j(\alpha_{m_i})$ may be computed: This is trivially the case for $j = m_i$. Writing $\alpha_{j-1} = g_j(u_j(\alpha_{m_i}))$, we see that u_{j-1} can be determined at the cost of a polynomial evaluation. Then, ξ_j^a and ξ_j^b can easily be rewritten in terms of α_{m_i} by polynomial evaluations since their expression in terms of α_j is known by ($\mathcal{B}$).

Therefore, each step ($\mathcal{D}$) requires three evaluations of polynomials of degree less than f_i at elements of k_i, at a cost $O(f_i^{(\omega+1)/2})$ (resp. $O(\mathcal{M}(f_i) f_i^{1/2} + f_i^2)$, $O(f_i \mathcal{M}(f_i))$ if approach ($\mathcal{C}_1$) (resp. ($\mathcal{C}_2$), ($\mathcal{C}_3$)) is used. Multiplying by $(r_i - s_i)$ and summing over i, we get respectively $O(\delta_F d_Y^{(\omega-1)/2})$, $O(\delta_F \mathcal{M}(d_Y))$ and $O(\delta_F \mathcal{M}(d_Y))$.

To conclude step ($\mathcal{D}$), remark that polynomials u_{j-1} could easily be returned by adding a fourth element to the triplet $[G, \xi_j^b P^q, P^m(\xi_j^a + Q)]$.

We have proved:

Proposition 5 *For an integer $N \geq \delta_F$, all steps of type ($\mathcal{A}$), ($\mathcal{B}$), ($\mathcal{C}$) and ($\mathcal{D}$) required by* `RNPuiseux`*(L, F, N) can be performed with $O(\delta_F^2 \mathcal{M}(d_Y))$ field L-operations.*

Remark 4 Eric Schost pointed out that recent and remarkable results of Kedlaya and Umans (27; 28) could improve estimates of this section and of Section 4.3 and 4.4. In this paper, we have decided not to take these results into account for three reasons: It is not clear that they lead to an overall improvement of Theorem 3, they do not seem to be of practical interest at this point, and finally, they are of binary nature: They do not yield estimates for the number of L-operations.

4.3 Changes of variables

Lemma 2 *Let N be a positive integer. Let $H \in L_t[X,Y]$ and $\xi \in L_t$ be as in* `RNPuiseux`*. Define $U(X,Y) = H(\xi^b X^q, X^m(\xi^a + Y))/X^l$, where (m,q,l) corresponds to an edge Δ of a Newton polygon and (a,b) is given by algorithm*

`Bézout`. *Then, the number of L_t-operations required to compute $\widehat{H} = \widetilde{U}^N$ is in:*

- $O(N\mathcal{M}(d_Y(H)))$ *if* $\Delta \in \mathcal{GN}(H)$,
- $O(N\mathcal{M}(d_Y(H)) + v(H))$ *if* $\Delta \in \mathcal{EN}(H)$, *where* $v(H) = v_X(\mathrm{lc}_Y(H))$.

Proof First of all, note that points (i, j) of $\mathrm{Supp}(H)$ along the line $m\,i+q\,j = w$ are transformed into points along the horizontal line $j = w - l$ by the change of variables; see Figure 1.

Consider first the case $\Delta \in \mathcal{GN}(H)$. Since the slope is negative, we just need to perform a change of variable in $\bar{H}(X, Y) = \sum_{w=l}^{N+l} H_w(X, Y)$ with $H_w(X, Y) = \sum_{mi+qj=w} \alpha_{ij} X^j Y^i$.

$$\begin{aligned} H_w(\xi^b X^q, \xi^a X^m(1+Y)) &= \sum_{mi+qj=w} \alpha_{ij} \left(\xi^b X^q\right)^j \left(X^m\left(\xi^a + Y\right)\right)^i \\ &= X^w \sum_{mi+qj=w} \alpha_{ij} \xi^{bj} (\xi^a + Y)^i \\ &= X^w V_w(Y + \xi^a) \end{aligned}$$

where $V_w(Z) = \sum_{mi+qj=w} \alpha_{ij} \xi^{bj} Z^i$ is a univariate polynomial of degree at most d_Y.

If a and b are chosen as in `Bézout`, $0 \le b < q \le d_Y$ and $|a| \le m$. Since $m/q \le 1$, ξ^a and ξ^b can be computed with $O(\log_2 d_Y)$ L_t-operations using "square and multiply" technique. We then form relevant powers of ξ^b. The exponent j is bounded by $(N + l)/q$, as illustrated by Figure 1. Since slopes of generic Newton polygons are at least -1, we have $l/q \le d_Y/q \le d_Y$; computing all powers up to this bound is achieved in $O(N + d_Y)$ L_t-operations. Constructing the polynomials V_w requires at most $N d_Y$ multiplications in L_t. Finally, since $p > d_Y$, a shift in V_w can be reduced to the multiplication of two polynomials of degree at most d_Y, with cost $O(\mathcal{M}(d_Y))$; see (1, Problem 2.6). Since we must perform N such shifts, the total cost is in $O(\mathcal{M}(d_Y)N)$.

For $\Delta \in \mathcal{EN}(H)$, we proceed similarly. It is easily seen that $|a| \le |m| \le v(H)$, so that ξ^b may be computed with $O(\log_2 v(H))$ L_t-operations. The largest exponent j for powers of ξ^b is now at most $(N + l - m d_Y)/q$; see Figure 2. From $(l - m d_Y)/q \le v(H)$, we get a number of operations in $O(N + v(H))$. To complete the proof, remark that the number of H_w is the same as before.

Proposition 6 *Let $N \ge \delta_F$ be an integer. Changes of variables induced by* `RNPuiseux`*(L,F,N) can be performed with a number of L-operations in:*

$$O\left(\left[N\,\eta_F \mathcal{M}(d_Y) + v d_Y\right] \widetilde{\mathcal{M}}(d_Y) \log d_Y\right)$$

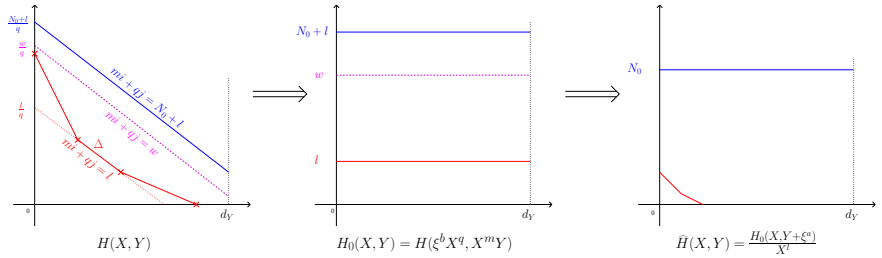

Fig. 1 Change of variables for negative slopes.

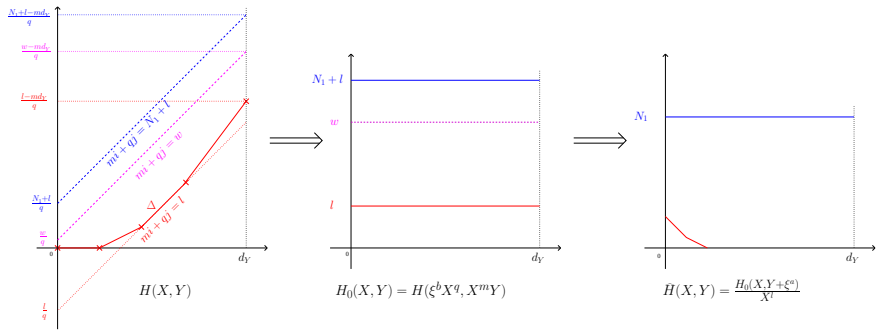

Fig. 2 Change of variables for negative slopes.

Proof To compute the RPE R_i, `RNPuiseux` performs at most $(r_i - s_i + 1)$ changes of variables over a field included in $L_{f_i} = k_i$, the first one corresponding to the exceptional Newton polygon $\mathcal{EN}(F)$.

Taking into account the extension L_{f_i}/L and Lemma 2, the number of L-operations required by the changes of variables for the i-th RPE is in:

$$O\left([N\,\mathcal{M}(d_Y)(r_i - s_i + 1) + v]\,\mathcal{M}(f_i)\log f_i\right).$$

Writing $\mathcal{M}(f_i)\log f_i \leq f_i\widetilde{\mathcal{M}}(d_Y)\log d_Y$ and summing over i yields the proposition, considering Theorem 1.

4.4 Factorizations cost

If N and t are positive integers, define:

$$\mathcal{F}(N,t) = \mathcal{M}(N^2)\log N + t\,\mathcal{M}(N)\log N\log p.$$

The factorization over $\mathbb{F}_{p^t}$ of a univariate polynomial of degree N in $\mathbb{F}_{p^t}[T]$ can be determined with an expected number of $O(\mathcal{F}(N,t))$ $\mathbb{F}_{p^t}$-operations; see (21, Corollary 14.30).

Remark 5 There exist algorithms with better complexity bounds; see (41; 26) for surveys and (28) for recent results. However, we have prefered to stick to well established algorithms, since factorization is not the bottleneck of our algorithm.

Proposition 7 *All factorizations of characteristic polynomials required by* `RNPuiseux`*(L,F,*δ_F*) can be computed with an expected number of L-operations in:*

$$O\left(\delta_F \mathcal{F}(d_Y, t_0)\mathcal{L}(d_Y)\log^2 d_Y\right) \subset O^{\sim}\left(\delta_F\left[\mathcal{M}(d_Y^2) + t_0 \log p \mathcal{M}(d_Y)\right]\right).$$

Proof Let L_t be the extension of L over which a factorization of a characteristic polynomial ϕ_Δ must be determined. First of all, we prove that the degree d_Δ of ϕ_Δ is at most d_Y/t. This is obviously true at the first stage of the algorithm, where $t = 1$. Assume that the property is true at a given stage of the algorithm and denote $H \in L_t[X, Y]$ the input polynomial, with an edge Δ. Let ξ be a root of ϕ_Δ and k be its mutiplicity in ϕ_Δ. At the next function call, let $\phi_{\Delta'}$ be a characteristic polynomial of the polygon yielded by this choice of Δ and ξ; denote its degree by $d_{\Delta'}$. Since k is the number of Puiseux series of H having $\xi X^{m_\Delta/q_\Delta}$ as initial term, necessarily, $d_{\Delta'} \leq k$. But $k[L_t(\xi) : L_t] \leq d_\Delta$ and $d_\Delta \leq d_Y/t$ by the induction hypothesis. Therefore $d_{\Delta'} \leq d_Y/[L_t(\xi) : L]$; this proves the property. Hence, factorization of a characteristic polynomial can be achieved with an average number of $O(\mathcal{F}(\frac{d_Y}{t}, t_0 t))$ L_t-operations. We must now multiply by the cost of each L_t-operations, namely $O(\mathcal{M}(t)\log t)$ L-operations. By property (7), we have $O(\mathcal{M}((\frac{d_Y}{t})^2)\log\frac{d_Y}{t}\mathcal{M}(t)\log t) \subset O(\mathcal{M}(d_Y^2)\mathcal{L}(d_Y)\log^3 d_Y)$ and $O(\mathcal{M}(\frac{d_Y}{t})\log\frac{d_Y}{t}\mathcal{M}(t)\log t) \subset O(\mathcal{M}(d_Y)\mathcal{L}(d_Y)\log^3 d_Y)$, no matter which arithmetic is used. Hence, the number of L-operations belongs to:

$$O\left(\mathcal{F}(d_Y, t_0 t)\mathcal{L}(d_Y)\log^2 d_Y\right). \tag{8}$$

We treat separately characteristic polynomials of $\mathcal{EN}(F)$. Remark that the sum of their degree is less than d_Y. Since $\mathcal{F}$ is a superadditive function of its first variable, the total cost of factoring characteristic polynomials of $\mathcal{EN}(F)$ is in $\mathcal{F}(d_Y, t)$. For other characteristic polynomials, we multiply estimate (8) by $r_i - s_i$, bound t by f_i and sum over i to obtain the result.

4.5 Bounding δ_F

The quantities δ_F and η_F are a priori unknown. To obtain a complete algorithm, we provide a bound for δ_F. In fact, we have:

Proposition 8 $\delta_F + v \leq v_X(R_F)$.

Proof By definition of the regularity index, for each CPS S_{ijk}, there exists $S_{i_0 j_0 k_0}$ with $i_0 \in \{1 \dots \rho\}$, $j_0 \in \{1 \dots f_{i_0}\}$ and $k_0 \in \{1 \dots e_{i_0}\}$ such that:

$$\frac{r_i - 1}{e_i} < v_X(S_{ijk} - S_{i_0 j_0 k_0}) \leq \frac{r_i}{e_i}.$$

This is equivalent to:

$$\frac{r_i - s_i - 1}{e_i} < v_X(S_{ijk} - S_{i_0j_0k_0}) - \frac{s_i}{e_i} \leq \frac{r_i - s_i}{e_i}.$$

If $v_X(S_{ijk} - S_{i_0j_0k_0}) \neq \frac{r_i}{e_i}$ $(\star)$, then e_i is a proper divisor of e_{i_0}. Thus, denoting $q = e_{i_0}/e_i > 1$, there exists $m \in \mathbb{N}$ and $\alpha \neq 0 \in \overline{L}$ such that $1 \leq m < q$ and:

$$S_{i_0j_0k_0}(X) = \widetilde{S_{ijk}}^{\frac{r_i}{e_i}}(X) + \alpha X^{\frac{r_i - 1}{e_i} + \frac{m}{e_{i_0}}} + \cdots$$

For any integer n, we denote $S_{ijk}^{[n,e_i]}$ the series obtained by applying the element $X \mapsto \zeta_{e_i}^n X^{1/e_i}$ of $\mathbb{G}_{e_i}$ to S_{ijk}. Hence, for $0 \leq l \leq q-1$, we have:

$$S_{i_0j_0k_0}^{[le_i,e_{i_0}]}(X) = \widetilde{S_{ijk}}^{\frac{r_i}{e_i}}(X) + \zeta_q^{ml}\alpha X^{\frac{r_i - 1}{e_i} + \frac{m}{e_{i_0}}} + \cdots$$

Remark that $q(r_i - s_i) + m - q = (r_i - s_i) + (q-1)(r_i - s_i - 1) + m - 1$. But $r_i - s_i - 1 \geq 0$, otherwise inequation $(\star)$ above cannot hold. Morever, $q > 1$ and $m \geq 1$. Therefore:

$$\sum_{l=0}^{q-1} \left(v_X(S_{ijk} - S_{i_0j_0k_0}^{[le_i,e_{i_0}]}) - \frac{s_i}{e_i} \right) = q\frac{r_i - s_i}{e_i} + \frac{m-q}{e_i} \geq \frac{r_i - s_i}{e_i}. \qquad (9)$$

In case, If $v_X(S_{ijk} - S_{i_0j_0k_0}) = \frac{r_i}{e_i}$, we trivially have $v_X(S_{ijk} - S_{i_0j_0k_0}) - s_i/e_i \geq (r_i - s_i)/e_i$.

Consider now a triplet $(i', j', k') \neq (i, j, k)$. Obviously:

$$v_X(S_{ijk} - S_{i'j'k'}) - \min\{\frac{s_i}{e_i}, \frac{s_{i'}}{e_{i'}}\} \geq 0.$$

Taking relation (9) and the subsequent remark into account, we get:

$$\sum_{\substack{(i',j',k') \\ (i',j',k') \neq (i,j,k)}} \left(v_X(S_{ijk} - S_{i'j'k'}) - \min\{\frac{s_i}{e_i}, \frac{s_{i'}}{e_{i'}}\} \right) \geq \frac{r_i - s_i}{e_i}.$$

Finally, summing over (i, j, k), we obtain $v_X(R_F) - v(2d_Y - 1) - M \geq \delta_F$, where:

$$M = \sum_{(i,j,k)} \sum_{\substack{(i',j',k') \\ (i',j',k') \neq (i,j,k)}} \min\{\frac{s_i}{e_i}, \frac{s_{i'}}{e_{i'}}\}.$$

To conclude the proof, we just need to show that $M \geq -v(2d_Y - 2)$. We proceed as follow: Since $s_i \leq 0$, we have $\min\{\frac{s_i}{e_i}, \frac{s_{i'}}{e_{i'}}\} \geq \frac{s_i}{e_i} + \frac{s_{i'}}{e_{i'}}$ and :

$$M \geq \sum_{(i,j,k)} \sum_{\substack{(i',j',k') \\ (i',j',k') \neq (i,j,k)}} \frac{s_i}{e_i} + \sum_{(i,j,k)} \left(\sum_{(i',j',k')} \frac{s_{i'}}{e_{i'}} \right) - \frac{s_i}{e_i} = -2v(d_Y - 1).$$

The last equality comes from Lemma 1.

4.6 Proof of Theorem 3 and 4

It is interesting to bound first the number of L-operations in terms of the output size, namely η_F.

Lemma 3 *Let $N \geq \delta_F$ be an integer. Assuming FFT-based multiplication, the number of L-operations required by* `RNPuiseux`*(L,F,N) belongs to:*

$$\mathcal{O}\left(d_Y\left[\eta_F N + v + \delta_F(\delta_F + d_Y + t_0 \log p)\right]\right).$$

Proof Follows easily from Propositions 5, 6 and 7, since $\mathcal{M}(d_Y^2) \in O(\mathcal{M}(d_Y)^2)$.

Corollary 1 *Define $\alpha_F = v_X(R_F)$. Assuming FFT-based multiplication, there is an algorithm to compute singular parts of all RPEs of F above 0 with a number of L-operations in:*

$$\mathcal{O}\left(\alpha_F d_Y\left[\alpha_F + d_Y + t_0 \log p\right] + v d_Y\right)$$

Proof By Proposition 8, `RNPuiseux`(L,F,α_F) returns the expected output. Then, apply Lemma 3 with $\delta_F \leq \alpha_F$ and $\eta_F \leq \alpha_F + d_Y$.

This proof raises a question: In practice, should R_F be computed or is it preferable to set $N = d_X(2d_Y - 1)$? We have decided not to include the computation of R_F in our bound since R_F is usually necessary to locate critical points anyway.

Proof of Theorem 3: Trivial consequence of Corollary 1 since $\alpha_F \leq d_X(2d_Y - 1)$.

Corollary 2 *Define $\mu_F = \deg_X R_F$. Assuming FFT-based multiplication, there exists an algorithm to compute singular parts of all RPEs of F above all affine critical points with a number of L-operations in:*

$$\mathcal{O}\left(d_Y \mu_F\left[d_Y + d_X + \mu_F t_0 \log p\right] + d_X d_Y^2\right).$$

Proof First of all, calculation of R_F can be done in $\mathcal{O}(d_X d_Y^2)$ L-operations (21, Corollary 11.18). Thus, this step is included in our complexity bound.

Moreover, R_F can be factorized over L in $O(\mathcal{F}(D, t_0)) \subset \mathcal{O}(\mu_F^2 + \mu_F t_0 \log p)$ L-operations using fast multiplication; see (21, Corollary 14.30). This step is also included in our complexity bound.

Then, let $R_F = c\prod_{i=1}^{m} \Phi_i^{h_i}$ be the factorization of R_F into monic irreducible factors Φ_i, set $t_i = \deg_X(\Phi_i)$ and let c_i be a root of Φ_i. Coefficients of $F_i = F(X + c_i, Y)$ can be computed at the cost of d_Y shifts in the coefficients of F in Y; the complexity of this step is in $\mathcal{O}(d_Y d_X)$ field operations in $L_{t_i} = L[T]/(\Phi_i(T))$ using a "divide and conquer" approach for shifts[4]. In terms of L-operations, the cost function is therefore in $\mathcal{O}(d_Y d_X t_i)$. Summing over i and bounding $\sum_i t_i$ by μ_F gives again a sufficient estimate for.

[4] Note that the method based on a reduction to a polynomial multiplication requires $p > d_X$ and cannot be applied

Define $v_i = v_X(\mathrm{lc}_Y(F_i))$. By Proposition 8, we can truncate F_i at order h_i. Hence, RPEs of F above $X = c_i$ may be computed by the function call `RNPuiseux`(L_{t_i},F_i,h_i), using:

$$\begin{aligned} &\mathcal{O}\left(d_Y\left[(h_i + d_Y)h_i + h_i t_i t_0 \log p\right]\right) \\ \subset\ &\mathcal{O}\left(d_Y\left[(\mu_F + d_Y)h_i + h_i \mu_F t_0 \log p\right]\right) \end{aligned}$$

field operation in L_{t_i}; see Lemma 3 with $\delta_{F_i} + v_i \leq h_i$, $\eta_{F_i} \leq h_i + d_Y$ and $N = h_i$. Multiplying by t_i and summing over i, trivial inequalities allow to conclude.

Proof of Theorem 4: For affine critical points, apply Corollary 2 with $\mu_F \leq d_X(2d_Y - 1)$. For RPEs above infinity, define $G(X,Y) = F(1/X,Y)X^{d_X} \in L[X,Y]$; G satifies hypotheses of `RNPuiseux` input. Obviously, $d_X(G) = d_X$ and $d_Y(G) = d_Y$. To complete the proof, remark that the bound of Theorem 3 applied to G is included in the bound of Theorem 4.

Corollary 3 *Assume that F is an irreducible polynomial of $\bar{L}[X,Y]$ and let d be its total degree. Suppose that $p > d_Y$. If FFT-based multiplication is used, there exists an algorithm to compute the genus of the curve $F(X,Y) = 0$ that requires $\mathcal{O}\left(d_Y^3 d_X^2 t_0 \log p\right) \subset \mathcal{O}\left(d^5 t_0 \log p\right)$ L-operations.*

Proof Trivial, by Riemann-Hurwitz formula; see (20), for instance.

5 Conclusion

We conclude this paper with a number of remarks:

Upgrading the arithmetic in a degree t finite field represented as a multiple extension to achieve an $O(M(t))$ complexity would render Section 4.2 obsolete.

Faster algorithms could be used for subproblems (factorization and polynomial evaluation, notably). It is not clear that this would improve our main result: Indeed the dominating term would be $O(d_Y \eta_F N)$ anyway (Lemma 3), since we have been unable to exhibit a better bound than $O(d_X d_Y)$ for N and η_F.

Algorithm `RNPuiseux` returns truncated series in "Horner-like form"; see examples in Section 3. If expanded forms are required, it is easy to check that they can be obtain in $O(\eta_F^2)$ operations in L.

Our bounds, combined with estimates for the size of a good prime p, yield bit-complexity results when the ground field K is an algebraic number field. Let us give an example: Define $K = \mathbb{Q}(\gamma)$ and let $M_\gamma \in \mathbb{Z}[T]$ be the minimal polynomial of γ over $\mathbb{Q}$.

If P is a multivariate polynomial in $K[\underline{X}]$, let (H,c) be the unique pair in $H \in \mathbb{Z}[T,\underline{X}]$, $c \in \mathbb{N}$, with $\deg_T(H) < w$ and $P(\underline{X}) = H(\gamma,\underline{X})/c$, where c is minimal. We define $\mathrm{ht}(P) = \max\{\log c, \log \|H\|_\infty\}$.

From (33, Theorem 17), we can easily deduce:

Theorem 5 *Let ϵ be a real number with $0 < \epsilon \leq 1$. There exists a Monte-Carlo like algorithm that computes the genus of the curve $F(X,Y) = 0$ with probability of error less than ϵ and a number of word operations in:*

$$\tilde{O}(d_Y^3 d_X^2 w^2 \log^2 \epsilon^{-1}[\mathrm{ht}(M_\gamma) + \mathrm{ht}(F)]).$$

Similar results can be obtained for Las Vegas and deterministic approaches.

References

1. Bini, D., Pan, V.Y.: Polynomial and Matrix Computations, *Progress in Theoretical Computer Science*, vol. 1. Birkhäuser, Saarbrücken (1994)
2. Bliss, G.A.: Algebraic functions (1933)
3. Bosma, W., Cannon, J., Playoust, C.: The Magma algebra system. I. The user language. J. Symbolic Comput. **24**(3-4), 235–265 (1997)
4. Bostan, A., Chyzak, F., Lecerf, G., Salvy, B., Schost, E.: Differential equations for algebraic functions. In: C.W. Brown (ed.) ISSAC'07: Proceedings of the 2007 international symposium on Symbolic and algebraic computation, pp. 25–32. ACM Press (2007). DOI 10.1145/1277548.1277553
5. Brent, R.P., Kung, H.T.: Fast algorithms for manipulating formal power series. J. ACM **25**(4), 581–595 (1978). DOI http://doi.acm.org/10.1145/322092.322099
6. Brieskorn, E., Knörrer, H.: Plane Algbebraic Curves. Birkhaüser (1986)
7. Chevalley, C.: Introduction to the Theory of Algebraic Functions of One Variable, *Mathematical Surveys*, vol. 6. AMS (1951)
8. Chistov, A.L.: Polynomial complexity of the Newton-Puiseux algorithm. In: Mathematical Foundations of Computer Science 1986, pp. 247–255. Springer-Verlag, London, UK (1986)
9. Chudnovsky, D.V., Chudnovsky, G.V.: On Expansion of Algebraic Functions in Power and Puiseux Series. I. Journal of Complexity **2**(4), 271–294 (1986)
10. Chudnovsky, D.V., Chudnovsky, G.V.: On Expansion of Algebraic Functions in Power and Puiseux Series. II. Journal of Complexity **3**(1), 1–25 (1987)
11. Cohn, P.M.: Puiseux's Theorem Revisited. Journal of Pure and Applied Algebra **24**, 1–4 (1984)
12. Comtet, L.: Calcul pratique des coefficients de Taylor d'une fonction algébrique. L'Enseignement Mathématique **2**(10), 267–270 (1964)
13. Coppersmith, D., Winograd, S.: Matrix multiplication via arithmetic progressions. In: STOC '87: Proceedings of the nineteenth annual ACM symposium on Theory of computing, pp. 1–6. ACM, New York, NY, USA (1987). DOI http://doi.acm.org/10.1145/28395.28396
14. Cormier, O., Singer, M.F., Trager, B.M., Ulmer, F.: Linear Differential Operators for Polynomial Equations. Journal of Symbolic Computation **34**(5), 355–398 (2002)

15. Dahan, X., Schost, E., Maza, M.M., Wu, W., Xie, Y.: On the complexity of the D5 principle. SIGSAM Bull. **39**(3), 97–98 (2005). DOI http://doi.acm.org/10.1145/1113439.1113457
16. Della Dora, J., Dicrescenzo, C., Duval, D.: About a New Method for Computing in Algebraic Number Fields. In: EUROCAL 85. Springer-Verlag LNCS 204 (1985)
17. Diaz-Toca, G., Gonzalez-Vega, L.: Determining Puiseux Expansions by Hensel's Lemma and Dynamic Evaluation. In: V. Ganzha, E. Mayr, E. Vorozhtsov (eds.) Computer Algebra in Scientific Computing, CASC 2002. Proceedings of the Fifth International Workshop on Computer Algebra in Scientific Computing, Yalta, Ukraine. Technische Universität München, Germany (2002)
18. Duval, D.: Diverses questions relatives au calcul formel avec des nombres algebriques (1987). Thèse d'État
19. Duval, D.: Rational Puiseux Expansions. Compositio Math. **70**(2), 119–154 (1989)
20. Eichler, M.: Introduction to the Theory of Algebraic Numbers and Functions. Academic Press (1966)
21. von zur Gathen, J., Gerhard, J.: Modern Computer Algebra. Cambridge University Press, Cambridge (1999)
22. Henry, J.P., Merle, M.: Complexity of Computation of Embedded Resolution of Algebraic Curves. In: Proceedings Eurocal 87, no. 378 in Lecture Notes in Computer Science, pp. 381–390. Springer-Verlag (1987)
23. van Hoeij, M.: An Algorithm for Computing an Integral Basis in an Algebraic Function Field. Journal of Symbolic Computation **18**, 353–363 (1994)
24. van der Hoeven, J.: Fast Evaluation of Holonomic Functions. Theoret. Comput. Sci. **210**(1), 199–215 (1999)
25. van der Hoeven, J.: Effective Analytic Functions. J. Symbolic Comput. **39**(3-4), 433–449 (2005)
26. Kaltofen, E.: Polynomial factorization: a success story. In: ISSAC '03: Proceedings of the 2003 international symposium on Symbolic and algebraic computation, pp. 3–4. ACM, New York, NY, USA (2003). DOI http://doi.acm.org/10.1145/860854.860857
27. Kedlaya, K.S., Umans, C.: Fast Modular Composition in Any Characteristic. In: I.C. Society (ed.) FOCS, pp. 481–490 (2008)
28. Kedlaya, K.S., Umans, C.: Fast Polynomial Factorization and Modular Composition (2009)
29. Kung, H.T., Traub, J.F.: All algebraic functions can be computed fast. J. ACM **25**(2), 245–260 (1978)
30. Li, X., Maza, M.M., Schost, E.: Fast arithmetic for triangular sets: from theory to practice. In: ISSAC '07: Proceedings of the 2007 international symposium on Symbolic and algebraic computation, pp. 269–276. ACM, New York, NY, USA (2007). DOI http://doi.acm.org/10.1145/1277548.1277585

31. Monagan, M.B., Geddes, K.O., Heal, K.M., Labahn, G., Vorkoetter, S.M., McCarron, J., DeMarco, P.: Maple 10 Programming Guide. Maplesoft, Waterloo ON, Canada (2005)
32. Poteaux, A.: Computing monodromy groups defined by plane algebraic curves. In: Proceedings of the 2007 International Workshop on Symbolic-numeric Computation, pp. 36–45. ACM, New-York (2007)
33. Poteaux, A.: Calcul de développements de puiseux et application au calcul de groupe de monodromie d'une courbe algébrique plane. Ph.D. thesis, Université de Limoges (2008)
34. Poteaux, A., Rybowicz, M.: Good Reduction of Puiseux Series and Complexity of the Newton-Puiseux Algorithm. In: Proceedings of the ISSAC '08 Conference, pp. 239–246. ACM, New-York (2008)
35. Poteaux, A., Rybowicz, M.: Towards a Symbolic-Numeric Method to Compute Puiseux Series: The Modular Part. http://arxiv.org/abs/0803.3027 (2008)
36. Poteaux, A., Rybowicz, M.: Good Reduction of Puiseux Series and Applications. Submitted to Journal of Symbolic Computation (2009)
37. Sasaki, T., Inaba, D.: Hensel Construction of $f(x, u_1, \ldots, u_i)$ at 2 Singular Point and its Application. Sigsam Bulletin **1**, 9–17 (2000)
38. Shoup, V.: Fast construction of irreducible polynomials over finite fields. J. Symbolic Comput **17**, 371–391 (1993)
39. Shoup, V.: Efficient computation of minimal polynomials in algebraic extensions of finite fields. In: ISSAC '99: Proceedings of the 1999 international symposium on Symbolic and algebraic computation, pp. 53–58. ACM, New York, NY, USA (1999). DOI http://doi.acm.org/10.1145/309831.309859
40. Teitelbaum, J.: The computational complexity of the resolution of plane curve singularities. Math. Comp. **54**(190), 797–837 (1990)
41. Von Zur Gathen, J., Panario, D.: Factoring polynomials over finite fields: a survey. J. Symb. Comput. **31**(1-2), 3–17 (2001). DOI http://dx.doi.org/10.1006/jsco.1999.1002
42. Walker, R.J.: Algebraic Curves. Springer Verlag, Berlin-New York (1978)
43. Walsh, P.G.: On the Complexity of Rational Puiseux Expansions. Pacific Journal of Mathematics **188**, 369–387 (1999)
44. Zariski, O.: Le problème des modules pour les branches planes. Hermann, Paris (1981)