

Computing Monodromy Groups defined by Plane Algebraic Curves

Adrien Poteaux
XLIM - DMI, UMR CNRS 6172
Université de Limoges
adrien.poteaux@unilim.fr

ABSTRACT

We present a symbolic-numeric method to compute the monodromy group of a plane algebraic curve viewed as a ramified covering space of the complex plane. Following the definition, our algorithm is based on analytic continuation of algebraic functions above paths in the complex plane. Our contribution is three-fold : first of all, we show how to use a minimum spanning tree to minimize the length of paths ; then, we propose a strategy that gives a good compromise between the number of steps and the truncation orders of Puiseux expansions, obtaining for the first time a complexity result about the number of steps; finally, we present an efficient numerical-modular algorithm to compute Puiseux expansions above critical points, which is a non trivial task.

Categories and Subject Descriptors

I.1.2 [Symbolic and Algebraic Manipulation]: Algorithms; G.1.5 [Numerical Analysis]: Roots of Nonlinear Equations

General Terms

Experimentation, Performance, Reliability

Keywords

Monodromy, Algebraic Curves, Symbolic-Numeric Computation, Riemann Surfaces

1. INTRODUCTION

We assume that the reader is acquainted with the basic theory of algebraic curves and Riemann surfaces. We suggest [19] and [15] for an introduction. We also assume some familiarity with the analytic continuation process for holomorphic functions.

1.1 Terminology and problem formulation

Let \mathcal{K} be a subfield of the complex number field \mathbb{C} , $\bar{\mathcal{K}}$ be its algebraic closure in \mathbb{C} and consider a plane algebraic curve

defined over \mathbb{C} , $\mathcal{C} = \{(x, y) \in \mathbb{C}^2 \mid F(x, y) = 0\}$ where $F \in \mathcal{K}[X, Y]$ is a squarefree polynomial, monic in the variable Y . We denote by d the degree of F in Y and by D its total degree. This algebraic curve along with the projection on the first coordinate x define a d -sheeted ramified covering of the complex x -plane \mathbb{C} that we refer to as (\mathcal{C}, x) . A complex number c such that the univariate polynomial $F(c, Y)$ has multiple roots is called a **critical point**. Critical points are in finite number since they are precisely the roots of the discriminant of F in Y . We denote them by c_1, \dots, c_p . A point that is not critical will be called **regular**.

Let a be a regular point. The polynomial $F(a, Y)$ has precisely d roots $\{y_1, \dots, y_d\}$. This set of roots form **the fiber** at a of the covering. By the Implicit Function Theorem, there exists d analytic functions $Y_1(x), \dots, Y_d(x)$ such that $F(x, Y_i(x)) = 0$ in a neighborhood of a and $Y_i(a) = y_i$. If $\gamma : [0, 1] \rightarrow \mathbb{C}$ is a loop in the x -plane starting and ending at a that does not meet any of the critical point, then the d functions Y_1, \dots, Y_d can be analytically continued along γ . When t gets close to 1, $\gamma(t)$ gets close to a so that the values of the continuations $\{Y_1(\gamma(t)), \dots, Y_d(\gamma(t))\}$ tend to the fiber at a . Therefore, there exists a permutation σ of $\{1, \dots, d\}$ so that :

$$Y_i(\gamma(t)) \rightarrow y_{\sigma(i)} = Y_{\sigma(i)}(a).$$

We obtain a morphism of the fundamental group into S_d :

$$\Psi : \Pi_1(a, \mathbb{C} \setminus \{c_1, \dots, c_p\}) \rightarrow S_d \\ \bar{\gamma} \mapsto \sigma.$$

We shall call the image of this morphism the **monodromy group** of the covering (\mathcal{C}, x) and denote it by \mathcal{M} . Changing the base point a or the numbering of y_1, \dots, y_d yields a conjugate of \mathcal{M} in S_d . Since any conjugate will suit our needs, we shall not worry further about this matter. Assuming that γ_i is a loop with base a that encloses a *single* critical point c_i , we set $\sigma_i = \Psi(\gamma_i)$. If the permutation σ_i is not the identity, c_i is called a **branch point**. The group \mathcal{M} is generated by the σ_i and **our goal is to compute a set of such generators**. For our purposes (see section 1.2), we choose a so that its real part is less than the real part of each critical points. Moreover we assume that a is chosen so that the critical points c_i can be ordered by increasing value of $\arg(c_i - a)$ (principal branch) as in [24]. Then, any set of loops $\{\gamma_i\}$ homotopic to those of Figure 1 will satisfy our need.

We emphasize that, although the construction of \mathcal{M} is purely analytic, the expected output is a set of permutations and has a combinatorial nature. Hence, no “approximation”

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SNC'07, July 25–27, 2007, London, Ontario, Canada.
Copyright 2007 ACM 978-1-59593-744-5/07/0007 ...\$5.00.

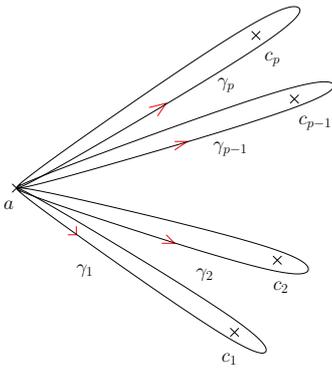


Figure 1: Classical paths for the monodromy

of the correct output can be tolerated. In particular, if \mathcal{K} is a number field, then our goal is to always output a correct result.

1.2 Motivations

Besides its direct relation to Galois theory (and inverse Galois problems, see [27] for instance), computing the monodromy group is one step of a more general program towards an effective Abel-Jacobi Theorem ([15, 19]). Indeed, to compute Abel's map, we need to obtain a canonical basis of the Riemann surface homology. In [24], Tretkoff and Tretkoff reduced the construction of this basis to the computation of the monodromy \mathcal{M} of the covering (\mathcal{C}, x) . This result motivated our interest in the monodromy problem.

This theorem allows to answer questions such as "Is a zero-degree divisor a function divisor?", which is important in Computer Algebra. For instance, it occurs when computing the antiderivative of an algebraic function [21, 4, 3], when studying algebraic solutions of ordinary differential equations [2], or when computing differential Galois groups [7]. Abel's map has also applications in Physics, notably to build solutions of KdV, KP and NLS equations ([11, 10]).

We note that there exists an implementation for Abel's map, described in [10]. In this work, Deconinck and Patterson use the Maple's `monodromy` command, which is not reliable (see section 1.4). Moreover, their implementation does not return provable error bounds. Our final aim is to have such bounds and our strategy is influenced by this purpose (see section 1.3).

1.3 Outline of our method

Assuming that the fiber at each regular point can be computed efficiently to arbitrary precision using a univariate polynomial solver (see [20]), it is tempting to apply the following process :

1. Choose $a = m_0, m_1, \dots, m_k = a$, a set of successive points on the loop.
2. Compute the fibers $F_i = \{y_{i,1}, \dots, y_{i,d}\}$ at m_i .
3. Connect elements of F_i to elements of F_{i+1} pairwise so that two connected elements correspond to values of the same continuation.

Several authors follow this scheme (see section 1.4). How each step is performed and how the paths are constructed

characterize the method. Our proposal matches this pattern.

First of all, we use a Euclidean minimal spanning tree to decrease the total path length. The idea was first suggested by Mark Van Hoeij (personal communication), but the practical application in our context requires some work. We show in section 3.1 how loops homotopic to those of Figure 1 can be emulated from the tree.

Then, we have chosen to connect fibers at m_i and m_{i+1} using truncated series expansions at controlled order, since they will provide error bounds for the connection and may also provide errors bounds for the integrals involved in Abel's map (see 1.2). The expansions are computed above appropriately chosen regular points x_i as well as above all the critical points c_j . Section 3.3 explains how to determine the m_i and the x_j so as to obtain a satisfactory trade-off between the number of steps and the series truncation orders : our strategy gives us an interesting complexity result for the number of m_i and x_j required.

Expansions above branch points are called **Puiseux series**. They provide important information : firstly, the permutation type of σ_i and secondly, how the functions connect when they are continued along a small loop around the critical point (see section 2.2). Moreover, expansions above critical points will be useful in the construction of the Abel map (see [10]). However, computing Puiseux expansion is not an easy task : pure numerical computations will lead to incorrect results, while symbolic methods suffer from overwhelming coefficient swell. To overcome these difficulties, we introduce in section 5.1 a hybrid method based on reduction modulo a prime number and floating point calculations that proves efficient in practice : modular computations provide the exact information that we need in order to proceed reliably with floating point numbers.

1.4 Previous works

We attempt to classify the possible approaches :

Compute fibers and connect. We gather in this paragraph methods that follow the strategy outlined in section 1.3. In [12], Mark van Hoeij and Bernard Deconinck describe a method that they implemented in Maple's `algcurses` package. Roughly speaking, they use first order approximation of the Y_j at m_i to predict the value of Y_j at m_{i+1} and connect to the point of F_{i+1} that seems the most likely to correspond. The correspondence criteria is heuristic and some heuristic error control is also performed. If the criteria fails, intermediary points between m_i and m_{i+1} are introduced and the process is restarted. In case inaccuracies are detected, the precision is increased or an error message is returned to the user suggesting that the precision should be increased. In general, the algorithm returns a correct result fairly quickly. But, as pointed out by the author themselves, it is very easy to exhibit examples where the result is wrong, runs into a very long subdivision process or detect unresolvable inaccuracies and returns an error message. The code is not reliable and requires human interaction to complete.

To resolve the matter, Van Hoeij and Rybowicz designed and implemented a different algorithm that returns a certified output. It relies on a theorem by Smith to compute step sizes and validate results. Some calculations are performed numerically, but others make use of interval arithmetic and exact arithmetic. This hybrid arithmetic guarantees correctness. At the time of writing, the performances of this

method are not satisfactory.

Differential equations. Van der Hoeven ([25, 26]) has given efficient algorithms to compute power series expansions of analytic functions defined by linear differential equations, so as to obtain asymptotically fast algorithms to compute functions at a high precision. In [8, 6, 9], a linear differential equation satisfied by the functions $\{Y_i(x)\}$ is constructed so that these methods could be used. Although obtaining the differential equation offers no theoretical difficulty, the size of the result and the running time of the computation can be overwhelming. Moreover, unlike Van der Hoeven or the Chudnovsky's, we are not interested in high precision evaluation of the functions. We just need enough precision to correctly continue the functions along the loops γ_i . Therefore, we have chosen not to follow this approach and have opted for a compromise between precision and step length.

Homotopy methods. Our continuation problem can be reformulated so as to fit in this class of methods ([1]). Indeed, the continuation $Y_i(\gamma(t))$ can be considered as a curve in \mathbb{R}^4 satisfying algebraic constraints. However, homotopy methods are too general in scope and not sufficiently adapted to our context. For instance, the strategy for finding roots of systems of polynomial equations is to avoid critical point that could cause instability and errors ([23]). Unfortunately, we do not have such freedom: our paths must get very close to the critical points if the c_i are close to each other. We have decided not to proceed further in this direction.

2. PUISEUX EXPANSIONS

If we consider F as a univariate polynomial in Y , Puiseux expansions are classical expressions for the roots of F . In this section, we recall the necessary material and refer the reader to [29] for proofs and details. If e is a positive integer, we introduce $\zeta_e = \exp \frac{2\pi i}{e}$. Let x_0 be an element of \mathcal{K} .

2.1 Algebraic point of view

THEOREM 1. *There exist positive integers e_1, \dots, e_s satisfying $\sum_{i=1}^s e_i = d$ (namely, a partition of d) so that, for each i ($1 \leq i \leq s$) and for each j ($1 \leq j \leq e_i$), there exists a fractional power series in $\mathcal{K} \left[\left[(X - x_0)^{\frac{1}{e_i}} \right] \right]$:*

$$S_{ij}(X) = \sum_{k=0}^{\infty} \alpha_{ik} \zeta_{e_i}^{jk} (X - x_0)^{\frac{k}{e_i}} \quad (1)$$

so that $F(X, S_{ij}(X)) = 0$ in $\mathcal{K} \left[\left[(X - x_0)^{\frac{1}{e_i}} \right] \right]$.

These d expansions are called **Puiseux expansions of F above x_0** and they can be effectively computed using the so-called Newton-Puiseux Algorithm ([29]). The coefficients α_{ik} belong to a finite algebraic extension of \mathcal{K} . The integer e_i is called the **ramification index** of the expansion and the partition $\{e_1, \dots, e_s\}$ will be referred to as the **ramification type** at x_0 . If there exists i so that $e_i > 1$, then x_0 is called a **branch point**. The branch points therefore form a subset of the critical points. In [13], Duval introduced a variation termed **rational Puiseux expansions** that allows to minimize the extension of \mathcal{K} in which the computation are performed (namely, the residue field). Rational Puiseux

expansions can be obtained using the Maple `puiseux` command. They are also implemented in the Magma and the Singular Computer Algebra Systems. If x_0 is a regular point, then ramification indices are all equal to 1. In this case, expansions (1) can also be computed using quadratic Newton iterations in power series fields (see [28] for instance).

It is important to note that Theorem 1 also holds if \mathcal{K} is any field of characteristic zero, or a field of characteristic $p > d$, with the obvious modifications (see [5] for positive characteristic). In this case, Newton-Puiseux algorithm can also be applied to compute the expansions.

2.2 Analytic point of view

For each positive integer e , we now choose a determination for the e -th root function that we denote by $\sqrt[e]{x}$. More precisely, the determination is characterized by an angle $\theta \in [-\pi, \pi[$ so that $\sqrt[e]{z} = |z|^{1/e} \exp(i(\arg z)/e)$ with $\arg z \in [\theta, \theta + 2\pi[$. The branch cut of the function is the half-line originating from 0 that form an angle θ with the positive real axis. The expression $X^{\frac{k}{e}}$ now corresponds to the function $\sqrt[e]{X^k}$ and the expansions (1) define d functions in an open disc centered at x_0 .

For any point (regular or not) x_0 , we define $\delta(x_0)$ to be the distance between x_0 and the nearest critical point (except itself!). If $\rho \in \mathbb{R}^{+*}$, then $D(x_0, \rho)$ will denote the open disc with center x_0 and radius ρ .

LEMMA 1. *The convergence radius of Puiseux expansions at point x_0 (regular or not) is equal to $\delta(x_0)$.*

PROOF. See [16]. \square

Let $B = (x_0, \theta)$ be a half-line in the x -plane originating from x_0 and characterized by the angle $\theta \in [-\pi, \pi[$ formed with the real axis. We select determinations for the e_i -th root functions so that the corresponding S_{ij} all admit B as branch cut. It is well known that the functions S_{ij} are analytic in the simply connected domain $D(x_0, \delta(x_0)) \setminus B$. Our path construction strategy is motivated by the following elementary facts:

FACT 1. *If r is a positive real number smaller than $\delta(x_0)$, the path $x(t) = x_0 + r \exp(2\pi(t + t_0)i)$ describes a circle $C(x_0, r)$ around x_0 when t varies from 0 to 1. For each i ($1 \leq i \leq s$), the set $\{S_{ij}(x_0)\}_{1 \leq j \leq e_i}$ is a subset of the fiber at x_0 . It can easily be shown that the analytic continuations along this path of the functions defined by F and these e_i values return to $\{S_{i\bar{j}+1}(x_0)\}_{1 \leq j \leq e_i}$, where $\bar{l} = l$ if $2 \leq l \leq e_i$ and $\bar{e_i} + 1 = 1$. In other words, the continuations permute cyclically the values $\{S_{ij}(x_0)\}_{1 \leq j \leq e_i}$. Therefore, Puiseux expansions allow to compute the action of this path on the fiber at x_0 . We shall call this action **the local monodromy** at x_0 . Moreover, the ramification type at x_0 coincides with the permutation type of the local action. Hence, if $x_0 = c_k$ is a branch point, the permutation type of σ_k is given by the the ramification type at c_k .*

FACT 2. *Since the S_{ij} are analytic in $D(x_0, \delta(x_0)) \setminus B$, continuations along a path in $D(x_0, \delta(x_0))$ that does not intersect B can be computed by evaluating the S_{ij} along the path.*

2.3 Truncation orders

In order to evaluate Puiseux expansions in the convergence disc, we shall need a bound for the truncation order. Let:

- $S(X) = \sum_{k=0}^{\infty} \mu_k (X - x_0)^{\frac{k}{e}}$ be a Puiseux expansion above x_0 ,
- $\overline{S}^n(X) = \sum_{k=0}^n \mu_k (X - x_0)^{\frac{k}{e}}$ be its order n truncation,
- $\rho = \delta(x_0)$ be the convergence radius of $S(X)$,
- $x_1 \in D(x_0, \rho)$,
- M be an upper bound for $\sup_{x \in D(x_0, \rho)} |S(x)|$,
- $\eta \in \mathbb{R}^{+*}$ be the precision required,
- $\beta = \left(\frac{|x_1 - x_0|}{\rho} \right)^{\frac{1}{e}}$.

PROPOSITION 1.

$$n \geq \frac{\ln\left(\frac{\eta}{M}\right) + \ln(1 - \beta)}{\ln(\beta)} - 1 \Rightarrow |S(x_1) - \overline{S}^n(x_1)| \leq \eta.$$

PROOF. If $e = 1$, the bound is a consequence of Cauchy's Theorem. If $e > 1$, perform a change of variable $G(X, Y) = F(x_0 + X^e, Y)$. \square

We still have to estimate M : we compute an upper bound for the value of $|S(x)|$ in the disk $D(x_0, \delta(x_0))$ by using root bounds for univariate polynomials given in [17, page 170].

2.4 Connecting expansions and fibers

Denote $\{S_1(X), \dots, S_d(X)\}$ the d Puiseux expansions above x_0 , x_1 a point in $D(x_0, \delta(x_0)) \setminus B$ and $\{y_1, \dots, y_d\}$ the fiber above x_1 . We shall need a method to determine a permutation σ so that $S_i(x_1) = y_{\sigma(i)}$. We use a black-box multiprecision root solver for univariate polynomials that outputs numerical approximations \tilde{y}_i for the y_i and positive real numbers ρ_i so that :

1. the discs $D(\tilde{y}_i, \rho_i)$ do not intersect and each disc contains exactly one point of the fiber,
2. if ϵ is the minimum distance between any pair of \tilde{y}_i and $r = \max\{\rho_1, \dots, \rho_d\}$, then $\epsilon/2 - r > 0$

For instance, such a black-box can be constructed using Smith's theorem (see section 4). Then, it is not difficult to prove the following result :

PROPOSITION 2. *If $|S_i(x_1) - \overline{S}_i^n(x_1)| < \epsilon/2 - r$ then σ is characterized by*

$$|\overline{S}_i^n(x_1) - y_{\sigma(i)}| = \min_{1 \leq j \leq d} \{|\overline{S}_i^n(x_1) - y_j|\}$$

To answer our question, we determine a truncation order n using Proposition 1 with η set to $\epsilon/2 - r$, compute expansions up to order n , evaluate numerically the $\overline{S}_i^n(x_1)$ and connect them to the \tilde{y}_i using this minimum distance criteria.

3. MONODROMY CALCULATION

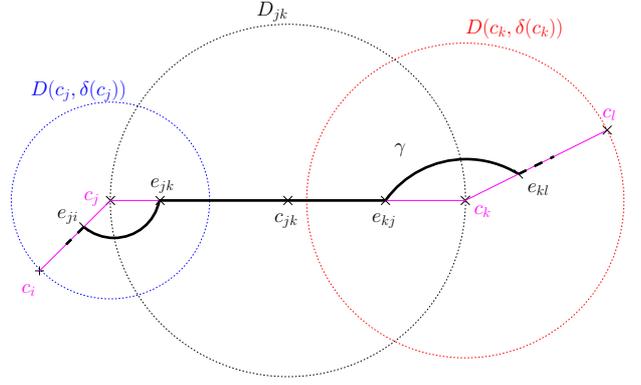


Figure 2: Convergence discs and connection points

3.1 Path construction

Our first contribution is to decrease the total length of the path we need to follow in order to compute \mathcal{M} . We explain below how to proceed. Let $V = \{c_0 = a, c_1, \dots, c_p\}$ be the set of critical points, augmented by the base point a . Let T be a Euclidean minimal spanning tree for the set V (such a tree can be computed with classical algorithms from Graph Theory). This tree has an interesting property, illustrated by Figure 2 :

LEMMA 2. *Let $e = \{c_j, c_k\}$ be an edge of the tree, $r = |c_j - c_k|$ and let c_{jk} be the midpoint of e . Then, there is no critical point in the disc $D_{jk} = D(c_{jk}, \frac{r}{2})$.*

PROOF. Simple consequence of the definition of T . \square

Thanks to this result, we can choose $e_{jk} \in D_{jk} \cap D(c_j, \delta(c_j))$ and $e_{kj} \in D_{jk} \cap D(c_k, \delta(c_k))$. We assume that, for all relevant indices k , the e_{jk} are all located at the same distance from c_j . Therefore, for each relevant indices l and k , there exists an arc of circle $A(e_{jk}, e_{jl}, +)$ centered at c_j and going counterclockwise from e_{jk} to e_{jl} . There is also an arc of circle oriented clockwise, that we denote by $A(e_{jk}, e_{jl}, -)$. See Figure 2. Let c_i be a critical point ($i > 0$). In T , there is a unique path from $a = c_0$ to c_i . Let $\{c_{i_j}\}_{1 \leq j \leq u}$ be the intermediary vertices (critical points) along this path. We define $\epsilon_{i_j i} = +$ if $i_j > i$ and $\epsilon_{i_j i} = -$ if $i_j < i$. Consider the path δ_i made of segments and arcs of circle :

$$[c_0, e_{i_1 0}]A(e_{i_1 0}, e_{i_1 i_2}, \epsilon_{i_1 i})[e_{i_1 i_2}, e_{i_2 i_1}]A(e_{i_2 i_1}, e_{i_2 i_3}, \epsilon_{i_2 i}), \dots [e_{i_u i}, e_{i i_u}]$$

Denote by β_i a loop centered at c_i , starting from $e_{i i_u}$ and going around c_i counterclockwise. Finally, we set :

$$\gamma'_i = \delta_i^{-1} \beta_i \delta_i.$$

PROPOSITION 3. *The path γ'_i is homotopic to the path γ_i in $\mathbb{C} \setminus \{c_1, \dots, c_p\}$*

PROOF. Consequence of the ordering defined for the critical points (see section 1.1). \square

This homotopy is illustrated by Figure 3.

3.2 Continuation

We need to continue analytically the functions $\{Y_j\}_{1 \leq j \leq d}$ defined in section 1.1 along the path γ'_i built in section 3.1.

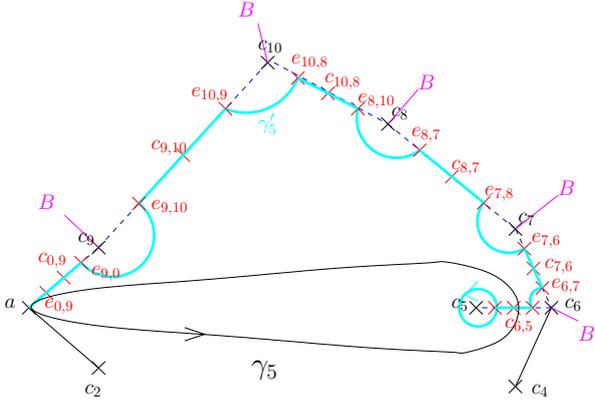


Figure 3: Path homotopic to γ_5

In this section, we describe the principles. In section 3.3 we shall explain how the efficiency can be dramatically improved.

First of all, we compute approximations of fibers above the base point a and the relevant points e_{jk} defined in section 3.1. To follow the segment $e = [e_{jk}, e_{kl}]$, we compute the expansions above c_{jk} and use them to connect the endpoints as described in section 2.4. To follow the arc $A(e_{jk}, e_{kl}, \pm)$, we compute Puiseux expansions at c_j and again make use of Fact 2 and Proposition 2 to determine the connection between the endpoints. To this end, we choose a determination for the e -th root functions so that the branch cut B of the expansions is “as far as possible” from both endpoints e_{jk} and e_{kl} . More precisely, we set $B = (c_j, (\arg(c_k - c_j) + \arg(c_l - c_j))/2 + \mu\pi)$, where μ equals 0 or 1, according to the arc orientation. With this value, the angle formed by c_j , B and each connection point is at least $\pi/4$. Therefore, we avoid as much as possible numerical problems that could occur if the connection points were too close to B . Finally, the action of loop β is emulated by means of Fact 1 and Proposition 2. It is clear that the continuation along δ_i^{-1} is not necessary since it can be obtained by reversing the correspondence of fibers given by δ_i . In summary, it suffices to compute $2p + 1$ Puiseux expansions as well as fibers above the $2p$ connection points e_{jk} to compute the monodromy. In practice however, it will be profitable to introduce more intermediary points to improve the efficiency (see section 3.3).

3.3 Truncation orders and connection points

Previous sections describe a strategy for computing the monodromy M that uses asymptotically $O(p)$ connection and expansion points, which in some sense is optimal since there are p branch points. Unfortunately, this is achieved at the cost of excessively high truncation orders for Puiseux expansions. Our second contribution is to give a good compromise between the number of connection points and the truncation orders.

In view of Proposition 1 with η set to $\epsilon/2 - r$, the truncation order n tends quickly to infinity when β tends to one, that is when the evaluation point x_1 is relatively close to the convergence radius, which is no surprise. This situation occurs when there are two close neighbors in T , say c_j and c_k , and a neighbor c_l of c_k at a greater distance. Then,

the connection point e_{kl} is necessarily relatively close to the convergence radius of the expansion at c_{kl} , midpoint of the edge $[c_k, c_l]$.

We propose to keep under control the bound growth by setting $\beta = \frac{1}{2}$. Indeed, with this value of β , we are left with the bound $n \geq 1 - \log_2\left(\frac{\epsilon - 2r}{M}\right)$, which depends essentially on the value of the function Y_i and is therefore hardly controllable. It turns out that we can enforce this value of β by introducing a logarithmic number of intermediary connection points between c_k and c_l . We explain our construction for the half-edge $\mathcal{H} = [c_k, c_{kl}]$.

Define $s = \left\lceil \log_3\left(\frac{\delta(c_{kl})}{\delta(c_k)}\right) + (e-1)\log_3(2) \right\rceil + 1$ and $\nu = 2^{-e}\delta(c_k)$. Set $p_0 = c_k$, $p_s = c_{kl}$ and for $1 \leq i \leq s-1$ define p_i to be the point of \mathcal{H} at distance $23^{i-1}\nu$ from c_k . Those are the expansion points. Then, for $1 \leq i \leq s$ let the connection point q_i be the point of \mathcal{H} at distance $3^{i-1}\nu$ from c_k . We obtain a sequence of consecutive points $c_k = p_0, q_1, p_1, q_2, \dots, q_s, p_s = c_{kl}$. We have the following proposition :

PROPOSITION 4. For $1 \leq i \leq s-1$, the value of β required to connect the expansions above p_{i-1} and p_i at q_i is no larger than $1/2$.

PROOF. The result can be deduced from a careful examination of the geometric situation. \square

A symmetric construction holds for the half-segment $[c_{kl}, c_l]$. The value of β at the midpoint c_{kl} to connect the two half-segment is obviously no larger than $1/2$. In practice, the truncation order of the expansions drops dramatically, at the cost of an acceptable increase of the number of steps. We obtain :

PROPOSITION 5. Let L_M and L_m denote respectively the length of the longest and shortest edges in T , and g be the genus of \mathcal{C} . Then, the number of expansion and connection points required to compute the monodromy by our method is in $O(p \log \frac{L_M}{L_m} + g)$, and so in $O(D^2 \log \frac{L_M}{L_m})$.

PROOF. Sum the value of s over all the half-edges of T and use trivial majoration to obtain the first term. Bound e at each critical point c_k by the greatest ramification index above c_k and apply Hurwitz formula to get the term g . Then, the genus g and the degree of the discriminant of F in Y are both in $O(D^2)$ (see respectively [19] and [28]). \square

For a family of F for which the ratio L_M/L_m is bounded, the previous corollary indicates that the number of steps grows linearly in the size of the output which is $\Omega(D^2)$. Finally, we obtain :

THEOREM 2. Assume that F belongs to $\mathbb{Z}[X, Y]$ and denote by $\|F\|_\infty$ the maximal absolute value of its coefficients. Then, the number of expansion and connection points is in $O(D^6 \log \|F\|_\infty)$.

PROOF. Let $P(X) = \text{Disc}_Y(F) \in \mathbb{Z}[X]$. The quantities L_M and L_m can respectively be bounded by $2\|P\|_2$ (twice the radius of a disc containing all the roots of P) and the separation bound $\text{sep}(P)$ [18]. We obtain that $L_M/L_m \leq \|P\|_2^p p^{(p+2)/2}$. Then, using the definition of P , we get

$$L_M/L_m \leq (2d-1)! \|F\|_2^{2d-1} d^{2f} 2^{2fd-f}$$

where f is the degree of F in X . Applying Sirling’s formula and using comparison of polynomial norms (see [18]), the result follows from Proposition 5. \square

To our knowledge, it is the first time that a such a bound for an algorithm computing \mathcal{M} is published. It is interesting to note that the growth is cubic in the size of the output.

4. ROOT ISOLATION

Several steps of our method require certified numerical root isolation. We present briefly techniques based on results by Smith ([22]) that satisfy our needs and are used in our implementation. It is understood that other approaches are possible and that we have not studied exhaustively the vast literature about this topic. In the same vein, we have not yet studied the error control of floating point computations.

4.1 Distinct roots

THEOREM 3. *Let $P(Z)$ be a N -th degree polynomial of $\mathbb{C}[Z]$ and suppose that z_1, \dots, z_N are N distinct complex numbers. Define :*

$$\rho_k = N |P(z_k)| / \prod_{\substack{i=1 \\ i \neq k}}^N |z_i - z_k|$$

Then the union of the discs $D(z_k, \rho_k)$ contains all the roots of $P(Z)$. Moreover, any connected component of this union consisting of K circles contains exactly K zeros of $P(Z)$.

Assume that a multiprecision numerical solver (see [20] for a survey) is available. We require that, given a polynomial P as above **with distinct roots**, when increasing the precision, the output of the solver converges towards the N roots of P . We apply the following algorithm :

1. Set heuristically a low precision.
2. Compute approximations z_1, \dots, z_N for the roots. If some roots coincides, increase precision and repeat this step until the approximations are all different.
3. Compute the ρ_k from Theorem 3. If the discs intersect, increase precision and go to step 2. Otherwise, return $\{(z_1, \rho_1), \dots, (z_N, \rho_N)\}$.

Assuming that the solver satisfies our requirement, the process converges. Moreover, the ρ_i can be made arbitrarily small. For instance, the last step can be repeated until the quantity $\epsilon/2 - r$ of section 2.4 is positive.

4.2 Multiple roots

Smith also established the following result :

THEOREM 4. *Let $P(Z)$ be a polynomial in $\mathbb{C}[Z]$. We suppose that L approximations $(z_1, M_1), \dots, (z_L, M_L)$ of the roots are given, where M_k is the multiplicity of the approximate root z_k . Then one can effectively compute positive real numbers ρ_1, \dots, ρ_L (depending on the M_i) such that, if the circles $D(z_k, \rho_k)$ do not intersect, then each circle $D(z_k, \rho_k)$ contains exactly one root of $P(Z)$ of multiplicity M_k .*

Therefore, using a multiprecision numerical solver and the strategy of the preceding paragraph, we can obtain arbitrarily small inclusion discs for the roots of a polynomial P with multiple roots :

1. Set heuristically a low precision.

2. Compute approximations z_1, \dots, z_L for the roots of P . Let M_1, \dots, M_L be the multiplicities returned by the numerical solver.
3. Compute the ρ_k from Theorem 4. If the discs intersect, increase precision and go to step 2. Otherwise, return $\{(z_1, M_1, \rho_1), \dots, (z_L, M_L, \rho_L)\}$.

The explicit description of the ρ_k would take too much space and we do not include it here. Radii computed by this theorem have the property to be small when the approximations are “close well separated” (in [22], B.T. Smith gives bounds for these radii). Moreover, if P is known only approximately, say by bounding boxes for its coefficients, then it is still possible to compute inclusion discs for the roots. We do not make use of the latter property at this point, but plan to investigate it further.

5. A HYBRID NUMERIC-MODULAR NEWTON-PUISEUX ALGORITHM

Expansions above regular points can be treated numerically with quadratic Newton iterations ([28]) or other means. So far, our experiments have shown that regular points are not the main concern and we shall not elaborate on this case. Hence, this section is devoted to the problem of computing Puiseux series above critical points with Newton-Puiseux Algorithm, and more precisely to the ramified part of this algorithm. Moreover, we restrict ourselves to the case where \mathcal{K} is a number field. Generalizations to finitely generated extensions of the field of rational numbers can also probably be devised.

With this restriction, critical points are algebraic numbers that annihilate the polynomial $\text{Disc}_Y(F)$. The degree in X of this polynomial is in $O(D^2)$, so it can have large degree irreducible factors. The coefficients of Puiseux expansions over a critical point c_k are also algebraic over the field $\mathcal{K}(c_k)$, so that the degree of the field over which the computations are performed can be excessively large ($O(D^3)$). Moreover, the size of the rational numbers involved grows very quickly. Gary Walsh shows in [31] that the ramified part of a classical Puiseux expansions can be computed in $O(d^{32})$ bit operations. Although it is not established that this bound is sharp, this estimate is not encouraging and confirm the poor performances experimentally observed. Similar estimates are obtained in [30] for a particular system of rational Puiseux extension. Even if we manage to obtain a symbolic expression for the series coefficients, the numerical evaluation of these coefficients is non trivial : due to devastating cancellations, numerical evaluations must sometimes be executed with a high number of digits (see section 5.5 for examples of this phenomenon). We conclude that symbolic computation followed by numerical evaluation is probably not the way to go.

On the other hand, pure numerical computations cannot be used directly. The slightest approximation causes Newton-Puiseux Algorithm to miss essential information, such as ramification indices. It also causes numerical instabilities. This fact is not surprising since any close approximation \bar{c}_k of a critical point c_k is a regular point and expansions above \bar{c}_k have a very small convergence radius $|\bar{c}_k - c_k|$. We need somehow to overcome these difficulties.

In this section, we will detail how we compute Puiseux expansions above critical points, introducing a new numeric-modular method. At this point, our method is not a fully

proved algorithm, but a heuristics that seems to perform unexpectedly well in practice (see section 5.5). We begin by recalling some facts about Newton-Puiseux Algorithm.

5.1 Newton-Puiseux Algorithm

Newton-Puiseux Algorithm imposes exact representation of the coefficients. In [13], D. Duval describes a variant that optimizes the field extension required. We sketch her approach below, although our method applies as well to the classical version.

Up to a change of variable, we can assume that the critical point is 0 and that there are Puiseux expansions $S_i(X)$ above 0 that satisfies $S_i(0) = 0$. We write e_i for their ramification index. In particular, these conditions imply that $F(0,0) = 0$ and $F_Y(0,0) = 0$ (F_Y is the derivative with respect to Y). We briefly explain how to compute the S_i . Let $F(X, Y) = \sum_{i,j} a_{ij} X^j Y^i$. The ramified part of the Newton-Puiseux algorithm makes successive change of variable to get a set of polynomials $G_i(X, Y) = F(P_i(X), Q_i(X, Y))$ for which the point $X = 0$ is regular. The **Newton Polygon** $\mathcal{N}(F)$ is defined as the lower part of the convex hull of the support $\text{Supp}(F) = \{(i, j) \in \mathbb{N}^2 \mid a_{ij} \neq 0\}$ of F . To a side Δ of $\mathcal{N}(F)$ corresponds three integers q, m and l with q and m coprime such that Δ is on the line $qj + mi = l$. Then, 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 such that (i, j) is in Δ . Here is a recursive version of Newton-Puiseux Algorithm for the ramified part (M_ξ denotes the multiplicity of ξ in ϕ_Δ) :

Algorithm Newton-Puiseux(F)

Input: F , a polynomial as above.

Output: a set $\{[G_i, P_i, Q_i]\}_i$ of triplets as above.

Begin

$L \leftarrow \{\}$

Compute $\mathcal{N}(F)$

For each side Δ of $\mathcal{N}(F)$ do

 Compute q, m, l and ϕ_Δ

 Compute u and v so that $uq - vm = 1$

 For each (distinct) root ξ of ϕ_Δ do

$H(X, Y) \leftarrow F(\xi^v X^q, \xi^u X^m(1 + Y))/X^l$

 If $M_\xi = 1$ then $L_1 \leftarrow \{[H, X, Y]\}$

 else $L_1 \leftarrow \text{Newton-Puiseux}(H)$ End

 For each $[G, P, Q]$ in L_1 do

$L \leftarrow L \cup \{[G, \xi^v P^q, \xi^u P^m(1 + Q)]\}$

 End

End

Return L

End.

It is important for the sequel to note that $(M_\xi, 0)$ will be one of the point of $\mathcal{N}(\mathcal{H})$ ([13]). The data u, v, q, m, l come directly from the polygons. They need to be computed exactly, since for instance q will contribute to the ramification index, which has to be obtained exactly for our strategy to succeed. In particular, root multiplicities of ϕ_Δ have to be known exactly. It is obvious that if ξ is replaced by a numerical approximation, the change of variable above will almost always produce a polynomial with trivial Newton polygon, that is, a quadrant rooted at $(0, 0)$. Moreover, multiplicities are not easy to determine if ϕ_Δ has numerical coefficients. Therefore the algorithm above cannot directly output any

useful information in the presence of approximations.

However, if we assume that Newton polygons are obtained by some other means (which implies that multiplicities are also known) and provided as an input, then we can :

1. Extract the approximate coefficients of F which are meaningful to compute $\mathcal{N}(F)$. The coefficients below $\mathcal{N}(F)$ should be equal to 0 : discard them.
2. Deduce the approximate ϕ_Δ .
3. Find root clusters of ϕ_Δ with the expected multiplicities.
4. For each cluster, deduce an approximate value of ξ , apply the numerical change of variable and proceed with the recursive call.

With this technique, we obtain approximate Puiseux series with correct ramification indices, that is correct combinatorial data. Our experiments show that approximations for the series coefficients are reasonably accurate, yielding accurate evaluations of algebraic functions in the neighborhood of critical points.

To compute the exact data that are needed, we use computations modulo a well chosen prime number p . However, the correspondence between data modulo p and numerical data is not easy to establish. This is the topic of the next section.

5.2 Modular-numeric strategy

Applying the recursive Newton-Puiseux algorithm of section 5.1 to a polynomial F yields a function call tree : vertices correspond to function calls and a directed edge (a, b) indicates that the function call a has produced the function call b . Each vertex can be labeled with the data $(\mathcal{P}, [d_1, \dots, d_u])$, where \mathcal{P} is the Newton polygon of the input (for instance, represented by a list giving the endpoints of each side) and each d_i is an integer partition describing root multiplicities of the i -th characteristic polynomial of \mathcal{P} . We denote this tree by $\mathcal{T}(F)$. In particular, the root vertex R of $\mathcal{T}(F)$ is labeled with $\mathcal{N}(F)$ and the corresponding partitions. The leaves are labeled with polygons that have only one side ending at $(0, 1)$ and only one trivial partition, since the characteristic polynomial of the unique side has degree 1.

From now on, we denote by p a prime number, and by $\bar{\mathbb{F}}_p$ an integral closure of \mathbb{F}_p . If A is a multivariate polynomial over \mathbb{Z} , we denote by \bar{A} its reduction mod p . If A has coefficients in a number field $\mathbb{Q}(\alpha)$, we also denote by \bar{A} its reduction modulo a prime ideal of $\mathbb{Q}(\alpha)$ dividing p , provided that such a reduction makes sense. Let us assume that p satisfies the following conditions :

1. The coefficients of F can be reduced mod p . We denote by \bar{F} the image of F in $\bar{\mathbb{F}}_p[X, Y]$.
2. \bar{F} is squarefree and $p > d$. These conditions ensure that Puiseux expansions exist for \bar{F} and can be computed using Newton-Puiseux algorithm (see [5]).
3. If $P(X) = \text{Disc}_Y(F)$, then the shape of the squarefree factorization of P is preserved. That is, if $P(X) = \prod_i P_i^i$, where the P_i are squarefree and coprime, then $\text{deg } \bar{P} = \text{deg } P$ and the \bar{P}_i are also squarefree and coprime.

4. The labeled trees $\mathcal{T}(F)$ and $\mathcal{T}(\bar{F})$ are isomorphic : there exist a bijection of the vertices that preserves edges and labels.

The first three conditions imply that the curve has good reduction at p (actually, one must also consider the situation above ∞ for condition 3 to ensure that there is good reduction at p). This criteria can be deduced from [14], chapter 3, section 6. It is easy to prove that there is only a finite number of primes p that do not satisfy the first three conditions. Moreover, they can easily be checked. As for the fourth condition, one can also prove that there is a finite number of p for which it fails, but we do not have a simple criteria to characterize them. Our implementation uses random primes chosen sufficiently large so that the probability that it is not fulfilled is very small. We have not yet met a case where this heuristics fails in practice.

We assume in the sequel that $\mathcal{T}(\bar{F})$ has been computed using Newton-Puiseux algorithm. We define the level of a vertex v of $\mathcal{T}(\bar{F})$ to be the number of edges between the root R of $\mathcal{T}(\bar{F})$ and v and we denote :

$$\mathcal{S}_l = \{(\mathcal{P}_1, [d_{1_1} \dots, d_{1_{u_1}}]), \dots, (\mathcal{P}_s, [d_{s_1} \dots, d_{s_{u_s}}])\}$$

the set of all labels of level l of $\mathcal{T}(\bar{F})$. By our choice of p , this set is also the set of level l labels of $\mathcal{T}(F)$. We denote by $\mathcal{T}_l(F)$ the subtree of $\mathcal{T}(F)$ induced by the vertices of level less or equal to l .

To explain our strategy for computing numerical Puiseux series, we proceed recursively (note that this recursion is different from that of our description of Newton-Puiseux) : we assume that all numerical computations have been achieved successfully until level l using $\mathcal{T}_{l-1}(F)$. We obtain approximate polynomials $\mathcal{H}_l = \{\tilde{H}_1, \dots, \tilde{H}_s\}$, which are the inputs for level l function calls. In order to proceed at level $l+1$, we need to determine Newton polygons for the \tilde{H}_i and find clusters of roots for characteristic polynomials, as explained in section 5.1. To this end we need to establish a bijection between the set of possible labels \mathcal{S}_l and \mathcal{H}_l . We use a three stage filtering process :

1. First of all, we group the H_i that have the same sequence of labels as ancestors in $\mathcal{T}_{l-1}(F)$ (on the path from R). We call \mathcal{H}'_l such a group. For each group \mathcal{H}'_l , we form the subset of elements of \mathcal{S}_l that have also the same sequence of labels as ancestors in $\mathcal{T}_{l-1}(F)$ and call it \mathcal{S}'_l . Each pair $(\mathcal{H}'_l, \mathcal{S}'_l)$ is then treated separately at the next stage.
2. Subsets of elements of \mathcal{H}'_l having the same Newton polygon are then formed, using the technique described in section 5.3. We call \mathcal{H}''_l such a set and \mathcal{S}''_l the subset of elements of \mathcal{S}'_l corresponding to this Newton polygon. Again, each pair $(\mathcal{H}''_l, \mathcal{S}''_l)$ is treated separately at the last stage.
3. At this point, all elements of \mathcal{H}''_l share the same Newton polygon. We still need to relate each element of \mathcal{S}''_l to the appropriate sequence of partitions in \mathcal{S}''_l . For this, we use the method given in section 5.4 below. Finally, we are left with a family of sets \mathcal{H}'''_l , each set corresponding to a unique label of level l . The method of section 5.4 also provides approximations for the roots of the characteristic polynomials. Therefore, we have all the information needed to compute \mathcal{H}_{l+1} and apply the process recursively.

In fact, our implementation proceeds slightly differently, but we do not have enough space to include all the details.

5.3 Relating polygons and polynomials

In this part, we are given $(\mathcal{H}'_l, \mathcal{S}'_l)$ from the preceding section. We drop the index l to simplify the notation. Noting that the polygons in \mathcal{S}' may not be distinct, we introduce :

- $L = [\mathcal{P}_1, \dots, \mathcal{P}_q]$, the list of distinct Newton polygons from \mathcal{S}' ,
- $n = [n_1, \dots, n_q]$, a list of integers such that n_k is the number of occurrences of \mathcal{P}_k in \mathcal{S}' .
- $\mathcal{H}' = [\tilde{H}_1, \dots, \tilde{H}_h]$ the list of polynomials with approximate coefficients.

We proceed recursively :

1. If $q = 1$, then output the set $(\mathcal{H}', \mathcal{P}_1)$.
2. We compute the lower convex envelope \mathcal{P} of the union of the Newton polygons in L .
3. There exists one point $(i, j) \in \mathcal{P}$, which belongs to one of the \mathcal{P}_k , but not to all of them. Therefore, we split the list L into two parts : L_1 is the sublist made of polygons for which $(i, j) \in \mathcal{P}_k$, and L_2 contain the others. We denote the latter $L_2 = [\mathcal{P}_{k_1}, \dots, \mathcal{P}_{k_r}]$. Then, we sort the elements of \mathcal{H}' by decreasing absolute values of the coefficient of $X^j Y^i$ and also split \mathcal{H}' as follow : the $N_1 = n_{k_1} + \dots + n_{k_r}$ first polynomials form a list \mathcal{H}'_1 and the remaining ones form the list \mathcal{H}'_2 . We then apply recursively the algorithm to (L_1, \mathcal{H}'_1) and (L_2, \mathcal{H}'_2) , obtaining two sets R_1 and R_2 .
4. We output the set $R_1 \cup R_2$.

This heuristics could be turned into a certified algorithm provided that we control the accuracy of all computations.

5.4 Relating partitions and polynomials

In this part, we are given a pair $(\mathcal{H}''_l, \mathcal{S}''_l)$, such that all elements of \mathcal{S}''_l have the same Newton polygon \mathcal{P} , but may differ by their partition sequence. We drop the index l in the sequel. Let Δ be a side of \mathcal{P} and denote by \tilde{H}_i ($1 \leq i \leq r$) the elements of \mathcal{H}'' . To each \tilde{H}_i correspond a characteristic polynomial $P_i = \phi_{\Delta i}$. All the P_i have the same degree d_0 . From \mathcal{S}'' , we obtain all the r possible partitions (d_1, \dots, d_r) of d_0 . We recall that d_k ($k > 0$) is a set of positive integers d_{k_j} such that $\sum_j d_{k_j} = d_0$, representing possible multiplicities for the roots of the P_i .

We are left with the following problem : given the $(P_i)_{1 \leq i \leq r}$ and the $(d_k)_{1 \leq k \leq r}$, find the bijective map between these two multisets that associates to each P_i the multiplicities of its roots. Moreover, determine approximations for the distinct roots. We just sketch a possible approach for this question. Our implementation is significantly more sophisticated.

1. For each P_i , compute complex numbers $\tilde{z}_{i1}, \dots, \tilde{z}_{ik_i}$, positive real numbers $\rho_{i1}, \dots, \rho_{ik_i}$ and a partition m_i of d_0 , $m_i = (m_{i1}, \dots, m_{ik_i})$ so that :
 - The disc $D_{ij} = D(\tilde{z}_{ij}, \rho_{ij})$ contains m_{ij} roots (not necessarily distinct) of P_i .
 - For $1 \leq j \leq k_i$, the discs D_{ij} do not intersect.

Again, this information can be obtained using Smith's results (see section 4.2).

2. Then, three cases may occur :

- (a) There is a permutation of the $(m_i)_{1 \leq i \leq r}$ that is equal to $(d_i)_{1 \leq i \leq r}$. In this case the $\tilde{z}_{i_1}, \dots, \tilde{z}_{i_k}$ are accepted as approximations of the roots of P_i for all $1 \leq i \leq r$.
- (b) There is a partition m_i that cannot be obtained by summing terms of one of the d_i . Hence, we cannot form root clusters for P_i that correspond to any of the partition d_i . In this case, we have lost too much accuracy along the computations : The coefficients of P_i are too far away from the exact ones. We restart the complete computation of the Puiseux series with an higher precision.
- (c) The partitions m_i that are not in $(d_i)_{1 \leq i \leq r}$ may split to give some of the d_i . We increase the precision and restart the computation at step 1.

We suspect that this heuristics can be turned into a certified algorithm provided that certified arithmetic is used. However, at the time of writing, some points are still under investigation.

5.5 Experimental results

In this section we will present some examples illustrating the efficiency and accuracy of our numerical-modular Newton-Puiseux algorithm. Examples have been computed with a Maple 10 prototype implementation. We emphasize that it is a work in progress and that the implementation is neither optimized nor polished.

We first set $M_{a,d}(X) = X^d - 2(aX - 1)^2$. This family of polynomials come from [17, page 170]. The polynomial $M_{a,d}(X)$ has two real roots whose distance is less than $2a^{-\frac{d+2}{2}}$.

The first example will show the stability of numerical computations. We consider the polynomial $F_1(X, Y) = Y^3 - M_{10,5}(X)$. To obtain the groupe \mathcal{M} , we need to compute Puiseux expansions above roots of $M_{10,5}(X)$ up to the term in $X^{\frac{16}{3}}$. In the next table, the first column gives the number of digits of the mantissa (setting of the Digits variable in Maple), the second column display the number of correct digits using symbolic computation followed by numerical evaluation, the last column is the number of correct digits using our method.

Digits	Symbolic + Numeric	Our algorithm
10	0	7
40	0	36
50	6	47

Thus, at a fixed precision, our algorithm gives better results than numerical evaluation of the symbolic Puiseux expansions. Moreover, we obtain a number of significant digits very close to the setting of Digits.

We now consider the ramified part of the following example : $F_2(X, Y) = (Y^3 - M_{10,6}(X))(Y^3 - M_{10,3}(X)) + Y^2X^5$. Its discriminant in Y has an irreducible factor $P(X) \in \mathbb{Z}[X]$ of degree 30, with some coefficients greater than 10^{13} . If we compute Puiseux expansions above roots of $P(X)$, we have the following results for the coefficient in $X^{\frac{1}{2}}$:

Digits	Symbolic + Numeric	Our algorithm
10	0	4
20	0	15
30	5	29

We also considered the family of polynomials defined by the following recursive expression:

$$G_n(X, Y) = \left(Y^{\lceil \frac{n}{2} \rceil} - P_{\lceil \frac{n}{2} \rceil}(X) \right) G_{\lfloor \frac{n}{2} \rfloor}(X, Y)$$

where

$$P_{n_0}(X) = \frac{1}{n_0 3!} X \left(X^{n_0} + (n_0 - 1) X - \frac{1}{n_0!} \right).$$

The Puiseux series above 0 for this family all have coefficients in \mathbb{Q} . It is a case most favorable for the classical Newton-Puiseux since no algebraic numbers are involved.

Here, we perform computations with a precision of ten digits, and consider computation times. Moreover, we give the number of correct digits of the results. Symbolic computations are obtained with the Maple algorithm from the `algcurves` package. Timings and precision are given for the ramified part of all the expansions above 0.

Polynomial used	time for symbolic computations	Our algorithm	
		time	precision
G_8	0.031 s	0.029 s	9
G_{12}	0.041 s	0.099 s	9
G_{16}	2.3 s	0.221 s	9
G_{20}	0.751 s	0.550 s	9
G_{24}	2.889 s	0.920 s	9
G_{28}	8.509 s	1.719 s	9
G_{32}	30.820 s	5.040 s	9

Modular computations have been done with the prime number $p = 100019$.

6. CONCLUSION

The hybrid method that we have introduced allows us to efficiently and accurately compute approximate Puiseux series above critical point. To our knowledge, it is the first of its kind. As a consequence, we can compute local monodromies and follow paths around critical path straightforwardly. Combining this we our strategy for analytic continuation of algebraic functions along segments between critical points, we obtain a new algorithm for computing the monodromy of the covering (\mathcal{C}, x) .

Because of the lack of space, we have omitted many technical details and optimizations. Moreover, several fundamental questions remain : error bound for the Puiseux series coefficients could lead to a certified algorithm, which we have not at this point. The coefficients computed with our hybrid algorithm seem to be reasonably accurate ; it remains to understand why. The choice of a good prime p still relies on a heuristics. We have reasons to believe, though, that this step could be turned into an algorithm (moreover, we thank an anonymous referee for his suggestion regarding this issue). Finally, the complexity of the overall algorithm is not clear, although the number of steps is under control as explained in the paper. All these points are still under investigation. The implementation is in a preliminary version and we are actively working on improvements.

Acknowledgments

We thank Mark Van Hoeij for pointing out the possibility of a minimal spanning tree strategy and for his inspiring work with Bernard Deconinck ([12] and the Maple `monodromy` command). Finally, we wish to thank Marc Rybowicz, for introducing us to the subject as well as many helpful discussions and suggestions regarding this work. He also significantly contributed to the redaction and proofreading of this paper.

7. REFERENCES

- [1] E. L. Allgower and K. Georg. Numerical Path Following. In *Handbook of Numerical Analysis, Vol. V*, Handb. Numer. Anal., V, pages 3–207. North-Holland, Amsterdam, 1997.
- [2] F. Baldassarri and B. Dwork. On Second Order Linear Differential Equations with Algebraic Solutions. *Amer. J. Math.*, 101(1):42–76, 1979.
- [3] L. Bertrand. Computing a Hyperelliptic Integral Using Arithmetic in the Jacobian of the Curve. *Applicable Algebra in Engineering, Communication and Computing*, 6:275–298, 1995.
- [4] M. Bronstein. Integration of Elementary Functions. *Journal of Symbolic Computation*, 9(2):117–173, 1990.
- [5] A. Campillo. *Algebroid Curve in Positive Characteristic*, volume 813 of *Lecture Notes in Mathematics*. Springer - Verlag, New York - Berlin, 1980.
- [6] D. V. Chudnovsky and G. V. Chudnovsky. On Expansion of Algebraic Functions in Power and Puiseux Series. I. *Journal of Complexity*, 2(4):271–294, 1986.
- [7] E. Compoint and M. Singer. Relations linéaires entre solutions d’une équation différentielle (Linear Relations Between the Solutions of a Differential Equation). *Ann. Fac. Sci. Toulouse, Série 6, Vol. 7*, no. 4:659–670, 1998.
- [8] L. Comtet. Calcul pratique des coefficients de Taylor d’une fonction algébrique. *L’Enseignement Mathématique*, 2(10):267–270, 1964.
- [9] O. Cormier, M. F. Singer, B. M. Trager, and F. Ulmer. Linear Differential Operators for Polynomial Equations. *Journal of Symbolic Computation*, 34(5):355–398, 2002.
- [10] B. Deconinck and M. S. Patterson. Computing the Abel Map. *preprint*, 2007.
- [11] B. Deconinck and H. Segur. The KP Equation with Quasiperiodic Initial Data. *Phys. D*, 123(1-4):123–152, 1998.
- [12] B. Deconinck and M. van Hoeij. Computing Riemann Matrices of Algebraic Curves. *Phys. D*, 152/153:28–46, 2001. *Advances in Nonlinear Mathematics and Science*.
- [13] D. Duval. Rational Puiseux Expansions. *Compositio Math.*, 70(2):119–154, 1989.
- [14] M. Eichler. *Introduction to the Theory of Algebraic Numbers and Functions*. Pure and Applied Mathematics. Academic Press, 1966.
- [15] O. Forster. *Lectures on Riemann Surfaces*. Graduate Text in Mathematics. Springer Verlag, New-York, Berlin, 1981.
- [16] A. I. Markushevich. *Theory of Functions of a Complex Variable. Vol. III*. Revised English edition, translated and edited by Richard A. Silverman. Prentice-Hall Inc., Englewood Cliffs, N.J., 1967.
- [17] M. Mignotte. *Mathematics for computer algebra*. Springer-Verlag, New York, 1992. Translated from the French by Catherine Mignotte.
- [18] M. Mignotte and D. Stefanescu. *Polynomials, an Algorithmic Approach*. Discrete Mathematics and Theoretical Computer Science. Springer, 1999.
- [19] R. Miranda. *Algebraic Curves and Riemann Surfaces*. Graduate Studies in Mathematics. American Mathematical Society, Providence, RI, 1995.
- [20] V. Pan. Solving Polynomials: Some History and Recent Progress. *SIAM Review*, 39(2):187–220, 1997.
- [21] R. H. Risch. The Problem of Integration in Finite Terms. *Transactions of the American Mathematical Society*, 139:167–189, 1969.
- [22] B. T. Smith. Error Bounds for Zeros of a Polynomial Based Upon Gerschgorin’s Theorems. *J. ACM*, 17(4):661–674, 1970.
- [23] A. Sommese, J. Verschelde, and C. Wampler. Numerical Factorization of Multivariate Complex Polynomials. *Theoretical Computer Science*, 315(2-3):651–669, 2004. Special Issue on Algebraic and Numerical Algorithms edited by I.Z. Emiris, B. Mourrain, and V.Y. Pan.
- [24] C. Tretkoff and M. Tretkoff. Combinatorial Group Theory, Riemann Surfaces and Differential Equations. *Contemp. Math.*, 33:467–517, 1984.
- [25] J. van der Hoeven. Fast Evaluation of Holonomic Functions. *Theoret. Comput. Sci.*, 210(1):199–215, 1999.
- [26] J. van der Hoeven. Fast Evaluation of Holonomic Functions Near and in Regular Singularities. *Journal of Symbolic Computation*, 31(6):717–743, 2001.
- [27] H. Volklein. *Groups as Galois Groups*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge, 1997.
- [28] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, Cambridge, 1999.
- [29] R. J. Walker. *Algebraic Curves*. Springer Verlag, Berlin-New York, 1978.
- [30] P. G. Walsh. On the Complexity of Rational Puiseux Expansions. *Pacific Journal of Mathematics*, 188:369–387, 1999.
- [31] P. G. Walsh. A Polynomial-time Complexity Bound for the Computation of the Singular Part of an Algebraic Function. *Mathematics of Computation*, 69:1167–1182, 2000.