On the Computational Complexity of Resolving Curve Singularities and Related Problems

Peter Stadelmeyer

Research Institute for Symbolic Computation
Johannes Kepler Universität
Linz, Austria

Peter.Stadelmeyer@risc.uni-linz.ac.at

PhD thesis

November 2000

On the Computational Complexity of Resolving Curve Singularities and Related Problems

Dissertation

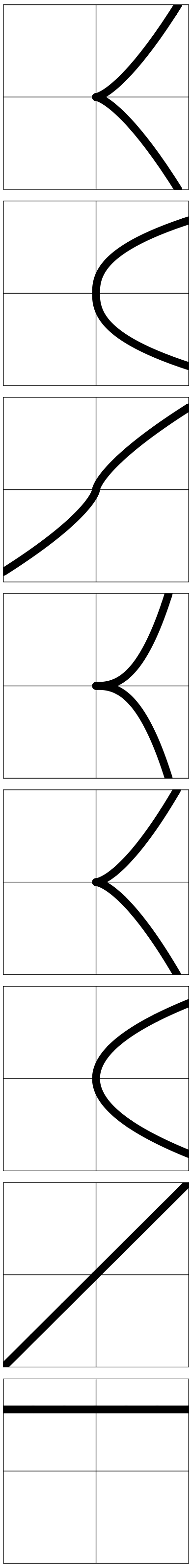
zur Erlangung des akademischen Grades
"Doktor der technischen Wissenschaften"

Eingereicht von
Peter Stadelmeyer

Erster Begutachter	: O.Univ.-Prof. Dr. Franz Winkler Institut für symbolisches Rechnen Johannes Kepler Universität Linz
Zweiter Begutachter	: O.Univ.-Prof. Dr. Jochen Pfalzgraf Institut für Computerwissenschaften Universität Salzburg

Angefertigt am: Institut für symbolisches Rechnen
Technisch–Naturwissenschaftliche Fakultät
Johannes Kepler Universität Linz

Linz, November 2000



On the Computational Complexity of Resolving Curve Singularities and Related Problems

Peter Stadelmeyer

Mathematics Subject Classification 2000.
Primary 14Q05. Secondary 14H20, 68W30, 68W40.

Linz, November 2000

To Elke,
Brigitte and Helmut

Summary

For a number of questions arising in the area of plane algebraic geometry full analysis (resolution) of curve singularities is of great importance. Examples for that are the determination of the genus of a curve, the computation of the linear system of adjoint curves of a curve, or the parametrization of a curve.

Constructive resolution methods are essentially based on repeated applications of transformations, which transform the original curve into a curve devoid of singularities. But not all of these methods are equally well suited for practical use in computer algebra systems. One of the main problems from an algorithmic point of view is the possibly very high degree of intermediate results.

In the framework of this thesis it turned out, that the above mentioned questions can be answered efficiently with the help of the Newton–Puiseux algorithm for computing local parametrizations of curve-branches. For this reason a considerable part of this thesis is devoted to a thorough complexity analysis of the Newton–Puiseux algorithm. There one sees that the determination low degree bounds for the intermediate results is crucial for the efficiency of this method.

We were able to show, that the resolution method described in this thesis has a computational complexity w.r.t. to the number of operations in the ground field of order μ^3, where μ is the Milnor number of the singularity.

This result serves as a basis for complexity statements about the determination of the genus of plane curves and the computation of the linear system of adjoint curves.

Zusammenfassung

Für eine Reihe von Fragestellungen in der ebenen algebraischen Geometrie ist die vollständige Analyse (Auflösung) von Kurvensingularitäten von grosser Bedeutung. Beispiele dafür sind die Bestimmung des Geschlechts einer Kurve, die Berechnung des linearen Systems der adjungierten Kurven für eine Kurve oder die Parametrisierung einer Kurve.

Konstruktive Auflösungsmethoden basieren im wesentlichen auf wiederholten Anwendungen von Transformationen, welche die ursprüngliche Kurve in eine Kurve ohne Singularitäten überführen. Nicht alle diese Methoden eignen sich jedoch in gleicher Weise für die praktische Verwendung in Computer Algebra Systemen. Eines der Hauptprobleme aus algorithmischer Sicht ist dabei der möglicherweise sehr hohe Grad der Zwischenresultate.

Im Rahmen dieser Dissertation hat sich gezeigt, dass mit Hilfe des Newton–Puiseux Algorithmus zur Berechnung lokaler Parametrisierungen von Kurvenzweigen obige Fragen sehr effizient beantwortet werden können. Ein wesentlicher Teil dieser Arbeit ist daher einer genauen Komplexitätsanalyse des Newton–Puiseux Algorithmus gewidmet. Als entscheidend für die Effizienz dieser Methode erweist sich dabei die Bestimmung niedriger Gradschranken für die Zwischenresultate.

So kann gezeigt werden, dass die hier vorgestelle Auflösungsmethode eine Komplexität bezüglich der Anzahl der Operationen im Grundkörper der Ordnung μ^3 besitzt, wobei μ die Milnor-Zahl der Singularität ist.

Dieses Resultat dient als Grundlage für Komplexitätsaussagen zur Geschlechtsbestimmung ebener Kurven sowie zur Berechnung des linearen Systems der adjungierten Kurven.

Ich versichere, dass ich die Dissertation selbständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich auch sonst keiner unerlaubten Hilfe bedient habe.

Linz, November 2000 Peter Stadelmeyer

Acknowledgments

I would like to sincerely thank my teacher and advisor Franz Winkler for his support and encouragement during my work on this thesis. Many thanks for all the discussions we had on this topic and his valuable comments.

I also would like to thank all members of the *Research Institute for Symbolic Computation* for giving me help whenever I needed some and for making this institute a highly creative and stimulating environment.

Contents

List of Figures

CHAPTER 1

Introduction

Resolution of singularities has a long history in algebraic geometry. The problem of resolving singularities can be stated informally as follows: "Given a hypersurface V. Does there exist a hypersurface W which is birational equivalent to V such that W has no singular points?". We know, that this is always true for fields of characteristic zero. Proofs were given (n denotes the dimension of V): For curves ($n = 1$) by B. Riemann 1865, M. Noether 1870, R. Dedekind 1882, for surfaces ($n = 2$) by R. J. Walker 1935 (only for the field of complex numbers), O. Zariski 1939, for $n = 3$ by O. Zariski 1944, and finally for any n by H. Hironaka 1964.

In this thesis a resolution algorithm for curves in characteristic zero is described and its computational complexity is analyzed. Although this is the best known case and solved theoretically more than 100 years ago, from a computational point of view, especially concerning the theoretical complexity and practical efficiency, it is still a challenging problem.

As computer algebra systems become more and more standard tools in many areas of science and engineering, one is interested in fast algorithms capable of solving increasingly complicated problems. In this context efficient resolution algorithms are of importance for several reasons.

First, they are a fundamental tool within constructive algebraic geometry. Once all singularities of a curve $\mathcal{C}$ are resolved, the genus of $\mathcal{C}$, a linear system of adjoint curves of $\mathcal{C}$, or a parametrization of $\mathcal{C}$ (if possible) can be computed rather easily.

Second, resolution algorithms can be used for further applications such as absolute factorization of bivariate polynomials, integration of algebraic functions, or algebraic coding theory.

Constructive resolution methods are essentially based on repeated applications of transformations, which transform the original curve into a curve devoid of singularities. But not all of these methods are equally well suited for practical use in computer algebra systems. One of the main problems concerning the computational complexity is the possibly very high degree of intermediate results.

We show, that curve singularities can be resolved efficiently with the help of the Newton–Puiseux algorithm for computing local parametrizations of curve-branches. For this reason a considerable part of this work is devoted to a thorough complexity analysis of the Newton–Puiseux algorithm. There one sees that the determination of low degree bounds for the intermediate results is crucial for the efficiency of this method.

The resolution method described in this thesis has a computational complexity w.r.t. to the number of operations in the ground field of order μ^3, where μ is the Milnor number of the singularity. This result serves as a basis for complexity statements about the determination of the genus of plane curves and the computation of the linear system of adjoint curves.

For the reason that questions concerning the computational efficiency play an important role in this work, most definitions and results are formulated using such concrete objects as polynomials or power series. So we share Abhyankar's view of algebra he stated in his "Historical Ramblings in Algebraic Geometry". There he divides algebra into three categories, namely high-school algebra (polynomials, power series), college algebra (rings, fields, ideals), and university algebra (functors) and says:

> The method of high-school algebra is powerful, beautiful, and accessible. So let us not be overwhelmed by the groups-rings-fields or the functorial arrows of the other two algebras and thereby lose sight of the power of the explicit algorithmic processes given to us by Newton, Tschirnhausen, Kronecker, and Sylvester.
>
> [**Abhyankar, 1976**]

In Chapter 2 we introduce notation and quote some well known results about local properties of plane algebraic curves. No new results are presented there. A detailed characterization and analysis of curve singularities are given in Chapter 3. In this chapter we also develop the theoretical results needed for low (optimal) degree bounds for intermediate results in the Newton–Puiseux algorithm. The last section of the chapter deals with a definition of adjoint curves based on truncated Puiseux parametrizations. This allows us, once the singular parts of all Puiseux parametrizations are computed, to avoid any lazy evaluation in the adjoint algorithm. In Chapter 4 a rational version of the Newton–Puiseux algorithm (as described by D. Duval) is presented and its computational complexity is analyzed. To be precise, we give an upper bound for the number of operations in the ground field needed to compute the singular parts of all Puiseux parameterizations of a given curve. This corresponds to resolving all singularities of the curve. Applications of the resolution algorithm together with complexity bounds are shown in Chapter 5. Finally, in Appendix A the most important algorithms are described using a self explaining pseudo code.

For practical experimentation the algorithms were implemented using Maple V (release 5) and will become part of the CASA-system. CASA is a system for computational algebra and constructive algebraic geometry developed at the Research Institute for Symbolic Computation, Linz, see [**Mňuk and Winkler, 1996**], [**Hemmecke et al., 2000**]).

This work was partially supported by the "Austrian Fonds zur Förderung der wissenschaftlichen Forschung" project number P11160-TEC "HySaX" and "SFB – Numerical & Symbolic Scientific Computing" project number F013 04.

CHAPTER 2

Basic Notions

In this chapter we introduce the notation used throughout the book and state well known results from algebraic geometry, mainly concerning plane algebraic curves. No new results are presented in this chapter.

Throughout the book let k be a commutative, algebraically closed field of characteristic zero, e.g. the complex numbers. When dealing with computational aspects (Chapter 4 and Chapter 5) we also have to consider a ground field, i.e. the field where the input objects are defined, which we denote by k'. We assume that k' is a subfield of k but we do not require that k' is algebraically closed. In practice k' will be the field of rational numbers.

We will use the following notation for domains over k, respectively k', where $\mathbf{X}$ denotes a finite set of variables.

$k[\mathbf{X}]$	Ring of all polynomials in $\mathbf{X}$ with coefficients in k.
$k(\mathbf{X})$	Field of all rational functions in $\mathbf{X}$ with coefficients in k.
$k[[\mathbf{X}]]$	Ring of all power series in $\mathbf{X}$ with coefficients in k.
$k((\mathbf{X}))$	Field of all meromorphic series in $\mathbf{X}$ with coefficients in k.
$k\langle\langle t\rangle\rangle$	Field of all Puiseux series in t with coefficients in k.

TABLE 1. Notation for Basic Domains

Since $k(\mathbf{X})$ and $k((\mathbf{X}))$ are the quotient fields of $k[\mathbf{X}]$ and $k[[\mathbf{X}]]$, respectively, the following inclusion diagram holds.

$$\begin{array}{ccc} k[\mathbf{X}] & \subset & k(\mathbf{X}) \\ \cap & & \cap \\ k[[\mathbf{X}]] & \subset & k((\mathbf{X})) \end{array}$$

As a consequence of this, definitions given for $k((\mathbf{X}))$ restrict in a canonical way to the other three domains.

Definition 1. Let $n \in \mathbb{N}$ and $f(x_1, \ldots, x_n) \in k((x_1, \ldots, x_n))$ be unequal to zero. We can write $f(x_1, \ldots, x_n)$ as

$$f(x_1, \ldots, x_n) = \sum_{d \in \mathbb{Z}} f_d(x_1, \ldots, x_n),$$

where $f_d(x_1, \ldots, x_n) = \sum_{i_1+\cdots+i_n=d} a_{i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$, i.e. $f_d(x_1, \ldots, x_n)$ is *homogenous of degree* d. We define the *order of* $f(x_1, \ldots, x_n)$ and the *degree of* $f(x_1, \dots, x_n)$ as

$$\begin{aligned}
\operatorname{ord}(f(x_1,\dots,x_n)) = m \quad & \text{if } f_d(x_1,\dots,x_n) = 0, \text{ for all } d < m,\\
& \text{and } f_m(x_1,\dots,x_n) \neq 0,\\
\deg(f(x_1,\dots,x_n)) = m \quad & \text{if } f_d(x_1,\dots,x_n) = 0, \text{ for all } d > m,\\
& \text{and } f_m(x_1,\dots,x_n) \neq 0.
\end{aligned}$$

The *initial* and *degree forms of* $f(x_1, \dots, x_n)$ are defined as $f_d(x_1, \dots, x_n)$, where $d = \operatorname{ord}(f(x_1, \dots, x_n))$, $\deg(f(x_1, \dots, x_n))$, respectively.

The order and degree of 0 are taken to be ∞ and $-\infty$.

1. Algebraic Sets

The most natural space to study local properties of algebraic curves is the affine plane. Nonetheless, sometimes we will need more general spaces, e.g. defining affine blowups requires the definition of the product space of an affine plane and a projective line. In the following we define affine and projective spaces of arbitrary dimension and show how they are related to each other.

Due to the arbitrary dimension of these spaces, the notation of this section has to be slightly different from the one used in the rest of the book.

For this section let $n \in \mathbb{N}$ denote the dimension of affine or projective spaces.

1.1. Affine Space

Definition 2. The set

$$\mathbf{A}^n = \{(a_1, \dots, a_n) \mid a_1, \dots, a_n \in k\}$$

is called the *affine* n*-dimensional space over* k. The spaces $\mathbf{A}^1$ and $\mathbf{A}^2$ are also called the *affine line over* k and the *affine plane over* k.

Definition 3. Let $f(x_1, \dots, x_n) \in k[x_1, \dots, x_n]$. The *affine hypersurface defined by* $f(x_1, \dots, x_n)$, written as $V(f(x_1, \dots, x_n))$, is the set

$$V(f(x_1, \dots, x_n)) = \{(a_1, \dots, a_n) \in \mathbf{A}^n \mid f(a_1, \dots, a_n) = 0\}.$$

For $n = 2$ and $n = 3$ affine hypersurfaces are also called *affine curves* and *affine surfaces*. The *degree of* $V(f(x_1, \dots, x_n))$ is defined as the degree of $f(x_1, \dots, x_n)$.

Let S be a set of polynomials in $k[x_1, \dots, x_n]$. The *affine algebraic set defined by* S, written as $V(S)$, is defined as

$$V(S) = \bigcap_{f(x_1,\dots,x_n)\in S} V(f(x_1, \dots, x_n)).$$

1.2. Projective Space

Many theorems in algebraic geometry (especially those including global statements about algebraic sets) are only true if we also take into account "points at infinity". To do so we define so-called projective spaces.

Definition 4. The *projective* n*-dimensional space over* k, denoted by $\mathbf{P}^n$, is defined as

$$\mathbf{P}^n = \{(a_0, \dots, a_n) \mid a_0, \dots, a_n \in k \text{ not all zero}\},$$

where points of the form $(\lambda a_0, \dots, \lambda a_n)$, for $\lambda \in k \setminus 0$, are considered to be equal.

As for affine spaces, the spaces $\mathbf{P}^1$ and $\mathbf{P}^2$ are also called the *projective line over* k and the *projective plane over* k.

Definition 5. Let $F(x_0, \dots, x_n) \in k[x_0, \dots, x_n]$ be a homogenous polynomial. The *projective hypersurface defined by* $F(x_0, \dots, x_n)$, denoted by $V(F(x_0, \dots, x_n))$, is the set of points

$$V(F(x_0, \dots, x_n)) = \{(a_0, \dots, a_n) \in \mathbf{P}^n \mid F(a_0, \dots, a_n) = 0\}.$$

For $n = 2$ and $n = 3$ projective hypersurface are also called *projective curves* and *projective surfaces*. The *degree of* $V(F(x_0, \dots, x_n))$ is again defined as the degree of $F(x_0, \dots, x_n)$.

For a set S of homogenous polynomials in $k[x_0, \dots, x_n]$ the *projective algebraic set defined by* S, written as $V(S)$, is defined as

$$V(S) = \bigcap_{F(x_0, \dots, x_n) \in S} V(F(x_0, \dots, x_n)).$$

1.3. Affine Coverings of Projective Spaces

As we said above, the n-dimensional projective space was introduced to add "points at infinity" to the n-dimensional affine space. Therefore, there is a natural correspondence between these two spaces which we describe now.

We split the space $\mathbf{P}^n$ into two subsets A, $B \subset \mathbf{P}^n$ which are defined as $A = \{(a_0, \dots, a_n) \in \mathbf{P}^n \mid a_0 \neq 0\}$ and $B = \{(a_0, \dots, a_n) \in \mathbf{P}^n \mid a_0 = 0\}$. By definition A and B are disjoint, $A \cup B = \mathbf{P}^n$, and A and B are isomorphic to $\mathbf{A}^n$ and $\mathbf{P}^{n-1}$, respectively. The isomorphisms are given by $h_A \colon A \to \mathbf{A}^n, (a_0, a_1, \dots, a_n) \mapsto (a_1/a_0, \dots, a_n/a_0)$ and $h_B \colon B \to \mathbf{P}^{n-1}, (a_0, a_1, \dots, a_n) \mapsto (a_1, \dots, a_n)$. The subspace B is called the *hyperplane at infinity* and points of B are called *points at infinity*.

So we can think of a projective space as being the disjoint union of an affine space and a hyperplane at infinity, which justifies our terminology used above of adding "points at infinity" to the affine space.

In general, we can decompose $\mathbf{P}^n$ into

$$A_i = \{(a_0, \dots, a_n) \in \mathbf{P}^n \mid a_i \neq 0\}$$

and

$$B_i = \{(a_0, \dots, a_n) \in \mathbf{P}^n \mid a_i = 0\},$$

for $0 \leq i \leq n$. Since $\mathbf{P}^n = \bigcup_{i=0}^n A_i$ we see that $\mathbf{P}^n$ can be covered by $n+1$ copies of $\mathbf{A}^n$. Of course this covering is by no means disjoint.

Using this correspondence between affine and projective spaces we define for affine hypersurfaces corresponding projective hypersurfaces and vice versa in the following ways.

Definition 6. Let $f(x_1, \dots, x_n) \in k[x_1, \dots, x_n]$ be of degree D defining the affine hypersurface $X \subset \mathbf{A}^n$. We can write $f(x_1, \dots, x_n)$ as $f(x_1, \dots, x_n) = \sum_{d=0}^D f_d(x_1, \dots, x_n)$, where each $f_d(x_1, \dots, x_n)$ is homogenous of degree d. The *homogenization of* $f(x_1, \dots, x_n)$ *w.r.t. the variable* x_0 is defined as the homogenous polynomial $F(x_0, \dots, x_n) \in k[x_0, \dots, x_n]$, where

$$F(x_0, \dots, x_n) = \sum_{d=0}^{D} x_0^{D-d} f_d(x_1, \dots, x_n).$$

The projective hypersurface $V(F(x_0, \dots, x_n))$ is called the *homogenization of* X *w.r.t.* x_0 or the *projective closure of* X *w.r.t.* x_0.

Definition 7. Let $F(x_0, \dots, x_n) \in k[x_0, \dots, x_n]$ be a homogenous polynomial defining the projective hypersurface $X \subset \mathbf{P}^2$. The *dehomogenization of* $F(x_0, \dots, x_n)$ *w.r.t. the variable* x_0 is defined as the polynomial $f(x_1, \dots, x_n) \in k[x_1, \dots, x_n]$, where

$$f(x_1, \dots, x_n) = F(1, x_1, \dots, x_n).$$

The affine hypersurface $V(f(x_1, \dots, x_n))$ is called the *dehomogenization of* X *w.r.t.* x_0.

The dehomogenizations of $F(x_0, \dots, x_n)$ and X w.r.t. x_i, for $1 \leq i \leq n$, are defined analogously.

The definitions of homogenization and dehomogenization can be generalized in a straight forward way from hypersurfaces to arbitrary algebraic sets.

We close this subsection with a short note about "points at infinity" of affine hypersurfaces. For this, let $f(x_1, \dots, x_n) \in k[x_1, \dots, x_n]$ be the defining polynomial of some $X \subset \mathbf{A}^n$. To find the points at infinity of X we look at the homogenized hypersurface $X' \subset \mathbf{P}^n$ of X w.r.t. x_0. The points at infinity of X are then obtained by intersecting X' with the hypersurface at infinity (defined by $x_0 = 0$). But this is simply the same as looking at the degree form of $f(x_1, \dots, x_n)$.

1.4. Product Spaces

Let n_1 and n_2 be two natural numbers. Since the product space of the affine space $\mathbf{A}^{n_1}$ and $\mathbf{A}^{n_2}$ is again an affine space with is isomorphic to $\mathbf{A}^{n_1+n_2}$, the algebraic set of $\mathbf{A}^{n_1} \times \mathbf{A}^{n_2}$ are defined by the algebraic sets of $\mathbf{A}^{n_1+n_2}$.

To define algebraic sets of $\mathbf{P}^{n_1} \times \mathbf{P}^{n_2}$ we have to introduce the notion of bihomogenous. In the following we will use $\mathbf{X}$ and $\mathbf{Y}$ as abbreviations for $x_0, \dots, x_{n_1}$ and $y_0, \dots, y_{n_2}$.

Definition 8. A polynomial $F(\mathbf{X}, \mathbf{Y}) \in k[\mathbf{X}, \mathbf{Y}]$ is called *bihomogenous in* $\mathbf{X}$ *and* $\mathbf{Y}$ if $F(\mathbf{X}, \mathbf{Y})$ is homogenous when considered as a polynomial in $\mathbf{X}$ with coefficients in $k[\mathbf{Y}]$ and $F(\mathbf{X}, \mathbf{Y})$ is homogenous when considered as a polynomial in $\mathbf{Y}$ with coefficients in $k[\mathbf{X}]$.

Definition 9. Let $F(\mathbf{X}, \mathbf{Y}) \in k[\mathbf{X}, \mathbf{Y}]$ be a bihomogenous polynomial. The *biprojective hypersurface defined by* $F(\mathbf{X}, \mathbf{Y})$, denoted by $V(F(\mathbf{X}, \mathbf{Y}))$, is the set of points

$$V(F(\mathbf{X}, \mathbf{Y})) = \{(\mathbf{a}, \mathbf{b}) \in \mathbf{P}^{n_1} \times \mathbf{P}^{n_2} \mid F(\mathbf{a}, \mathbf{b}) = 0\}.$$

For a set S of bihomogenous polynomials in $k[\mathbf{X}, \mathbf{Y}]$ the *biprojective algebraic set defined by* S, written as $V(S)$, is defined as

$$V(S) = \bigcap_{F(\mathbf{X}, \mathbf{Y}) \in S} V(F(\mathbf{X}, \mathbf{Y})).$$

Algebraic sets of multiprojective spaces like $\mathbf{P}^{n_1} \times \dots \times \mathbf{P}^{n_r}$ and mixed spaces like $\mathbf{A}^{n_1} \times \mathbf{P}^{n_2}$ are defined likewise.

Definition 10. Let Y be a product space of the form $Y = \mathbf{A}^{n_1} \times \mathbf{P}^{n_2} \times \dots \times \mathbf{P}^{n_r}$. An algebraic set $X \subset Y$ is called *reducible* if there exist algebraic sets X_1, $X_2 \subset Y$ both unequal to X, such that $X = X_1 \cup X_2$.

1.5. Topology

Let Y be a product space of the form $Y = \mathbf{A}^{n_1} \times \mathbf{P}^{n_2} \times \dots \times \mathbf{P}^{n_r}$. We define a topology on Y in the following way.

Definition 11. The *Zariski topology on* Y is defined by: A set $X \subset Y$ is *open* if $Y \setminus X$ is an algebraic set.

Definition 12. Let $x \in Y$ and $X \subset Y$. Any open set $U \ni x$ is called a *neighborhood of* x. The intersection of all closed subsets of Y containing X is called the *Zariski-closure of* X and denoted by $\overline{X}$, i.e. $\overline{X} = \bigcap_{X' \supset X \text{ closed}} X'$. We say that X is *dense in* Y if $\overline{X} = Y$.

2. Local Geometry of Curves

From now on we will mainly deal with algebraic (and analytic) sets of the affine plane. For this reason we adapt the general notation of Section 1 to the case of two dimensions. Whenever possible and suitable we will use the notation shown in the following table.

$f(x,y),\ g(x,y) \in k[x,y]$	Polynomials (defining curves).
$A(t),\ B(t) \in k((t))$	Meromorphic series (defining curve-branches).
$\mathfrak{A}(t),\ \mathfrak{B}(t) \in k\langle\langle t\rangle\rangle$	Puiseux series.
$p,\ q \in \mathbf{A}^2$	Points in the affine plane.
$\mathfrak{z},\ \mathfrak{z}' \subset \mathbf{A}^2$	Curve-branches in the affine plane.
$\mathcal{C},\ \mathcal{D} \subset \mathbf{A}^2$	Curves in the affine plane.

TABLE 2. Notation for Affine Curves

2.1. Curve-branches

Definition 13. A *local parametrization* $\delta(t)$ is a pair $\delta(t) = (A(t), B(t))$, with $A(t),\ B(t) \in k((t))$ and not both constant. If $A(t) \in k[[t]]$ we say that $\delta(t)$ is centered over the affine point $x = A(0)$, otherwise we say that $\delta(t)$ is centered over $x = \infty$. If also $B(t) \in k[[t]]$ then $\delta(t)$ is called an *affine local parametrization* and we define its *center* q as $q = (A(0), B(0)) \in \mathbf{A}^2$.

A local parametrization $\delta(t)$ is called *redundant* if there exists another local parametrization $\delta'(t)$ and a power series $T'(t) \in k[[t]]$ with $\mathrm{ord}(T'(t)) > 1$ such that $\delta(t) = \delta'(T'(t))$. A local parametrization is called *irredundant* if it is not redundant.

Two local parametrizations $\delta(t)$ and $\delta'(t)$ are called *equivalent* if $\delta(t) = \delta'(T'(t))$, for some power series $T'(t) \in k[[t]]$ with $\mathrm{ord}(T'(t)) = 1$.

Definition 14. Let $\delta(t) = (A(t), B(t))$ be a local parametrization. The *curve-branch* $\mathfrak{z}$ *defined by* $\delta(t)$ is the set of points

$$\mathfrak{z} = \{(A(t), B(t)) \mid t \in k\}.$$

The curve-branch $\mathfrak{z}$ is called an *affine curve-branch* if $\delta(t)$ is an affine local parametrization. The *center of* $\mathfrak{z}$ is defined as the center of $\delta(t)$.

Let $\mathcal{C}$ be a curve defined by $f(x,y) \in k[x,y]$. We say that $\mathfrak{z}$ is a *curve-branch of* $\mathcal{C}$, written as $\mathfrak{z} \subseteq \mathcal{C}$, if $f(A(t), B(t)) = 0$.

Obviously, equivalent local parametrizations have the same center and define the same curve-branch.

Definition 15. Let $\delta(t) = (A(t), B(t))$ be a local irredundant parametrization of a curve-branch $\mathfrak{z}$. We say that the curve-branch $\mathfrak{z}$ is *linear (or regular)* if $\min(|\mathrm{ord}(A(t) - A(0))|, |\mathrm{ord}(B(t) - B(0))|) = 1$.

Definition 16. A curve C is *locally reducible at* q if there is more than one curve-branch of C with center q. Otherwise C is called *locally irreducible at* q.

2.2. Newton's Theorem

We now show, how the curve-branches of the curve defined by a polynomial $f(x, y) \in k[x, y]$ over $x = 0$ are related to the factors of $f(x, y)$ in $k((x))[y]$. To do so, we start with a fundamental factorization-theorem which was first proved by Newton (see [**Newton, 1660**]) and later revived by Puiseux (see [**Puiseux, 1850**]).

Theorem 1 (Newton's Theorem). *Let* $f(x, y) \in k((x))[y]$ *be a monic polynomial in* y *of* y*-degree* $N > 0$*, i.e.* $f(x, y)$ *can be written as*

$$f(x, y) = y^N + A_{N-1}(x)y^{N-1} + \cdots + A_1(x)y + A_0(x),$$

where $A_i(x) \in k((x))$*, for* $0 \leq i < N$*. Then there exists a positive integer* e *, such that* $f(x, y)$ *can be written as*

$$f(t^e, y) = \prod_{j=0}^{N-1}(y - B_j(t)),$$

where $B_j(t) \in k((t))$*, for* $0 \leq j < N$.

Proof. There are several ways to prove Newton's Theorem. Algebraic proofs can be found in [**Abhyankar, 1990**], [**Walker, 1978**], and [**Keller, 1974**].

An analytic proof is given e.g. in [**Brieskorn and Knörrer, 1986**]. □

Two supplementals to Newton's Theorem are also of interest to us.

- In the case that $A_i(x) \in k[[x]]$, for $0 \leq i < N$, it holds that $B_j(t) \in k[[t]]$, for $0 \leq j < N$.
- If $f(x, y)$ is irreducible in $k((x))[y]$ then the smallest possible value for e is N and the meromorphic series $B_0(t), \ldots, B_{N-1}(t)$ can be labeled in such a way that $B_j(t) = B_0(\xi^j t)$, for $0 \leq j < N$, where ξ is a primitive N-th root of unity.

Definition 17. We use the notation of Theorem 1. The series $\mathfrak{B}_j(t) = B_j(t^{1/e}) \in k((t^{1/e})) \subset k\langle\langle t\rangle\rangle$, for $0 \leq j < N$, are called *Puiseux series of* $f(x, y)$ *over* $x = 0$. The smallest possible positive integer e' such that $\mathfrak{B}_j(t) \in k((t^{1/e'}))$ is called the *index of* $\mathfrak{B}_j(t)$.

Using Puiseux series we can reformulate Theorem 1 as: Every polynomial $f(x, y) \in k((x))[y]$ monic in y and of y-degree $N > 0$ can be written as $f(x, y) = \prod_{j=0}^{N-1}(y - \mathfrak{B}_j(x))$, where $\mathfrak{B}_j(x) \in k\langle\langle x\rangle\rangle$, for $0 \leq j < N$.

So we can view the series $\mathfrak{B}_j(x)$ as roots in $k\langle\langle x\rangle\rangle$ of the univariate polynomial $f(x, y)$ over $k((x))$, which was the original motivation of Newton for this theorem.

Definition 18. Let $f(x, y) \in k((x))[y]$ be monic in y and of y-degree $N > 0$. Further, let $\mathfrak{B}(t) \in k\langle\langle t\rangle\rangle$ be a Puiseux series of $f(x, y)$ over $x = 0$. The pair $(t^e, \mathfrak{B}(t^e))$, where e is the index of $\mathfrak{B}(x)$, is called a *function-branch of* $f(x, y)$ *over* $x = 0$.

For the case, where the field k is the field of complex numbers a surface, called the *Riemann-surface of* $f(x, y)$, can be constructed such that there is a continuous one-to-one correspondence between the function branches of $f(x, y)$ and the places of the Riemann-surface of $f(x, y)$. For further details we refer to [**Bliss, 1933**, Chapter IV].

We now turn to the situation where $f(x, y) \in k[x, y]$ is no longer necessarily monic in y. What we still require is that $f(x, y)$ is primitive when viewed as a polynomial in y with coefficients in $k[x]$.

Let $f(x, y)$ now be given as

$$f(x, y) = A_N(x)y^N + A_{N-1}(x)y^{N-1} + \cdots + A_1(x)y + A_0(x),$$

where $A_i(x) \in k[x]$, for $0 \leq i \leq N$. If we divide $f(x, y)$ by $A_N(x)$ we get a monic polynomial $g(x, y)$ in y whose coefficients are rational functions in x, i.e. $g(x, y) = f(x, y)/A_N(x) \in k(x)[y]$. The polynomials $f(x, y)$ and $g(x, y)$ define the same curve and since $k(x) \subseteq k((x))$ Newton's Theorem can be applied to $g(x, y)$. So everything said above also holds for the case where $f(x, y)$ is not monic in y.

Summarizing the above, we get for a polynomial $f(x, y) \in k((x))[y]$, monic in y, according to Newton's Theorem the following factorizations.

$$f(x, y) = \prod_{i=1}^{I} f_i^{d_i}(x, y),$$

where $f_i(x, y) \in k((x))[y]$ irreducible and of y-degree e_i, for $1 \leq i \leq I$, and

$$f_i(t^{e_i}, y) = \prod_{j=0}^{e_i-1} (y - B_i(\xi_i^j t)),$$

where $B_i(t) \in k((t))$ and ξ_i a primitive e_i-th root of unity, for $1 \leq i \leq I$. Obviously it holds that $\sum_{1 \leq i \leq I} d_i e_i = N$.

Viewing the pairs $(t^{e_i}, B_i(\xi_i^j t))$ as local parametrizations $\delta_{i,j}(t)$ centered over $x = 0$, we see that all $\delta_{i,j}(t)$, for $0 \leq j < e_i$, are equivalent and hence define the same curve-branch which we denote by $\mathfrak{z}_i$. By definition $\mathfrak{z}_i$ is a curve-branch of $\mathcal{C}$ over $x = 0$ which consists of the e_i function-branches defined by $\delta_{i,j}(t)$, for $1 \leq j \leq e_i$. This leads us to the following definition.

Definition 19. Let e be the uniquely determined positive integer associated to a curve-branch $\mathfrak{z}$ as shown above. We define e as the *ramification index of* $\mathfrak{z}$.

The set $\mathbf{I} = \{1, \dots I\}$ can be partitioned into subsets $\mathbf{I}_k$, for $0 \le k \le K$, as follows: Let $i, i' \in \mathbf{I}$. If $\operatorname{ord}(B_i(t)) < 0$ then $i \in \mathbf{I}_0$, otherwise i and i' belong to the same subset $\mathbf{I}_k$, $k > 0$, if and only if $B_i(0) = B_{i'}(0)$. We assume that K is chosen in such a way that the sets $\mathbf{I}_k$, for $1 \le k \le K$, are non-empty and we denote their common value of $B_i(0)$ by η_k.

The points $(0, \eta_k)$, $1 \le k \le K$, are exactly all the distinct points of $\mathcal{C}$ over $x = 0$ and there are $|\mathbf{I}_k|$ different curve-branches of $\mathcal{C}$ centered at $(0, \eta_k)$. If $|\mathbf{I}_0| > 0$ the curve $\mathcal{C}$ has a point at infinity over $x = 0$ and again $|\mathbf{I}_0|$ is the number of different curve-branches.

In particular, if $(0,0) \in \mathcal{C}$ there must be a k such that $\eta_k = 0$. W.l.o.g. we assume that $\eta_1 = 0$. The curve-branches of $\mathcal{C}$ centered at $(0,0)$ are therefore given by $(t^{e_i}, B_i(t))$, for $i \in \mathbf{I}_1$.

A more direct way (without going from $k[[x]][y]$ to $k((x))[y]$) to describe all those curve-branches of $\mathcal{C}$ which are centered at some specific point is provided by the Preparation Theorem of Weierstrass.

Definition 20. Given a power series $f(x,y) \in k[[x,y]]$. We say that $f(x,y)$ is y-*general of order* m if $m = \operatorname{ord}(f(0,y))$.

Theorem 2 (Weierstrass Preparation Theorem). *Let* $f(x,y) \in k[[x,y]]$ *be* y-*general of order* $m < \infty$. *Then there exists a unique power series* $g(x,y) \in k[[x,y]]$ *and a unique polynomial* $h(x,y) \in k[[x]][y]$ *such that* $f(x,y) = g(x,y)h(x,y)$ *and*

$$
\begin{aligned}
&g(0,0) \neq 0,\\
&h(0,0) = 0,\\
&h(x,y) = y^m + A_{m-1}(x)y^{m-1} + \cdots + A_1(x)y + A_0(x),
\end{aligned}
$$

where $A_i(x) \in k[[x]]$, *for* $0 \le i < m$.

Proof. For a proof we again refer to [**Abhyankar, 1990**]. $\square$

The curve-branches of the curve defined by $f(x,y)$ centered at the origin are exactly defined by $h(x,y)$ which is monic in y and of y-degree m. The Preparation Theorem of Weierstrass does not give us detailed information about the curve-branches at the origin, it rather puts us in a situation where we directly can apply Newton's Theorem no matter whether $f(x,y)$ is monic in y or not (see also Chapter 4 Section 1 in which the Newton–Puiseux algorithm is described).

2.3. Puiseux Parametrizations

For this subsection let $f(x, y) \in k[x, y]$ be a polynomial, $\mathcal{C}$ the curve defined by $f(x, y)$, and $\mathfrak{z}$ an affine curve-branch of $\mathcal{C}$.

It is clear from the definition of curve-branches that there are infinitely many different local parametrizations for $\mathfrak{z}$. In the following define Puiseux parametrizations which can be seen as a kind of normal-form local parametrizations for affine curve-branches.

Definition 21. A local irredundant parametrization $\delta(t)$ of $\mathfrak{z}$ is called a *Puiseux parametrization of* $\mathfrak{z}$ if $\delta(t)$ is of the form

$$\delta(t) = (a_0 + a_e t^e, B(t)),$$

where e is a positive integer and $B(t) \in k[[t]]$. We say that $\delta(t)$ is a *classical Puiseux parametrization* if $a_e = 1$.

A consequence of Newton's Theorem and the Preparation Theorem of Weierstrass is that every curve-branch centered over some finite point of a curve has a classical Puiseux parametrization.

Definition 22. Let $\delta(t) = (a_0 + a_e t^e, B(t))$ be a Puiseux parametrization of the curve-branch $\mathfrak{z}$. The curve-branch $\mathfrak{z}$ is said to be in *general position* if $e \leq \operatorname{ord}(B(t) - B(0))$.

Now we define two equivalent invariants of curve-branches, namely k-sequences and characteristic pairs.

Definition 23. Let $\mathfrak{z}$ be a curve-branch and $\delta(t) = (a_0 + a_e t^e, B(t))$ a Puiseux parametrization of $\mathfrak{z}$. We assume w.l.o.g. that $\mathfrak{z}$ is in general position.

If $\mathfrak{z}$ is linear then no characteristic exponents are defined. Otherwise, let $k_0 = d_0 = e$ and $S_k = \{k' \mid k' \leq k \text{ and } b_{k'} \neq 0\}$. The *first characteristic exponent* k_1 *of* $\delta(t)$ is defined as

$$k_1 = \min\left(k \mid \gcd(\{d_0\} \cup S_k) < d_0\right).$$

Let $d_1 = \gcd(\{d_0\} \cup S_{k_1})$. For $i > 1$, the *i.th characteristic exponent* k_i *of* $\delta(t)$ is defined recursively by

$$k_i = \min\left(k \mid \gcd(\{d_{i-1}\} \cup S_k) < d_{i-1}\right),$$

where $d_i = \gcd(\{d_{i-1}\} \cup S_{k_i})$.

By I we denote the total number of characteristic exponents of $\delta(t)$. The k-*sequence of* $\delta(t)$ is then defined as $(k_i)_{0 \leq i \leq I}$. The *singular part of* $\delta(t)$

is defined as

$$(a_0 + a_e t^e, \sum_{k=0}^{k_I} b_k t^k).$$

As one can show, equivalent Puiseux parametrizations determine the same k-sequences. Therefore k-sequences are well defined for curve-branches (see [**Brieskorn and Knörrer, 1986**] for more details).

Definition 24. The *characteristic pairs* $(a_i, b_i)_{1 \le i \le I}$ *of a curve-branch* $\mathfrak{z}$ with k-sequence $(k_i)_{0 \le i \le I}$ are defined as

$$a_i = \frac{c_i}{d_{i+1}} \text{ and } b_i = \frac{d_i}{d_{i+1}},$$

where $c_1 = k_1$, $c_i = k_i - k_{i-1}$, for $2 \le i \le I$, and $d_i = \gcd(k_0, \ldots, k_{i-1})$.

For further details about characteristic pairs and other invariants of curve-branches we refer to the book of [**Brieskorn and Knörrer, 1986**], [**Abhyankar, 1967**], or [**Abhyankar, 1990**, Lecture 21]

It would also have been possible to first define characteristic pairs and then k-sequences of curve-branches. For this, let d_i, for $1 \le i \le I$, be defined recursively by $d_i = b_i d_{i+1}$, starting by $d_{I+1} = 1$. The k-sequence is then given by $k_0 = d_1$, $k_1 = a_1 d_2$, $k_i = k_{i-1} + a_i d_{i+1}$, for $2 \le i \le I$.

2.4. Intersection Multiplicities

Definition 25. Let $\mathfrak{z}$, $\mathfrak{z}'$ be two curve-branches with center q and classical Puiseux parametrizations $(a_0 + t^m, B(t))$ and $(c_0 + t^n, D(t))$. We define the *intersection multiplicity* $\mathrm{mult}_q(\mathfrak{z}, \mathfrak{z}')$ *of* $\mathfrak{z}$ *and* $\mathfrak{z}'$ *at* q as

$$\mathrm{mult}_q(\mathfrak{z}, \mathfrak{z}') = \mathrm{ord}\left(\prod_{i=1}^{n}\left(B(t) - D(\zeta^i t^{m/n})\right)\right),$$

where ζ is a primitive n-th root of unity.

From Definition 25 it is not obvious that the intersection multiplicity of two curve-branches is symmetric. But as shown in [**Abhyankar, 1990**, Lecture 17] the intersection multiplicity can also be defined using the Preparation Theorem of Weierstrass and resultants of power-series. In that case symmetry is obvious.

The next Lemma provides an easy way to compute the intersection multiplicity of two curve-branches.

Lemma 1. *Let* $\mathfrak{z}$*,* $\mathfrak{z}'$ *be two different curve-branches in general position both centered at* $q = (0, 0)$ *with classical Puiseux parametrizations* $(t^m, B(t))$ *and* $(t^n, D(t))$*.*

The power series $B(t)$ *and* $D(t)$ *can be written as*

$$B(t) = b_0 t^{m_0} + b_1 t^{m_0+m_1} + \cdots + b_s t^{m_0+m_1+\cdots+m_s} + \cdots,$$
$$D(t) = d_0 t^{n_0} + d_1 t^{n_0+n_1} + \cdots + d_s t^{n_0+n_1+\cdots+n_s} + \cdots,$$

where $m_0 = m$, $n_0 = n$ *and* b_i, $d_i \neq 0$ *and* m_i, $n_i > 0$, *for* $i \geq 1$.

Further let s *be defined as the number for which*

$$b_i = d_i \quad \textit{and} \quad \frac{m_i}{m_0} = \frac{n_i}{n_0},$$

for $0 \leq i < s$, *and*

$$a_s \neq b_s \quad \textit{or} \quad \frac{m_s}{m_0} \neq \frac{n_s}{n_0}$$

holds.

Then the intersection multiplicity of $\mathfrak{z}$ *and* $\mathfrak{z}'$ *at* q *is equal to*

$$\mathrm{mult}_q(\mathfrak{z}, \mathfrak{z}') = \min(\lambda_1, \lambda_2),$$

where

$$\lambda_1 = n_0 \left(m_0 + m_1 + \frac{m_2}{l_1} + \frac{m_3}{l_1 l_2} + \cdots + \frac{m_s}{l_1 \cdots l_{s-1}} \right),$$
$$\lambda_2 = m_0 \left(n_0 + n_1 + \frac{n_2}{l_1} + \frac{n_3}{l_1 l_2} + \cdots + \frac{n_s}{l_1 \cdots l_{s-1}} \right),$$

and the l_i *are recursively defined by*

$$\begin{array}{ll} \dfrac{m_1}{m_0} = \dfrac{n_1}{n_0} = \dfrac{k_1}{l_1} & , \ \gcd(k_1, l_1) = 1, \\ \dfrac{m_2}{m_0} = \dfrac{n_2}{n_0} = \dfrac{k_2}{l_1 l_2} & , \ \gcd(k_1, l_2) = 1, \\ \vdots & \quad \vdots \\ \dfrac{m_s}{m_0} = \dfrac{n_s}{n_0} = \dfrac{k_s}{l_1 \cdots l_s} & , \ \gcd(k_s, l_s) = 1. \end{array}$$

Proof. For a proof we refer to [**van der Waerden, 1973**, Paragraph 54]. □

Based on the definition of intersection multiplicity of two curve-branches we define the intersection multiplicity of a curve-branch and a curve, and the intersection multiplicity of two curves.

Definition 26. Let $\mathcal{C}$, $\mathcal{D}$ be two curves and $\mathfrak{z}$ a curve-branch with center q. The *intersection multiplicity* $\mathrm{mult}_q(\mathfrak{z}, \mathcal{C})$ *of* $\mathfrak{z}$ *and* $\mathcal{C}$ *at* q is defined as

$$\mathrm{mult}_q(\mathfrak{z}, \mathcal{C}) = \sum_{\mathfrak{z}' \subseteq \mathcal{C} \text{ centered at } q} \mathrm{mult}_q(\mathfrak{z}, \mathfrak{z}')$$

and the *intersection multiplicity* $\mathrm{mult}_q(\mathcal{C}, \mathcal{D})$ *of* $\mathcal{C}$ *and* $\mathcal{D}$ *at* q is defined as

$$\mathrm{mult}_q(\mathcal{C}, \mathcal{D}) = \sum_{\mathfrak{z}' \subseteq \mathcal{C} \text{ centered at } q} \mathrm{mult}_q(\mathfrak{z}', \mathcal{D}).$$

Lemma 2. *Given a curve* $\mathcal{C}$ *defined by* $f(x, y) \in k[x, y]$ *and a curve-branch* $\mathfrak{z}$ *with center* q *defined by* $(A(t), B(t))$, *where* $A(t)$, $B(t) \in k[[t]]$.

The intersection multiplicity $\mathrm{mult}_q(\mathfrak{z}, \mathcal{C})$ *of* $\mathfrak{z}$ *and* $\mathcal{C}$ *at* q *is equal to*

$$\mathrm{mult}_q(\mathfrak{z}, \mathcal{C}) = \mathrm{ord}(f(A(t), B(t))).$$

Proof. A proof can be found in [**Brieskorn and Knörrer, 1986**]. □

In addition to the previously defined k-sequences and characteristic pairs and now want to introduce another invariant of a curve-branch, namely a semigroup of natural numbers generated by intersection multiplicities.

Definition 27. Let $\mathfrak{z}$ be a curve-branch defined by $(A(t), B(t))$, where $A(t)$, $B(t) \in k[[t]]$. We denote its center by q. The *value (intersection) semigroup* $\Gamma_{\mathfrak{z}}$ *of* $\mathfrak{z}$ is defined as

$$\begin{aligned}\Gamma_{\mathfrak{z}} &= \{\mathrm{mult}_q(\mathfrak{z}, \mathcal{C}) \mid \mathcal{C} \text{ a curve and } \mathfrak{z} \not\subseteq \mathcal{C}\} \\ &= \{\mathrm{ord}(f(A(t), B(t))) \mid f(x, y) \in k[x, y],\ f(A(t), B(t)) \neq 0\}.\end{aligned}$$

2.5. Singular Points

Definition 28. For a curve-branch $\mathfrak{z}$ with center q the *multiplicity of* q *w.r.t.* $\mathfrak{z}$, written as $\mathrm{mult}_q(\mathfrak{z})$, is defined to be

$$\mathrm{mult}_q(\mathfrak{z}) = \min\left(\mathrm{mult}_q(\mathfrak{z}, \mathcal{L}) \mid \mathcal{L} \text{ a line through } q\right).$$

For a curve $\mathcal{C}$ the *multiplicity of* q *w.r.t.* $\mathcal{C}$ is defined as

$$\mathrm{mult}_q(\mathcal{C}) = \sum_{\mathfrak{z}' \subseteq \mathcal{C} \text{ centered at } q} \mathrm{mult}_q(\mathfrak{z}').$$

Definition 29. A line $\mathcal{L}$ is called a *tangent of a curve* $\mathcal{C}$ *at* q if $\mathrm{mult}_q(\mathcal{C}) < \mathrm{mult}_q(\mathcal{C}, \mathcal{L})$.

A direct consequence of the above definitions is, that curve-branches have unique tangents at their centers, which allows us to define the tangent of a curve-branch. Obviously, a curve-branch is in general position if and only if its tangent is not parallel to the y-axis.

Definition 30. Let $\mathcal{C}$ be a curve. All points of $\mathcal{C}$ with multiplicity greater than one are called *singular points*. Non-singular points of $\mathcal{C}$ are also called *regular* or *simple points*. If $\mathcal{C}$ has no singular points then $\mathcal{C}$ is called *regular* or *simple*.

A singular point q with multiplicity m is called an *ordinary singular point* if there are m different tangents to $\mathcal{C}$ at q, otherwise q is called a *non-ordinary singular point*.

Given a curve $\mathcal{C}$ defined by $f(x,y) \in k[x,y]$ and a point $q = (q_x, q_y) \in \mathcal{C}$. We can write $f(x,y)$ as $f(x,y) = \sum_{i,j} a_{i,j}(x-q_x)^i(y-q_y)^j$, with $a_{i,j} \in k$. It then holds that

$$\begin{aligned}\mathrm{mult}_q(\mathcal{C}) &= \min\left(i+j \mid a_{i,j} \neq 0\right) \\ &= \min\left(i+j \mid \frac{\partial^{i+j}}{\partial x^i \partial y^j} f(x,y)|_q \neq 0\right).\end{aligned}$$

Hence, the set of all singular points of $\mathcal{C}$ can alternatively be defined as the set of common zeros of $f(x,y) = \frac{\partial}{\partial x} f(x,y) = \frac{\partial}{\partial y} f(x,y) = 0$. If $f(x,y)$ is square-free the set of singular points of $\mathcal{C}$ is therefore finite.

CHAPTER 3

Curve Singularities

In the first section of this chapter we define a special type of birational maps, namely affine blowups or quadratic transformations. As one can show, any algebraic curve can be transformed by a finite number of such maps into a birational equivalent curve which is simple, see Section 2. In Section 3 we use this result to investigate and classify the singularities of algebraic curves.

For further details about birational maps and resolution of singularities we refer to [**Brieskorn and Knörrer, 1986**], [**Shafarevich, 1994**], [**Mumford, 1976**], and [**Hartshorne, 1977**].

1. Affine Blowups

Let $X = \mathbf{A}^2$ with affine coordinates (x_0, y_0) and let $T = \mathbf{P}^1$ with homogeneous coordinates (u_0, v_0). We define the irreducible algebraic set $X' \subset X \times T$ as

$$X' = \{((x_0, y_0), (u_0, v_0)) \in X \times T \mid x_0 v_0 = y_0 u_0\}.$$

Definition 31. The map $\sigma\colon X' \to X$ defined by restricting the first projection $X \times T \to X$ to X' is called the *affine blowup or quadratic transformation (QT) of* X *centered at the origin.*

There are several other names in literature which are used for σ, e.g. monoidal transformation, Hopf-map, dilatation, or simply σ-process.

In Figure 1 (left part) we try to illustrate the algebraic set X', where T is replaced by a segment of the affine line, so that we get a picture of X' as a surface in $\mathbf{A}^2 \times \mathbf{A}^1$.

Since we will also need to blowup points different from the origin, we define the blowup of X centered at an arbitrary point $q \in X$.

Definition 32. Let $q = (q_x, q_y) \in X$ and let $\phi_q\colon X \to X$ be the linear map defined by $(x_0, y_0) \mapsto (q_x + x_0, q_y + y_0)$. The map $\sigma_q = \phi_q \circ \sigma\colon X' \to X$ is called the *affine blowup of* X *centered at* q.

To simplify notation we assume for the following that $q = (0, 0) \in X$. As we can see from the above definition of σ_q, for any point $p = (p_x, p_y) \in X$

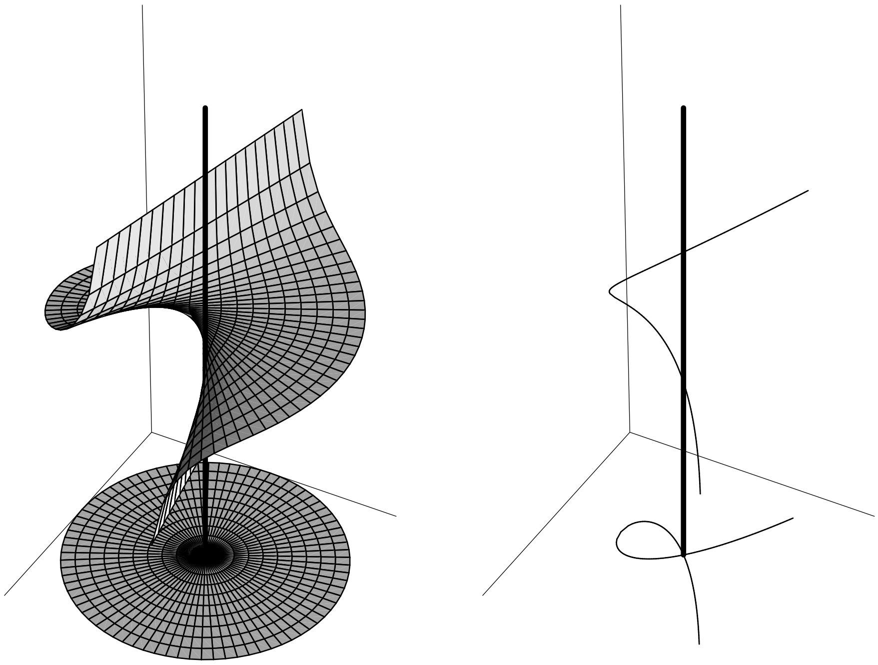


FIGURE 1. Affine blowup

different from q there is exactly one point $p' \in X'$ such that $\sigma_q(p') = p$, namely $p' = ((p_x, p_y), (p_x, p_y))$. Therefore the inverse of σ_q on $X \setminus q$ is defined by $\sigma_q^{-1}|_{X \setminus q} \colon X \setminus q \to X', (x_0, y_0) \mapsto ((x_0, y_0), (x_0, y_0))$, and σ_q defines an isomorphism between $X' \setminus (q \times T)$ and $X \setminus q$. However, if p is equal to q we have $\sigma_q^{-1}(p) = q \times T$.

For a more detailed understanding of the map $\sigma_q^{-1} \colon X \to X'$, we consider σ_q^{-1} over some line $\mathcal{L} \subset X$ defined by $f(x, y) = ax - by$. Obviously, σ_q^{-1} on $\mathcal{L}$ is given by $(x_0, y_0) \mapsto ((x_0, y_0), (b, a))$. Although σ_q^{-1} is not regular on X at q, its restriction to $\mathcal{L}$ is regular. The curve $\mathcal{L}$ is mapped to a curve $\mathcal{L}' = \sigma_q^{-1}|_{\mathcal{L}}(\mathcal{L}) \subset X'$ which intersects $q \times T$ in the point $(q, (b, a))$.

The numbers a and b in $f(x, y)$ correspond to the direction of $\mathcal{L}$. Thus we can set up a one-to-one correspondence between the directions of lines through q and the points of $(q \times L)$ (see also Figure 1).

For arbitrary curves the inverse images of σ_q can be characterized as follows.

Theorem 3. *Let $\mathcal{C} \subset X$ be a curve, $q \in \mathcal{C}$ a point on it, and $\sigma_q \colon X' \to X$ the blowup of X centered at q. The inverse image $\sigma_q^{-1}(\mathcal{C})$ of $\mathcal{C}$ is of the form*

$$\sigma_q^{-1}(\mathcal{C}) = (q \times T) \cup \mathcal{C}',$$

where $\mathcal{C}'$ *is the closure of* $\sigma_q^{-1}(\mathcal{C} \setminus q)$*, i.e.* $\mathcal{C}' = \overline{\sigma_q^{-1}(\mathcal{C} \setminus q)}$*. The curve* $\mathcal{C}'$ *is reducible if and only if* $\mathcal{C}$ *is reducible.*

The restriction of σ_q *to* $\mathcal{C}'$ *defines a regular map* $\sigma_q|_{\mathcal{C}'} \colon \mathcal{C}' \to \mathcal{C}$ *which is an isomorphism of some neighborhood* U *of* p *if* $p \neq q$*.*

Proof. The equality $\sigma_q^{-1}(\mathcal{C}) = (q \times T) \cup \mathcal{C}'$ follows by definition: If $p \in \mathcal{C} \setminus q$ then $\sigma_q^{-1}(p) \in \mathcal{C}'$, and $\sigma_q^{-1}(q) = (q \times T)$.

As we saw above, σ_q defines an isomorphism between $X' \setminus (q \times T)$ and $X \setminus q$. Hence, the curve $\mathcal{C}'$ is reducible if and only if $\mathcal{C}$ is reducible, and $\sigma_q|_{\mathcal{C}'}$ is regular. □

According to the Theorem above we make the following definitions.

Definition 33. Let $\mathcal{C} \subset X$ be a curve, $q \in \mathcal{C}$ a point on it, and $\sigma_q \colon X' \to X$ the blowup of X centered at q. The inverse image $\sigma_q^{-1}(\mathcal{C}) \subset X'$ is called the *total transform of* $\mathcal{C}$ *w.r.t.* σ_q and denoted by $\mathcal{C}^*$. From Theorem 3 we know that $\mathcal{C}^* = \mathcal{C}' \cup \mathcal{E}'$, where $\mathcal{C}' = \overline{\sigma_q^{-1}(\mathcal{C} \setminus q)}$ and $\mathcal{E}' = \sigma_q^{-1}(q)$. The curve $\mathcal{C}'$ is called the *proper transform of* $\mathcal{C}$ *w.r.t.* σ_q and $\mathcal{E}'$ is called the *exceptional line of* σ_q or the *exceptional divisor of* σ_q.

The elements of the non-empty set of points $\mathcal{C}' \cap \mathcal{E}'$ are called *points in the first neighborhood of* q or *first neighboring points of* q.

To be able to apply further blowups to a transformed curve, there are two natural possibilities: First, we could extend our definition of affine blowups to blowups defined for arbitrary algebraic sets as done e.g. in [**Mumford, 1976**], [**Shafarevich, 1994**]. Second, we could cover the variety X' by affine planes and map the total transform to these affine planes. So we would be again in a same situation where we could apply affine blowups (see e.g. [**Brieskorn and Knörrer, 1986**]). We choose the second possibility and briefly explain how this can be done.

Let $\sigma \colon X' \to X$ be the affine blowup of X centered at the origin. In the same way as the projective line can be covered by two affine lines, the variety X' can be covered by two subsets X'_1 and X'_2, both isomorphic to the affine plane, in the following way:

Let the two subsets be defined by

$$X'_1 = \{((x_0, y_0), (u_0, v_0)) \in X' \mid u_0 \neq 0\},$$
$$X'_2 = \{((x_0, y_0), (u_0, v_0)) \in X' \mid v_0 \neq 0\}.$$

Of course it holds that $X' = X'_1 \cup X'_2$. From the definition of X' we get that

$$X'_1 = \{((x_0, x_0 v_0), (1, v_0)) \mid (x_0, v_0) \in X\},$$
$$X'_2 = \{((y_0 u_0, y_0), (u_0, 1)) \mid (y_0, u_0) \in X\}.$$

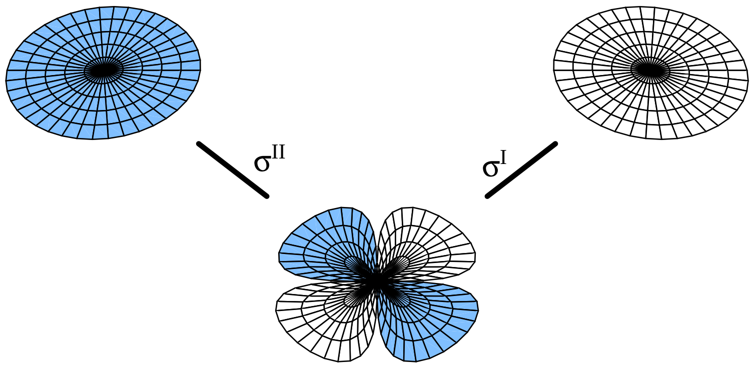


FIGURE 2. Decomposition of σ into σ^{I} and σ^{II}

So we see that X_1' and X_2' are both isomorphic to X. Introducing local coordinates for X_1' and X_2'

$$X_1' : (x_1, y_1) = (x_0, v_0),$$
$$X_2' : (x_2, y_2) = (y_0, u_0),$$

the subspaces are related to each other by

$$x_2 = x_1 y_1,$$
$$y_2 = y_1^{-1}.$$

Applying the above covering of X' to the blowup $\sigma \colon X' \to X$, we can write σ in local coordinates of X_1 and X_2 as

$$\sigma^{I} = \sigma|_{X_1'} : X_1' \to X, (x_1, y_1) \mapsto (x_1, x_1 y_1)$$

and

$$\sigma^{II} = \sigma|_{X_2'} : X_2' \to X, (x_2, y_2) \mapsto (x_2 y_2, y_2)$$

respectively. For an illustration of this covering see Figure 2.

Example 1. Applying this decomposition to the blowup of the curve defined by $y^2 - x^3 - x^2$ yields $x^2(y^2 - x - 1)$ and $y^2(1 - x^3 y - x^2)$. In Figure 3 pictures of these affine curves are shown (see also Figure 1 for a "full" picture of the blowup).

Of course, the above decomposition of σ into σ^{I} and σ^{II} can also be applied to an affine blowup centered at q, for $q \in \mathbf{A}^2$. Therefore, from now

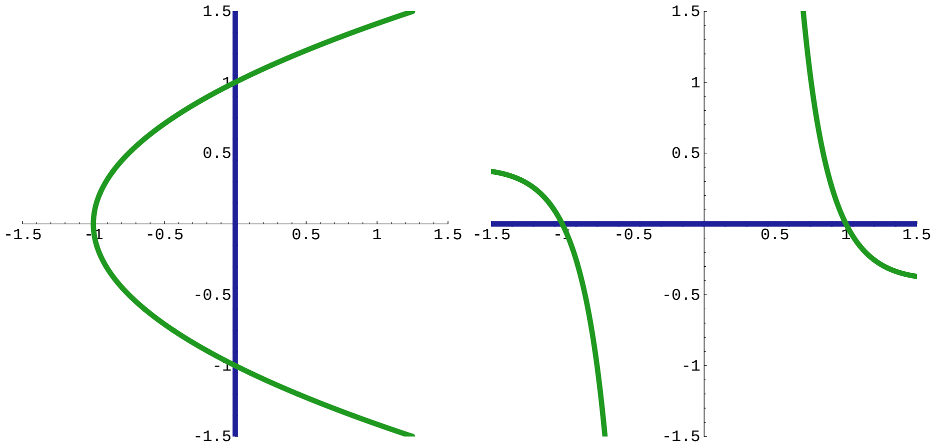


FIGURE 3. Decomposition of the blowup of $y^2 - x^3 - x^2$

on we will view affine blowups $\sigma_q\colon X' \to X$ as being the result of combining the two maps $\sigma_q^{\mathrm{I}}\colon X_1' \to X$ and $\sigma_q^{\mathrm{II}}\colon X_2' \to X$.

Definition 34. The maps $\sigma_q^{\mathrm{I}}\colon X_1' \to X$ and $\sigma_q^{\mathrm{II}}\colon X_2' \to X$ are called *affine blowup (or QT) of type* I and *type* II respectively.

Let $\mathcal{C}$ be the curve defined by $f(x,y) \in k[x,y]$. If D and m denote the degree and order of $f(x,y)$ we can write $f(x,y)$ as $f(x,y) = \sum_{d=m}^{D} f_d(x,y)$, where $f_d(x,y)$ are homogenous polynomials of degree d.

Applying a QT of type I (centered at the origin) to $\mathcal{C}$ we get

$$\begin{aligned}(f\circ\sigma^{\mathrm{I}})(x,y) &= f(x_1, x_1y_1)\\ &= \sum_{d=m}^{D} f_d(x_1, x_1y_1) = \sum_{d=m}^{D} x_1^d f_d(1,y_1) = x_1^m \sum_{d=m}^{D} x_1^{d-m} f_d(1,y_1)\\ &= x_1^m f^{\mathrm{I}}(x_1,y_1),\end{aligned}$$

where x_1^m and $f^{\mathrm{I}}(x_1,y_1)$ are the defining polynomials of the exceptional line and the proper transform of $\mathcal{C}$. We also see that the coordinates of the first neighboring points are given by $(0,\eta)$, η a root of $f_m(1,y_1)$. For QTs of type II the exceptional line is given by y_2^m and the coordinates of the first neighboring points are of the form $(\eta, 0)$, η a root of $f_m(x_2, 1)$.

Since $f_m(x,y)$ is homogenous it can be factored as $f_m(x,y) = \prod_{i=1}^{h}(a_ix + b_iy)^{e_i}$, with $\sum_{i=1}^{h} e_i = m$. So we see once more, that the coordinates of the first neighboring points are in one-to-one correspondence with the tangents of $\mathcal{C}$ at q, which are defined by $a_ix + b_iy$, for $1 \le i \le h$. But that means, that QTs of type I take into account all tangents different from the y-axis and QTs of type II all those different from the x-axis.

After applying an affine blowup of type I (of type II) to $\mathcal{C}$ exactly those neighboring points of q are missing in the affine plane which correspond to tangents defined by x, i.e. the y-axis (defined by y, i.e. the x-axis).

Let $\mathcal{C}$ be a curve and q a point on it. Using the decomposition of σ_q into σ_q^{I} and σ_q^{II} the multiplicity of a point q' in the first neighborhood of q is simply defined as the multiplicity of q' w.r.t. the proper transform of $\mathcal{C}$ in X_1' or X_2'.

Example 2. We consider the curve $\mathcal{C}$ defined by $f(x,y) = y^5 - 2x^2y^2 + 3x^4y - x^6$. The total transforms of $\mathcal{C}$ w.r.t. σ^{I} and σ^{II} are defined by

$$\mathcal{C}^{I} = (\sigma^{I})^{-1}(\mathcal{C}) : x^4(xy^5 - 2y^2 + 3xy - x^2),$$
$$\mathcal{C}^{II} = (\sigma^{II})^{-1}(\mathcal{C}) : y^4(-x^6y^2 + 3x^4y + y - 2x^2).$$

The origin is again a singular point of $\mathcal{C}^{I}$, so we apply a further blowup and get

$$\mathcal{C}^{I,I} = (\sigma^{I})^{-1}(\mathcal{C}_0) : x^6(x^4y^5 - 2y^2 + 3y - 1),$$
$$\mathcal{C}^{I,II} = (\sigma^{II})^{-1}(\mathcal{C}_0) : y^6x^4(xy^4 - x^2 + 3x - 2).$$

In Figure 4 affine real pictures of the above curves are shown. One can also see there the overlapping of $\mathcal{C}^{I,I}$ and $\mathcal{C}^{I,II}$.

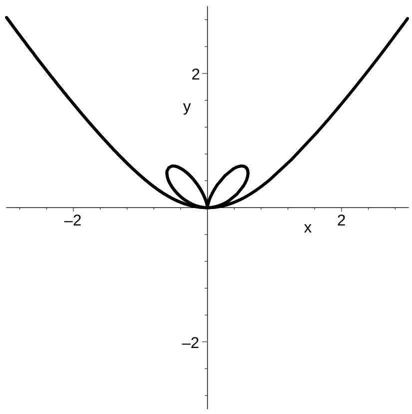
Until now, we have defined affine blowups only for curves. The definition for curve-branches is obtained in a straight-forward way. Let $\mathfrak{z}$ be a curve-branch (w.l.o.g. centered at the origin) given by $\delta(t) = (A(t), B(t))$, where $A(t)$, $B(t) \in k[[t]]$ and $a = \mathrm{ord}(A(t))$, $b = \mathrm{ord}(B(t))$.

First, we assume that $a < b$, i.e. $\mathfrak{z}$ is tangent to the x-axis. As mentioned before an affine blowup of type II would transform $\mathfrak{z}$ into a curve-branch which is no longer centered in the affine plane. For this reason we only allow blowups of type I in this case. On the other hand, for the case $a > b$, i.e. $\mathfrak{z}$ is tangent to the y-axis, we do not allow blowups of type I. If $a = b$ then both types of QTs will transform $\mathfrak{z}$ into a curve-branch which is centered in the affine plane.

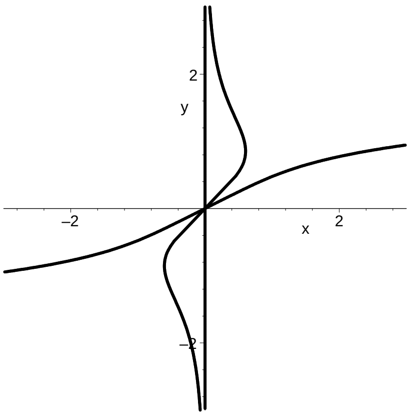
So we get for the exceptional lines and proper transforms of $\mathfrak{z}$ w.r.t. the QTs of type I and type II

$$x_1^{\min(a,b)} \text{ and } \delta^{I}(t) = (A(t), \frac{B(t)}{A(t)}), \quad \text{for } a \leq b,$$
$$y_1^{\min(a,b)} \text{ and } \delta^{II}(t) = (\frac{A(t)}{B(t)}, B(t)), \quad \text{for } a \geq b.$$

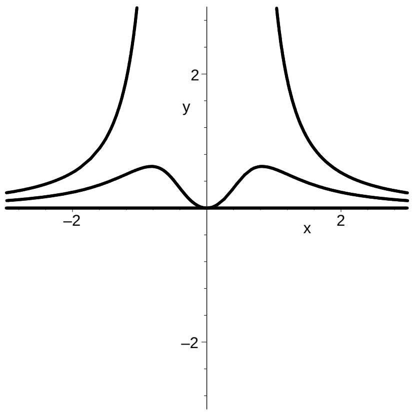
From the above it is clear that the proper transform of a curve-branch is again a curve-branch which intersects the exceptional divisor in exactly one point. Furthermore, this point may be unequal to the origin only in the case $a = b$.



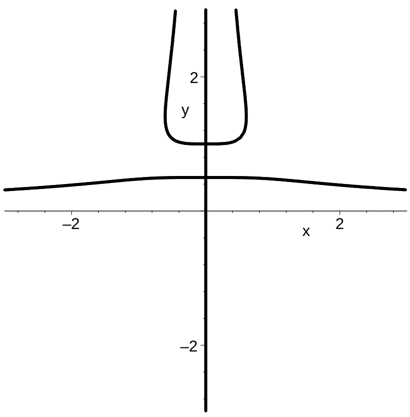
$$\mathcal{C} : y^5 - 2x^2y^2 + 3x^4y - x^6$$



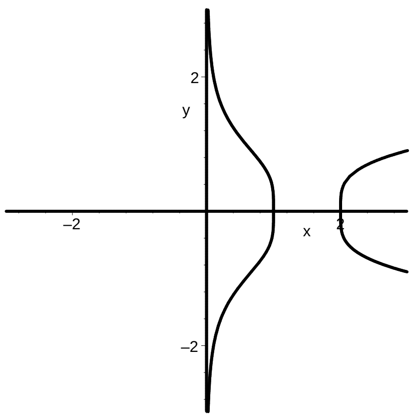
$$\mathcal{C}^{\mathrm{I}} : x^4(xy^5 - 2y^2 + 3xy - x^2)$$



$$\mathcal{C}^{\mathrm{II}} : y^4(-x^6y^2 + 3x^4y + y - 2x^2)$$



$$\mathcal{C}^{\mathrm{I,I}} : x^6(x^4y^5 - 2y^2 + 3y - 1)$$



$$\mathcal{C}^{\mathrm{I,II}} : y^6x^4(xy^4 - x^2 + 3x - 2)$$

FIGURE 4. Resolution of $y^5 - 2x^2y^2 + 3x^4y - x^6$

2. Resolution

The investigation and classification of curve singularities is based on the following fundamental theorems. For notational reasons let $X = \mathbf{A}^2$.

Definition 35. A *finite sequence of blowups* $\pi\colon X' \to X$ is a mapping defined by

$$\pi\colon X' = X_n \underset{\pi_n}{\longrightarrow} X_{n-1} \underset{\pi_{n-1}}{\longrightarrow} \cdots \underset{\pi_2}{\longrightarrow} X_1 \underset{\pi_1}{\longrightarrow} X_0 = X,$$

where each $\pi_i\colon X_i \to X_{i-1}$, for $1 \le i \le n$, is a finite composition of blowups centered at different points.

Definition 36. Let $\mathcal{C} \subset X$ be a curve and q a point on it. For a finite sequence of blowups $\pi\colon X' = X_n \to X_0 = X$, where $\pi = \pi_n \circ \cdots \circ \pi_1$ we define recursively:

$\mathcal{C}_i^*$ is the total transform $\mathcal{C}_{i-1}^*$ w.r.t. π_i, $\mathcal{C}_0^* = \mathcal{C}$,
$\mathcal{C}_i'$ is the proper transform of $\mathcal{C}_{i-1}'$ w.r.t. π_i, $\mathcal{C}_0' = \mathcal{C}$,
$\mathcal{E}_i'$ is the excetional divisor of $\pi_i \circ \cdots \circ \pi_1$.

The exceptional divisor $\mathcal{E}'$, total and proper transform $\mathcal{C}^*$ and $\mathcal{C}'$ of $\mathcal{C}$ w.r.t. π are defined as $\mathcal{E}' = \mathcal{E}_n'$, $\mathcal{C}^* = \mathcal{C}_n^*$, and $\mathcal{C}' = \mathcal{C}_n'$.

Theorem 4 (Resolution of Curves). *Let $\mathcal{C} \subset X$ be a plane algebraic curve devoid of multiple components. Then there exists a finite sequence of blowups $\pi\colon X' \to X$ such that the proper transform $\mathcal{C}'$ of $\mathcal{C}$ w.r.t. π has no singular points.*

Proof. See [**Mumford, 1976**] Theorem 8.4. □

Definition 37. The curve $\mathcal{C}'$ of Theorem 4 is called a *non-singular model of $\mathcal{C}$*.

Definition 38. Let $\mathcal{C}$ and $\mathcal{D}$ be curves intersecting at a point q. We say that $\mathcal{C}$ and $\mathcal{D}$ are *normal crossing at* q or *transveral at* q if $\mathrm{mult}_q(\mathcal{C}, \mathcal{D}) = \mathrm{mult}_q(\mathcal{C}) + \mathrm{mult}_q(\mathcal{D})$.

Theorem 5 (Embedded Resolution of Curves in Surfaces). *Let $\mathcal{C} \subset X$ be a plane algebraic curve devoid of multiple components. Then there exists a finite sequence of blowups $\pi\colon X' \to X$ such that the proper transform of $\mathcal{C}$ w.r.t. π is non-singular. Furthermore for every singular point $q \in \mathcal{C}$, $\pi^{-1}(q)$ is the union of non-singular curves, crossing only two at a time and then normal.*

Proof. For a proof we refer to [**Mumford, 1976**] Theorem 8.4 together with Corollary 8.7. □

Definition 39. Given a curve $\mathcal{C}$ and a point q on it. By the *standard resolution of* $\mathcal{C}$ *at* q we mean the finite sequence of blowups $\pi\colon X' = X_n \to X_0 = X, \pi = \pi_n \circ \cdots \circ \pi_1$, where $\pi_i\colon X_i \to X_{i-1}$ is defined as: π_1 is the blowup of q, π_i, for $2 \leq i \leq n$, is the composition of the blowups centered at the points $\mathcal{C}'_{i-1} \cap \mathcal{E}'_{i-1}$ which are still singular points on $\mathcal{C}'_{i-1}$ or $\mathcal{E}'_{i-1}$, or in which $\mathcal{C}'_{i-1}$ and $\mathcal{E}'_{i-1}$ do not meet transversally.

The points $\mathcal{C}'_i \cap \mathcal{E}'_i$ are called the *points in the* i*.th neighborhood of* q.

Applying Theorem 5 to one specific curve-branch $\mathfrak{z}$ we know that there exists a finite sequence of blowups $\pi = \pi_n \circ \cdots \circ \pi_1$ such that the proper transform of $\mathfrak{z}$ w.r.t. π is a linear curve-branch intersecting with only one exceptional divisor and this intersection is transversal. In this case the π_i consist of only one blowup.

Definition 40. Given a curve-branch $\mathfrak{z}_0$ with local parametrization $\delta_0(t) = (A_0(t), B_0(t))$. Let $a_0 = \text{ord}(A_0(t) - A_0(0))$ and $b_0 = \text{ord}(B_0(t), B_0(0))$. The *resolution sequence* $(\sigma_h)_{1\leq h}$ *of* $\mathfrak{z}_0$ is recursively defined by:

$$
\begin{aligned}
a_{h-1} < b_{h-1} &: (\text{QT of type I centered at } (0,0)) && \sigma_h = (I, m_h, 0),\\
a_{h-1} > b_{h-1} &: (\text{QT of type II centered at } (0,0)) && \sigma_h = (II, m_h, 0),\\
a_{h-1} = b_{h-1} &: (\text{QT of type I centered at } (0,\eta_h)) && \sigma_h = (I, m_h, \eta_h),
\end{aligned}
$$

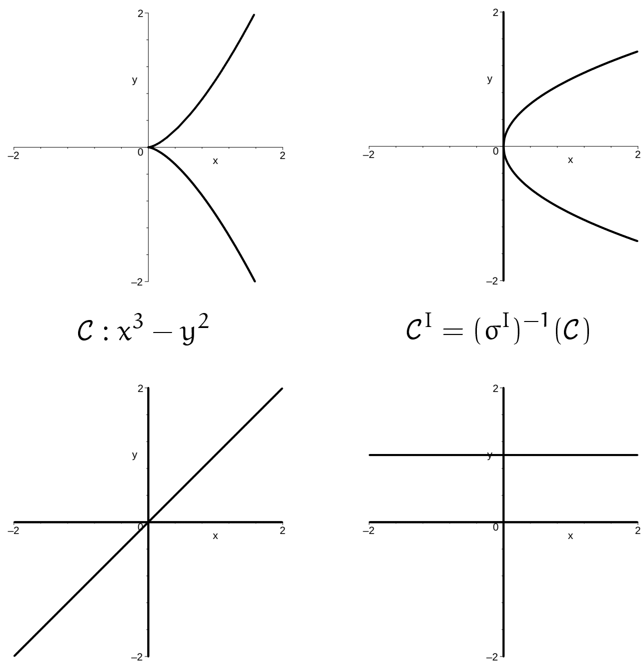
where

$$
\begin{aligned}
\mathfrak{z}_h &= \text{the proper transform of } \mathfrak{z}_{h-1} \text{ w.r.t. } \sigma_h,\\
\delta_h(t) &= (A_h(t), B_h(t)) \text{ is a local parametrization of } \mathfrak{z}_h,\\
a_h &= \text{ord}(A_h(t) - A_h(0)),\\
b_h &= \text{ord}(B_h(t) - B_h(0)), \text{ and}\\
m_h &= \min(a_h, b_h).
\end{aligned}
$$

The center q_h of $\mathfrak{z}_h$ is equal to the center of the QT σ_h.

Let H be the smallest number such that $\mathfrak{z}_H$ is a linear curve-branch intersecting with only one exceptional divisor and this intersection is transversal. The number H is called the *resolution length of* $\mathfrak{z}_0$ and $(\sigma_h)_{1\leq h\leq H}$ the *proper resolution sequence of* $\mathfrak{z}_0$.

Remark: In Definition 40 for the case $a_{h-1} = b_{h-1}$ we could also have chosen a QT of type II. This would result in a different (but basically equivalent) definition of resolution sequences. In any case, once we have fixed some type for the case $a_{h-1} = b_{h-1}$ the resolution sequence of a curve-branch is *uniquely* defined.



$\mathcal{C} : x^3 - y^2$ $\quad$ $\mathcal{C}^{\mathrm{I}} = (\sigma^{\mathrm{I}})^{-1}(\mathcal{C})$

$\mathcal{C}^{\mathrm{I,II}} = (\sigma^{\mathrm{II}})^{-1}(\mathcal{C}_0)$ $\quad$ $\mathcal{C}^{\mathrm{I,II,I}} = (\sigma^{\mathrm{I}})^{-1}(\mathcal{C}_{01})$

FIGURE 5. Embedded resolution of $x^3 - y^2$

Definition 41. For a curve $\mathcal{C}$ and a point q on it, let $\mathfrak{z}_i$, for $1 \leq i \leq I$, be the curve-branches of $\mathcal{C}$ centered at q. The *resolution graph of* $\mathcal{C}$ *at* q is the set of resolution sequences of the curve-branches $\mathfrak{z}_i$.

An algorithm for computing the resolution graph of a curve at the origin is given in Appendix A Section 1.

Example 3. Let $\mathcal{C}$ be the curve defined by $f(x,y) = x^3 - y^2$. In Figure 5 affine real plots of the transformed curves of $\mathcal{C}$ are shown until a normal crossing situation is reached. The resolution sequence of $\mathcal{C}$ at the origin is of the form $((\mathrm{I},2,0),(\mathrm{II},1,0),(\mathrm{I},1,0))$.

Example 4. The curve defined in Example 2 has three curve-branches $\mathfrak{z}_1$, $\mathfrak{z}_2$, and $\mathfrak{z}_3$ centered at the origin, whose resolution sequences are given by

$$\begin{aligned}
\mathfrak{z}_1 &: ((\mathrm{I},1,0),(\mathrm{I},1,0)),\\
\mathfrak{z}_2 &: ((\mathrm{I},1,0),(\mathrm{I},1,0)),\\
\mathfrak{z}_3 &: ((\mathrm{II},2,0),(\mathrm{I},1,0),(\mathrm{I},1,0)).
\end{aligned}$$

3. Classification

One possible way to classify singularities of plane algebraic curves is using multiplicity sequences and multiplicity graphs. In this section we will define both of them and we will show how they can easily be computed once the curve-branches are given by their Puiseux parametrizations.

Let $\mathcal{C}$ be a curve and q a point on it. Depending whether $\mathcal{C}$ is locally reducible or not we will distinguish two cases:

- First we assume that $\mathcal{C}$ is locally irreducible at q, i.e. there is only one curve-branch of $\mathcal{C}$ centered at q. In this case we will describe the resolution of q by a *multiplicity sequence*.
- On the other hand, if the curve $\mathcal{C}$ is locally reducible at q, i.e. $\mathcal{C}$ has more than one curve-branch at q, in a first step we will resolve each curve-branch separately getting multiplicity sequences for each curve-branch. Since these multiplicity sequences do not describe the structure of the singularity in full detail, we also need a sequence of equivalence relations describing how the branches interact with respect to each other during the resolution process. What we finally get is the so called *multiplicity graph*.

3.1. Irreducible Singularities

Definition 42. Let $\mathfrak{z}$ be a curve-branch centered at q and let Σ and H be the resolution sequence and the resolution length of $\mathfrak{z}$. The sequence $(q_h)_{0\leq h}$ of neighboring points of q is called the *neighboring point sequence of* $\mathfrak{z}$. The *proper neighboring point sequence of* $\mathfrak{z}$ is defined as $(q_h)_{0\leq h\leq H}$.

The corresponding sequences $(m_h)_{0\leq h}$ and $(m_h)_{0\leq h\leq H}$ of multiplicities of the neighboring points q_h are called the *multiplicity sequence* and *proper multiplicity sequence of* $\mathfrak{z}$, respectively.

Definition 43. We use the notation from above. The *conductor number* $c_{\mathfrak{z}}$ *of* $\mathfrak{z}$ is defined as

$$c_{\mathfrak{z}} = \sum_{h=0}^{H} m_h(m_h - 1).$$

We note, that we also could have defined the conductor number using the value semigroup of $\Gamma_{\mathfrak{z}}$ of $\mathfrak{z}$. It holds that $c_{\mathfrak{z}} = \inf\{\gamma \mid \gamma' \in \Gamma_{\mathfrak{z}}, \text{ for all } \gamma' \geq \gamma\}$.

Let $f(x, y) \in k[[x, y]]$ be the defining power series of $\mathfrak{z}$. As shown in [**Brieskorn and Knörrer, 1986**] the conductor number $c_{\mathfrak{z}}$ is then also equal to $c_{\mathfrak{z}} = 2\dim_k A^*/A$, where $A = k[[x, y]]/f(x, y)$ and A^* is the integral closure of A in its quotient field.

Let $\mathfrak{z}$ be a curve-branch defined by $\delta(t) = (A(t), B(t))$, where

$$A(t) = a_0 + a_{k_1} t^{k_1} + a_{k_2} t^{k_2} + \cdots,$$
$$B(t) = b_0 + b_{l_1} t^{l_1} + b_{l_2} t^{l_2} + \cdots,$$

$a_{k_i}, b_{l_i} \neq 0$, for $i > 0$.

If we look at the resolution sequence and the corresponding proper transforms of $\mathfrak{z}$ we saw in the last section that we are always in one of the following three cases.

(i) **Case I:** $k_1 < l_1$.
In this case the blowup is defined by

$$(x, y) \mapsto (x + a_0, xy + b_0)$$

and $\mathfrak{z}'$ is given by

$$A'(t) = a_{k_1} t^{k_1} + a_{k_2} t^{k_2} + \cdots,$$
$$B'(t) = b'_{l_1 - k_1} t^{l_1 - k_1} + \cdots.$$

(ii) **Case II:** $k_1 > l_1$.
Here the blowup is given by

$$(x, y) \mapsto (xy + a_0, y + b_0)$$

and $\mathfrak{z}'$ is of the form

$$A'(t) = a'_{k_1 - l_1} t^{k_1 - l_1} + \cdots,$$
$$B'(t) = b_{l_1} t^{l_1} + b_{l_2} t^{l_2} + \cdots.$$

(iii) **Case III:** $k_1 = l_1$.
In this case the blowup is defined by

$$(x, y) \mapsto (x + a_0, xy + b_0)$$

and $\mathfrak{z}'$ is given by

$$A'(t) = a_{k_1} t^{k_1} + a_{k_2} t^{k_2} + \cdots,$$
$$B'(t) = b'_{l_1 - k_1} t^{l_1 - k_1} + \cdots.$$

In both cases the multiplicity m of $q = (a_0, b_0)$ w.r.t. $\mathfrak{z}$ is equal to $\min(k_1, l_1)$. As we can see, for the curve-branch $\mathfrak{z}'$ we are in the same situation as we were for $\mathfrak{z}$. Hence, by iterating the above procedure we obtain an algorithm for computing the resolution and multiplicity sequence for irreducible singularities.

The following theorem shows that the multiplicity sequence of a curve-branch is already determined by its k-sequence. For reasons of notational convenience we introduce the following abbreviations for sequences.

Definition 44. Let a be an integer or a symbol. We define

$$\begin{aligned}
[a] &= (a),\\
[a]^n &= (\underbrace{a, \dots, a}_{n \text{ times}}) \qquad , \text{for } n \in \mathbb{N},\\
[a]^* &= (a, a, \dots).
\end{aligned}$$

If $A = (a_1, \dots, a_m)$ and $B = (b_1, \dots, b_n)$ are two sequences then their concatenation $(a_1, \dots, a_m, b_1, \dots, b_n)$ is written as AB.

Theorem 6 (Enriques – Chisini). *Let $\mathfrak{z}$ be a curve-branch and $(k_i)_{1 \le i \le I}$ be its k-sequence. We define the following Euclidean division chains ($i = 1, \dots, I$):*

$$\begin{aligned}
q_i &= \mu_{i,1} m_{i,1} + m_{i,2},\\
m_{i,1} &= \mu_{i,2} m_{i,2} + m_{i,3},\\
&\vdots\\
m_{i,J_{i-1}} &= \mu_{i,J_i} m_{i,J_i},
\end{aligned}$$

with

$$\begin{aligned}
q_0 &= 0,\\
q_i &= k_i - k_{i-1} \qquad , \textit{for } i = 1, \dots, I,\\
m_{1,1} &= m,\\
m_{i,1} &= m_{i-1,J_{i-1}} \qquad , \textit{for } i > 1.
\end{aligned}$$

The proper multiplicity sequence of $\mathfrak{z}$ is then of the form

$$[m_{1,1}]^{\mu_{1,1}} [m_{1,2}]^{\mu_{1,2}} \cdots [m_{1,J_1}]^{\mu_{1,J_1}} [m_{2,1}]^{\mu_{2,1}} \cdots [m_{I,J_I}]^{\mu_{I,J_I}} [1]$$

and the proper resolution length H is equal to

$$H = \sum_{i=1}^{I} \sum_{j=1}^{J_i} \mu_{i,j} + 1.$$

Proof. The proof consists basically of a formalization of the considerations sketched above taking into account that non-characteristic terms in the Puiseux parametrization of $\mathfrak{z}$ can be omitted.

Details can be found in [**Brieskorn and Knörrer, 1986**]. □

We can reformulate the Theorem of Enriques – Chisini in terms of characteristic pairs as shown in the next corollary.

Corollary 1. *Let* $(a_i, b_i)_{1\leq i\leq I}$ *be the characteristic pairs of a curve-branch* $\mathfrak{z}$. *The proper multiplicity sequence of* $\mathfrak{z}$ *is equal to*

$$[d_2m_{1,1}]^{\mu_{1,1}}[d_2m_{1,2}]^{\mu_{1,2}}\cdots[d_2m_{1,J_1}]^{\mu_{1,J_1}}[d_3m_{2,1}]^{\mu_{2,1}}\cdots[d_{I+1}m_{I,J_I}]^{\mu_{I,J_I}}[1],$$

where $\mu_{i,j}$ *and* $m_{i,j}$ *are defined by the Euclidean division chains (*$i = 1,\dots,I$*):*

$$\begin{aligned} a_i &= \mu_{i,1}m_{i,1} + m_{i,2},\\ m_{i,1} &= \mu_{i,2}m_{i,2} + m_{i,3},\\ &\vdots\\ m_{i,J_i-1} &= \mu_{i,J_i}m_{i,J_i}, \end{aligned}$$

with $m_{i,1} = b_i$*, for* $i \geq 1$*, and* $d_i = b_i d_{i+1}$*, starting with* $d_{I+1} = 1$*.*

Proof. Follows from Theorem 6 and Definition 24. □

Also the type of blowups in the resolution sequence of a curve-branch is fully determined by its k-sequence as shown in the next corollary.

Corollary 2. *We use the notation from Theorem 6. The types of the blowups in the resolution sequence of* $\mathfrak{z}$ *are given by*

$$[I]^{\mu_{1,1}}[II]^{\mu_{1,2}}[I]^{\mu_{1,3}}[II]^{\mu_{1,4}}\cdots[a_1]^{\mu_{1,J_1}-1}[I][II]^{\mu_{2,1}}\cdots[a_I]^{\mu_{I,J_I}-1}[I][I]^*,$$

where $a_i = I$ *if* J_i *is odd and* $a_i = II$ *otherwise, for* $1 \leq i \leq I$.

Proof. Each step in the division chains of Theorem 6 corresponds to a sequence of blowups of the same type. Therefore the sequence of types is determined in the same way as the sequence of multiplicities in Theorem 6. □

Given a Puiseux parametrization of $\mathfrak{z}$, we can compute the resolution sequence of $\mathfrak{z}$ by successively dividing the power-series of its parametrization. From the Theorem of Enriques – Chisini we know which terms in the parametrization correspond to proper neighboring points in the resolution sequence of $\mathfrak{z}$ (we will make this precise in Lemma 4).

Using this information we get optimal degree-bounds where we can truncate the power-series expansions of the transformed curve-branches $\mathfrak{z}_h$, for $1 \leq h \leq H$, without changing the proper resolution sequence of $\mathfrak{z}$. For details see the Algorithm in Appendix A Section 2.

Let $\mathfrak{z}$ be a curve-branch with center q defined by $\delta(t) = (A(t), B(t))$, where $A(t) = \sum_{k\geq 0} a_k t^k$ and $B(t) = \sum_{k\geq 0} b_k t^k$.

We now want to investigate the neighboring point sequence of $\mathfrak{z}$ in more detail. The coordinates of q_0 depend on a_0, b_0 and the multiplicity m_0 is equal to $m_0 = \min(\operatorname{ord}(A(t) - a_0), \operatorname{ord}(B(t) - b_0))$.

As we saw, the coordinates of the first neighboring point q_1 depend continuously on the tangent direction of $\mathfrak{z}$ at q which itself is determined by the coefficients a_{m_0}, b_{m_0}.

Repeating this observation we can divide the points in the neighboring point sequence of $\mathfrak{z}$ into two classes: Points whose coordinates can be changed (i.e. which can be moved freely) and points whose coordinates are already fixed by the existence of preceding neighboring points.

Definition 45. Let $(q_h)_{0\leq h}$ be the sequence of neighboring points of a curve-branch $\mathfrak{z}$ with center q defined by $\delta(t) = (A(t), B(t))$.

A point q_h is called a *free neighboring point of* q if its coordinates depend continuously on the coefficients of $A(t)$ and $B(t)$. All other neighboring points of q are called *non-free neighboring points* or *satellite points of* q.

A satellite point q_h is a *terminal satellite point of* q if q_{h+1} is free. In this case the point q_{h+1} is also called a *leading free neighboring point of* q.

Lemma 3. *Let $\mathfrak{z}$ be a linear curve branch with origin q. Then all neighboring points of q are simple and free.*

Proof. W.l.o.g. we can assume that $\mathfrak{z}$ is in general position. Since $\mathfrak{z}$ is linear, it has a Puiseux parametrization of the form $(a_0 + t, B(t))$, where $B(t) = \sum_{k\geq 0} b_k t^k \in k[[t]]$. Therefore we see that $q_0 = q$ is simple and free (depending on a_0, b_0) and all blowups in the resolution sequence of $\mathfrak{z}$ are of type I. Recursively one can show: If we assume that the coordinates of $q_0, \ldots, q_{h-1}$ are already fixed, the coordinates of q_h depend continously on b_h. □

Lemma 4. *Given a curve-branch $\mathfrak{z}$ with k-sequence $(k_i)_{1\leq i\leq I}$. Let the numbers J_i and $\mu_{i,j}$, for $1 \leq i \leq I$, $1 \leq j \leq J_i$, be defined as in the Theorem of Enriques–Chisini (Theorem 6). The neighboring points of q are free points (F) respectively satellite points (S) of q according to the following scheme*

$$[F]^{\mu_{1,1}+1}[S]^{\tau_1-\mu_{1,1}-1}[F]^{\mu_{2,1}+1}[S]^{\tau_2-\mu_{2,1}-1}\cdots[F]^{\mu_{I,1}+1}[S]^{\tau_I-\mu_{I,1}-1}[F][F]^*,$$

where $\tau_i = \sum_{1\leq j\leq J_i} \mu_{i,j}$, for $1 \leq i \leq I$.

Hence, the leading free neighboring points of q are exactly the points $q_{\tau_1}, q_{\tau_1+\tau_2}, \ldots, q_{\tau_1+\cdots+\tau_I}$.

Proof. Let $\delta(t) = (a_0 + a_e t^e, \sum_{0\leq k} b_k t^k)$ be a Puiseux parametrization of $\mathfrak{z}$. We also define the numbers $m_{i,j}$ as in Theorem 6. According to the k-sequence of $\mathfrak{z}$ we know that only the following coefficients b_k, $k \leq k_I$, of $B(t)$ may be unequal to zero (where $b_{k_1}, \ldots, b_{k_I}$ have to be unequal to zero):

$$\begin{array}{lllll}
b_0, & b_{m_{1,1}}, & b_{2m_{1,1}}, & \cdots, & b_{\mu_{1,1}m_{1,1}}, \\
b_{k_1}, & b_{k_1+m_{2,1}}, & b_{k_1+2m_{2,1}}, & \cdots, & b_{k_1+\mu_{2,1}m_{2,1}}, \\
\vdots & & & & \\
b_{k_{I-1}}, & b_{k_{I-1}+m_{I,1}}, & b_{k_{I-1}+2m_{I,1}}, & \cdots, & b_{k_{I-1}+\mu_{I,1}m_{I,1}}, \\
b_{k_I}. & & & &
\end{array}$$

We denote the proper resolution length of $\mathfrak{z}$ by H and the neighboring point sequence as usually by $(q_h)_{0\leq h}$. Further, let $\mathfrak{z}_0 = \mathfrak{z}$.

It is obvious that q_0 is free (its coordinates depend on a_0, b_0). Applying $\mu_{1,1}$ QTs (of type I) we see that the next $\mu_{1,1}$ neighboring points are also free and the curve-branch $\mathfrak{z}_{\mu_{1,1}}$ is of the form

$$(t^{m_{1,1}}, b_{\mu_{1,1}m_{1,1}} + b_{k_1}t^{m_{1,2}} + \cdots),$$

with $0 \neq m_{1,2} < m_{1,1}$.

The next QT is of type II and $\mathfrak{z}_{\mu_{1,1}+1}$ is given by

$$(t^{m_{1,1}-m_{1,2}} + \cdots, b_{k_1}t^{m_{1,2}} + \cdots).$$

So we see, that $q_{\mu_{1,1}+1}$ is not free. The same is true for the next $\mu_{1,2}-1$ neighboring points. After applying in total $\mu_{1,2}$ QTs of type II, we have

$$\mathfrak{z}_{\mu_{1,1}+\mu_{1,2}} : (t^{m_{1,3}} + \cdots, b_{k_1}t^{m_{1,2}} + \cdots).$$

If $m_{1,3} \neq 0$ we are basically in the same situation as we were for $\mathfrak{z}_{\mu_{1,1}}$ with the difference that now QTs of type I have to be applied.

After repeating the above procedure J_1 times we have applied $\tau_1 = \mu_{1,1} + \mu_{1,2} + \cdots + \mu_{1,J_1}$ QTs and the curve-branch $\mathfrak{z}_{\tau_1}$ is of the form

$$\mathfrak{z}_{\tau_1} : (at^{m_{1,l-1}} + \cdots, b + \cdots),$$

where a and b depend on a_0, b_0, $b_{m_{1,1}}$, $\ldots$, $b_{\mu_{1,1}m_{1,1}}$, b_{k_1}, hence q_{τ_1} is free.

At this stage we have resolved the first characteristic term of $\mathfrak{z}$ and we find ourselves in the same situation as we started with ($\mathfrak{z}_{\tau_1}$ corresponds now to $\mathfrak{z}_0$ and q_{τ_1} corresponds to q_0).

Resolving all characteristic terms we have to apply H QTs and we end up with a linear curve-branch $\mathfrak{z}_H$. It is easily seen, that the coordinates of q_H are determined by a_0, b_0, $\ldots$, b_{k_I}.

Putting all together we get the following distribution of free and satellite points for the proper neighboring point sequence

$$[F]^{\mu_{1,1}+1}[S]^{\tau_1-\mu_{1,1}-1}[F]^{\mu_{2,1}+1}[S]^{\tau_2-\mu_{2,1}-1}\ldots[F]^{\mu_{I,1}+1}[S]^{\tau_I-\mu_{I,1}-1}[F].$$

From Lemma 3 we know, that from q_H on all neighboring points are simple and free. $\square$

Corollary 3. *Let $\mathfrak{z}$ be a curve-branch. The coordinates of the points in the proper neighboring point sequence of $\mathfrak{z}$ are determined by the coefficients of the singular part of the Puiseux parametrization of $\mathfrak{z}$.*

Proof. Let $(a_0 + t^e, \sum_{k=0}^{K} b_k t^k)$ be the singular part of a Puiseux parametrization of $\mathfrak{z}$. As we saw in the proof of Lemma 4 the terms a_0, b_k, $0 \leq k \leq K$, determine the coordinates of the last leading free neighboring point q_H, where H is the proper resolution length. Since the coordinates of q_h, for $0 \leq h < H$, depend on terms of order less than K the singular part of the Puiseux expansion of $\mathfrak{z}$ determines the coordinates of all points in the proper neighboring point sequence. □

Example 5. The curve-branch defined by $\delta(t) = (t^{100}, t^{250} + t^{375} + t^{390} + t^{391})$ has characteristic-pairs

$$((5,2),(5,2),(7,5),(7,5)).$$

The types of QTs in its proper resolution sequence and its multiplicity sequence are given by

$$[\mathrm{I}]^2[\mathrm{II}][\mathrm{I}]^3[\mathrm{II}][\mathrm{I}][\mathrm{II}][\mathrm{I}][\mathrm{II}][\mathrm{I}][\mathrm{II}]^4[\mathrm{I}],$$

$$[100]^2[50]^4[25]^2[15][10][5]^2[1]^5[1].$$

The free neighboring points are

$$[\mathrm{F}]^3[\mathrm{S}][\mathrm{F}]^3[\mathrm{S}][\mathrm{F}][\mathrm{S}]^3[\mathrm{F}][\mathrm{S}]^4[\mathrm{F}][\mathrm{F}]^*.$$

For a multiplicity m_h occurring in a multiplicity sequence $(m_h)_{0\leq h}$ of a curve-branch there are according to Theorem 6 three possibilities. Either the multiplicity m_h is equal to m_{h+1} or the multiplicity m_h is greater than m_{h+1} which then implies that either

$$m_h = \mu m_{h+1} + m_{h+\mu+1}$$

or

$$m_h = \mu m_{h+1}.$$

Anyway, for every neighboring point q_h there exist neighboring points which immediately follow q_h, such that the sum of their multiplicities is equal to m_h.

Definition 46. Let $\mathfrak{z}$ be a curve branch with origin q. For a neighboring point q_h of q with multiplicity m_h we define $l_h \in \mathbb{N}$ in such a way that $m_h = \sum_{h'=h+1}^{h+l_i} m_{h'}$. The points $q_{h'}$, for $h < h' \leq h + l_h$, are called *proximate points of* q_h.

Example 6. We look again at the curve-branch defined by the Puiseux parametrization $\delta(t) = (t^{100}, t^{250} + t^{375} + t^{390} + t^{391})$. It has the following structure of proximate points

h	l_h	h	l_h	h	l_h	h	l_h	h	l_h
0	1	3	1	6	1	9	2	12	1
1	3	4	1	7	3	10	2	13	1
2	1	5	2	8	2	11	5	14	1

3.2. Reducible Singularities

In the case of reducible singularities we also have to take into account that it may happen that neighboring points of different curve-branches coincide up to some QT, i.e. different curve-branches have the same center for the first QTs.

Definition 47. Let $\mathcal{C}$ be a curve and q a point on it. Assume that there are I curve-branches $\mathfrak{z}_i$ of $\mathcal{C}$ centered at q, for $1 \le i \le I$. Let the multiplicity sequences of $\mathfrak{z}_i$, for $1 \le i \le I$, be given by

$$(m_{i,h})_{0 \le h}.$$

For each $h \in \mathbb{N}$ we define an equivalence relation $\sim_h$ in such a way that $i_1 \sim_h i_2$ if and only if $q_{i_1,h'} = q_{i_2,h'}$, for all $h' \le h$, i.e. the multiplicities $m_{i_1,h}$ and $m_{i_2,h}$ belong to the same neighboring point.

The multiplicity sequences $(m_{i,h})_{0 \le h}$, $1 \le i \le I$, together with the equivalence relations $(\sim_h)_{0 \le h}$ are called the *multiplicity graph of $\mathcal{C}$ at* q.

The corresponding neighboring points together with $(\sim_h)_{0 \le h}$ are called the *neighboring point graph of $\mathcal{C}$ at* q.

As in the case of irreducible singularities only a finite part of the neighborhood graph is non-trivial , which we call again the proper part.

Definition 48. We use the same notation as in Definition 47. Let H_i', for $1 \le i \le I$, be the length of the proper multiplicity sequence belonging to $\mathfrak{z}_i$ if we resolve each curve-branch separately. Furthermore let H_i^*, for $1 \le i \le I$, be defined as the smallest number such that q_{i,H_i^*} does not coincide with any other neighboring point.

According to Theorem 5 all these numbers are finite and we define the *proper resolution length* H_i *of a curve-branch* $\mathfrak{z}_i$ *(w.r.t. the curve $\mathcal{C}$)* as $H_i = \max(H_i', H_i^*)$. Finally, let $H = \max(H_i \mid 1 \le i \le I)$.

The multiplicity sequences $(m_{i,h})_{0 \le h}$, $1 \le i \le H$, truncated at H_i together with the equivalence relations $(\sim_h)_{0 \le h \le H}$ are called the *proper multiplicity graph of $\mathcal{C}$ at* q.

The neighboring point sequences $(q_{i,h})_{0\leq h}$, $1 \leq i \leq I$, truncated at H_i together with the equivalence relations $(\sim_h)_{0\leq h\leq H}$ are called the *proper neighboring point graph of $\mathcal{C}$ at* q.

We are now able to define the singular part of a Puiseux parametrization $\delta(t)$ w.r.t. a singular point of a curve, as all those terms of $\delta(t)$ which correspond to neighboring points in the proper neighboring point graph of q.

Just to clarify notation we remark that the multiplicity of a neighboring point $q_{i,h}$ w.r.t. to the curve-branch $\mathfrak{z}_j$ is $m_{i,h}$, whereas the multiplicity of $q_{i,h}$ w.r.t. to the curve $\mathcal{C}$ is $n_{i,h} = \sum_{i'\sim_h i} m_{i',h}$.

The definition of the conductor number of a curve-branch generalizes to reducible singularities as follows

Definition 49. Using the notation from above we define the *conductor numbers* $c_{\mathfrak{z}_i}$ *of* $\mathfrak{z}_i$ *w.r.t.* $\mathcal{C}$ as

$$c_{\mathfrak{z}_i} = \sum_{h=0}^{H_i} m_h(n_h - 1).$$

The Milnor-number μ_q of $\mathcal{C}$ at q is equal to

$$\mu_q = \sum_{1\leq i\leq I} c_{\mathfrak{z}_i} - I + 1.$$

Example 7. Given a curve $\mathcal{C}$ with five curve-branches $\mathfrak{z}_1, \dots, \mathfrak{z}_5$ centered at the origin $q = (0,0)$. The curve-branches have Puiseux expansions

$$\begin{aligned}
\mathfrak{z}_1 &: (t^4, t^6 + t^9),\\
\mathfrak{z}_2 &: (t^2, t^3 + t^5),\\
\mathfrak{z}_3 &: (t^2, t^3 + 2t^5),\\
\mathfrak{z}_4 &: (t^2, t^5),\\
\mathfrak{z}_5 &: (t, t^3).
\end{aligned}$$

The proper multiplicity graph of $\mathcal{C}$ at q is shown in Figure 6. Its conductor numbers and Milnor number are given by $c_{\mathfrak{z}_1} = 66$, $c_{\mathfrak{z}_2} = 34$, $c_{\mathfrak{z}_3} = 34$, $c_{\mathfrak{z}_4} = 33$, $c_{\mathfrak{z}_5} = 17$, and $\mu_q = 180$.

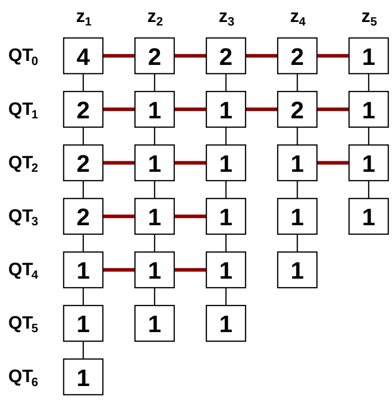


FIGURE 6. Multiplicity graph of a reducible singularity

4. Intersection Multiplicities

As one could expect, once we have resolved a curve-branch we can easily compute its intersection multiplicities with other curve-branches and curves.

We start quoting two results showing how multiplicities of points and intersection multiplicities behave w.r.t. blowups.

Lemma 5. *Let $\mathcal{C} \subset X$ be a curve, $q \in \mathcal{C}$ a point on it, and $\sigma_q : X \to X'$ the blowup of X centered at q. We denote the proper transform of $\mathcal{C}$ w.r.t. σ_q by $\mathcal{C}'$ and the exceptional line by $\mathcal{E}'$. Then it holds that*

$$\operatorname{mult}_q(\mathcal{C}) = \sum_{p \in \mathcal{C}' \cap \mathcal{E}'} \operatorname{mult}_p(\mathcal{C}', \mathcal{E}').$$

Proof. See [**Brieskorn and Knörrer, 1986**, Section 8.4, Lemma 1]. □

Lemma 6. *Let $\mathfrak{z}_1$, $\mathfrak{z}_2$ be two distinct curve-branches both centered at q. By $\mathfrak{z}_1'$, $\mathfrak{z}_2'$ we denote their proper transforms w.r.t. the affine blowup σ_q. If $\mathfrak{z}_1'$ and $\mathfrak{z}_2'$ have again a common center q' the intersection multiplicity of $\mathfrak{z}_1$ and $\mathfrak{z}_2$ is equal to*

$$\operatorname{mult}_q(\mathfrak{z}_1, \mathfrak{z}_2) = \operatorname{mult}_q(\mathfrak{z}_1)\operatorname{mult}_q(\mathfrak{z}_2) + \operatorname{mult}_{q'}(\mathfrak{z}_1', \mathfrak{z}_2'),$$

and otherwise equal to

$$\operatorname{mult}_q(\mathfrak{z}_1, \mathfrak{z}_2) = \operatorname{mult}_q(\mathfrak{z}_1)\operatorname{mult}_q(\mathfrak{z}_2).$$

Proof. A proof can be found e.g. in [**van der Waerden, 1973**, Paragraph 56, Theorem 6]. □

Lemma 7. *Let $\mathfrak{z}_1$, $\mathfrak{z}_2$ be two distinct curve-branches both centered at q. Further, let their multiplicity graph be given by $(m_{1,h})_{0 \le h}$, $(m_{2,h})_{0 \le h}$ and $(\sim_h)_{0 \le h}$. Because $\mathfrak{z}_1$ and $\mathfrak{z}_2$ are distinct, there exists a natural number h^*, s.t. $1 \sim_h 2$, for $0 \le h < h^*$, and $1 \not\sim_{h^*} 2$.*

The intersection multiplicity of $\mathfrak{z}_1$ and $\mathfrak{z}_2$ at q satisfies

$$\operatorname{mult}_q(\mathfrak{z}_1, \mathfrak{z}_2) = \sum_{h=0}^{h^*-1} m_{1,h} m_{2,h}.$$

Proof. Follows by induction from Lemma 6. □

Remark: Lemma 7 in combination with Lemma 2 and Theorem 6 provides an easy way to compute the equivalence relations $\sim_h$ in the case the curve-branches are given by their classical Puiseux parametrizations. In the next chapter when discussing the Newton–Puiseux algorithm, we will see how the intersection multiplicities between curve-branches of reducible singularities can directly be computed along with the Newton transformations.

Example 8. The intersection multiplicities of the curve-branches given in Example 7 are shown in the following table.

	$\mathfrak{z}_1$	$\mathfrak{z}_2$	$\mathfrak{z}_3$	$\mathfrak{z}_4$	$\mathfrak{z}_5$
$\mathfrak{z}_1$	–	15	15	12	6
$\mathfrak{z}_2$	15	–	8	6	3
$\mathfrak{z}_3$	15	8	–	6	3
$\mathfrak{z}_4$	12	6	6	–	5
$\mathfrak{z}_5$	6	3	3	5	–

Lemma 7 also motivates the following Definition of truncated intersection multiplicity which we will need for an alternative definition of adjoint curves in the next Section.

Definition 50. We use the notation from Lemma 7. Let T be a non negative integer. The *intersection multiplicity* $\mathrm{mult}_q^T(\mathfrak{z}_1,\mathfrak{z}_2)$ *of* $\mathfrak{z}_1$ *and* $\mathfrak{z}_2$ *at* q *truncated at* T is defined as

$$\mathrm{mult}_q^T(\mathfrak{z}_1,\mathfrak{z}_2) = \sum_{h=0}^{\min(T,h^*)-1} m_{1,h}m_{2,h}.$$

For a curve-branch $\mathfrak{z}$ and curve $\mathcal{C}$ we define the *intersection multiplicity* $\mathrm{mult}_q^T(\mathfrak{z},\mathcal{C})$ *of* $\mathfrak{z}$ *and* $\mathcal{C}$ *at* q *truncated at* T as

$$\mathrm{mult}_q^T(\mathfrak{z},\mathcal{C}) = \sum_{\mathfrak{z}' \subseteq \mathcal{C} \text{ with origin } q} \mathrm{mult}_q^T(\mathfrak{z},\mathfrak{z}').$$

By definition it holds that $\mathrm{mult}_q^{T_1}(\mathfrak{z}_1,\mathfrak{z}_2) \geq \mathrm{mult}_q^{T_2}(\mathfrak{z}_1,\mathfrak{z}_2)$ iff $T_1 \geq T_2$, and $\mathrm{mult}_q^T(\mathfrak{z}_1,\mathfrak{z}_2) = \mathrm{mult}_q(\mathfrak{z}_1,\mathfrak{z}_2)$, for $T \geq h^*$.

Definition 51. Let $\mathfrak{z}$ be a curve-branch with Puiseux parametrization $(a_0+t^e, B(t))$, where $B(t) = \sum_{0\leq k} b_k t^k$, and neighboring point sequence $(q_h)_{0\leq h}$. By H we denote the proper resolution length of $\mathfrak{z}$. As shown in Lemma 4 for $T \geq H$ the neighboring points q_T are free and there exists a uniquely determined minimal positive integer $k(T)$ such that the coordinates of q_T depend on a_0, b_0, $\ldots$, $b_{k(T)}$.

The *curve-branch* $\mathfrak{z}$ *truncated at* T *with parameter* λ, written as $\mathfrak{z}_{T,h}$, is defined as

$$(a_0+t^e, \sum_{k=0}^{k(T)-1} b_k t^k + \lambda t^{k(T)}).$$

Lemma 8. *Let $\mathcal{C}$ be the curve defined by $f(x,y) \in k[x,y]$ and let $\mathfrak{z}$ be a curve-branch with proper resolution length H (w.r.t. $\mathcal{C}$). Then it holds* *for all* $T \geq H$ *and some generically chosen* $\lambda \in k$ *that*

$$\mathrm{mult}_q^T(\mathfrak{z}, \mathcal{C}) = \mathrm{mult}_q(\mathfrak{z}_{T,\lambda}, \mathcal{C}) = \mathrm{ord}(f(A(t), B_{T,\lambda}(t))),$$

where $(A(t), B_{T,\lambda}(t))$ *is the Puiseux expansion of* $\mathfrak{z}_{T,\lambda}$.

PROOF. Let $\mathfrak{z}_i$, $1 \leq i \leq I$, be the curve-branches of $\mathcal{C}$ at q, $T \geq H$ and $\mathfrak{z}_{T,\lambda}$ some generically chosen curve-branch of $\mathfrak{z}$ truncated at T. We will first show that $\mathrm{mult}_q^T(\mathfrak{z}, \mathfrak{z}_i) = \mathrm{mult}_q(\mathfrak{z}_{T,\lambda}, \mathfrak{z}_i)$. Let the numbers h_i^* be defined as in Lemma 7 for $\mathfrak{z}_{T,\lambda}$ and $\mathfrak{z}_i$. Since $h_i^* \leq T$ it holds that

$$\begin{aligned}\mathrm{mult}_q^T(\mathfrak{z}, \mathfrak{z}_i) = \mathrm{mult}_q^T(\mathfrak{z}_{T,\lambda}, \mathfrak{z}_i) &= \sum_{h=0}^{\min(h_i^*, T)-1} m_h m_{i,h} \\ &= \sum_{h=0}^{h_i^*-1} m_k m_{i,h} = \mathrm{mult}_q(\mathfrak{z}_{T,\lambda}, \mathfrak{z}_i) = \mathrm{ord}(f(A(t), B_{T,\lambda}(t))).\end{aligned}$$

$\square$

5. Adjoint Curves

Let $\mathcal{C}$ be an irreducible curve defined by $f(x,y) \in k[x,y]$ with singular points q_j, for $1 \leq j \leq J$, where we assume that $\mathcal{C}$ has no singular point at infinity. For each singularity q_j let $\mathfrak{z}_{j,i}$, for $1 \leq i \leq I_j$, be the curve-branches of $\mathcal{C}$ centered at q_j. We define $m_{j,i,h}$ and $n_{j,i,h}$, for $1 \leq h \leq H_{j,i}$, as the multiplicities of the proper neighboring points $q_{j,i,h}$ of q_j w.r.t. to the curve-branch $\mathfrak{z}_{j,i}$ and the curve $\mathcal{C}$, respectively.

Among the many ways defining adjoint curves we quote the following two ones, see [**Keller, 1974**] and [**Brieskorn and Knörrer, 1986**].

Definition 52. For each curve-branch $\mathfrak{z}_{j,i}$ we define a natural number $d_{j,i}$ in the following way

$$d_{j,i} = \sum_{h=0}^{H_{j,i}} m_{j,i,h}(n_{j,i,h} - 1).$$

A curve $\mathcal{D}$ (which may also be reducible) is called an *adjoint curve of* $\mathcal{C}$ if the intersection multiplicity of each curve-branch $\mathfrak{z}_{j,i}$ of $\mathcal{C}$ and the curve $\mathcal{D}$ at all singular points q_i is at least $d_{j,i}$, i.e.

$$\operatorname{mult}_{q_j}(\mathfrak{z}_{j,i}, \mathcal{D}) \geq d_{j,i} \ ,$$

for $1 \leq j \leq J$, $1 \leq i \leq I_j$.

Definition 53. We fix some curve point q of $\mathcal{C}$. For each neighboring point $(q_h)_{0 \leq h}$ of q let a natural number e_h be given. Let $\mathcal{D}$ be a curve and let o_h be the multiplicities of q_h w.r.t. $\mathcal{D}$.

The curve $\mathcal{D}$ has *multiplicities at least* e_h *at the neighboring points* q_h if

$$f_h = o_h - e_h + f_{h-1} \geq 0,$$

for all $0 \leq h$, where $f_{-1} = 0$.

Proposition 1. *A curve $\mathcal{D}$ is adjoint to $\mathcal{C}$ iff $\mathcal{D}$ has at all proper neighboring points $q_{j,i,h}$ of $\mathcal{C}$ multiplicity at least $n_{j,i,h} - 1$, for $1 \leq j \leq J$, $1 \leq i \leq I_j$ and $1 \leq h \leq H_{j,i}$.*

PROOF. A proof can be found in [**Schicho, 1995**] or [**Mňuk, 1995**]. □

The following Lemma shows that the intersection multiplicity in Definition 52 can be replaced by a truncated intersection multiplicity.

Lemma 9. *A curve $\mathcal{D}$ is adjoint to $\mathcal{C}$ if and only if*

$$\operatorname{mult}_{q_j}^{H_{j,i}}(\mathfrak{z}_{j,i}, \mathcal{D}) \geq d_{j,i},$$

for $1 \leq j \leq J$, $1 \leq i \leq I_j$.

PROOF. It is clear that if $\mathrm{mult}_{q_j}^{H_{j,i}}(\mathfrak{z}_{j,i}, \mathcal{D}) \geq d_{j,i}$, for $1 \leq j \leq J$, $1 \leq i \leq I_j$, then $\mathcal{D}$ is adjoint to $\mathcal{C}$.

So we assume that $\mathcal{D}$ is adjoint to $\mathcal{C}$ and we show that the intersection multiplicities fulfill the required inequalities. We fix some singular point q_j and some curve-branch $\mathfrak{z}_{j,i}$. Let $q = q_j$, $\mathfrak{z} = \mathfrak{z}_{j,i}$, $H = K_{j,i}$, $q_h = q_{j,i,h}$, $m_h = m_{j,i,h}$, $n_h = n_{j,i,h}$, $d = d_{j,i}$, and let o_h be the multiplicity of q_h w.r.t. $\mathcal{D}$. We then have to show that

$$\mathrm{mult}_q^H(\mathfrak{z}, \mathcal{D}) \geq \sum_{h=0}^{H} m_h(n_h - 1) = d.$$

From Proposition 1 we know that $\mathcal{D}$ has at least multiplicity $n_h - 1$ at q_h. Hence one gets (f_h defined as in Definition 53)

$$\begin{aligned}
\mathrm{mult}_q^H(\mathfrak{z}, \mathcal{D}) &= \sum_{h=0}^{H} m_h o_h = \sum_{h=0}^{H} m_h(n_h - 1 + f_h - f_{h-1}) \\
&= d + \sum_{h=0}^{H-1} m_h f_h + m_H f_H - m_0 f_{-1} - \sum_{h=0}^{H-1} m_{h+1} f_h \\
&= d + \sum_{h=0}^{H-1} (m_h - m_{h+1}) f_h + m_H f_H \\
&\geq d.
\end{aligned}$$

□

5.1. The Polar Curve

In order to give a more convenient definition of adjoint curves we introduce the notion of a polar curve of a curve w.r.t. to some affine point.

Definition 54. Given a point $p = (p_x, p_y) \in \mathbf{A}^2$ we define the *polar-curve* $\mathcal{P}$ *of* $\mathcal{C}$ *with respect to* p by the polynomial

$$p_x \frac{\partial}{\partial x} f(x, y) + p_y \frac{\partial}{\partial y} f(x, y).$$

In the following we will prove two short Lemmata which deal with the intersection-multiplicity of a curve-branch and the polar curve. (We also reproduce the proofs here, although we are quite sure that these Lemmata are well known but we could not find suitable proofs in the literature.)

Lemma 10. *Let* q *be a point of* $\mathcal{C}$, $\mathfrak{z}$ *a curve-branch of* $\mathcal{C}$ *centered at* q, *and* p_1, $p_2 \in \mathbf{A}^2$ *both not equal to* q. *We define* $\mathcal{P}_1$, $\mathcal{P}_2$ *as the polar curves of* $\mathcal{C}$ *w.r.t.* p_1, p_2, *and* $\mathcal{L}_1$, $\mathcal{L}_2$ *as the lines from* q *to* p_1, p_2, *respectively.*

Then it holds that

$$\mathrm{mult}_q(\mathfrak{z},\mathcal{P}_1)-\mathrm{mult}_q(\mathfrak{z},\mathcal{L}_1)=\mathrm{mult}_q(\mathfrak{z},\mathcal{P}_2)-\mathrm{mult}_q(\mathfrak{z},\mathcal{L}_2).$$

PROOF. W.l.o.g. we can assume that $q=(0,0)$ and that q, p_1, p_2 do not lie on a line. (If this were not the case, then we could introduce a third point p_3 not lying on this line and prove the Lemma for q, p_1, p_3 and q, p_2, p_3.) So we can assume that $p_1=(1,0)$ and $p_2=(0,1)$. This yields then

$$\mathcal{L}_1\colon y=0,$$
$$\mathcal{L}_2\colon x=0$$

and the polar curves $\mathcal{P}_1$ and $\mathcal{P}_2$ are given by

$$\mathcal{P}_1\colon \frac{\partial}{\partial x}f(x,y)=0,$$
$$\mathcal{P}_2\colon \frac{\partial}{\partial y}f(x,y)=0.$$

Let $(A(t),B(t))$ be a local parametrization of $\mathfrak{z}$. Since $\mathfrak{z}$ is a curve-branch of $\mathcal{C}$ it holds that

$$\frac{d}{dt}f(A(t),B(t))=0.$$

But this is equal to

$$\frac{\partial}{\partial x}f(A(t),B(t))\frac{d}{dt}A(t)+\frac{\partial}{\partial y}f(A(t),B(t))\frac{d}{dt}B(t)=0$$

from which follows that

$$\mathrm{ord}(\frac{\partial}{\partial x}f(A(t),B(t)))+\mathrm{ord}(\frac{d}{dt}A(t))=$$
$$=\mathrm{ord}(\frac{\partial}{\partial y}f(A(t),B(t)))+\mathrm{ord}(\frac{d}{dt}B(t)).$$

So we get (see Lemma 2)

$$\mathrm{mult}_q(\mathfrak{z},\mathcal{P}_1)-\mathrm{mult}_q(\mathfrak{z},\mathcal{L}_1)=\mathrm{mult}_q(\mathfrak{z},\mathcal{P}_2)-\mathrm{mult}_q(\mathfrak{z},\mathcal{L}_2).$$

□

Lemma 11. *Let* q *be a point of* $\mathcal{C}$ *and* $\mathfrak{z}$ *a curve-branch of* $\mathcal{C}$ *centered at* q. *Furthermore, let* $p\neq q\in\mathbf{A}^2$, $\mathcal{P}$ *be the polar curve of* $\mathcal{C}$ *w.r.t.* p, *and* $\mathcal{L}$ *the line from* q *to* p. *Then*

$$\mathrm{mult}_q(\mathfrak{z},\mathcal{P})-\mathrm{mult}_q(\mathfrak{z},\mathcal{L})=\sum_{h=0}^{H}m_h(n_h-1)-1.$$

PROOF. Using Lemma 10 we can assume that the point p is in general position w.r.t. the curve $\mathcal{C}$ which implies that $\mathrm{mult}_q(\mathcal{C})=\mathrm{mult}_q(\mathcal{C},\mathcal{L})=n_0$ and $\mathrm{mult}_q(\mathfrak{z})=\mathrm{mult}_q(\mathfrak{z},\mathcal{L})=m_0$. Also w.l.o.g. we assume that the coordinate system is chosen in such a way that $q = (0,0)$ and $p = (0,1)$. Hence $\mathcal{P}$ and $\mathcal{L}$ are given by

$$\mathcal{L}\colon x = 0,$$
$$\mathcal{P}\colon \frac{\partial}{\partial y} f(x,y) = 0.$$

We prove the Lemma by induction on the length H of the standard resolution of $\mathfrak{z}$ (within $\mathcal{C}$).

- $z = 0$:
  In this case q is a regular point. So we get $\mathrm{mult}_q(\mathfrak{z},\mathcal{P}) - \mathrm{mult}_q(\mathfrak{z},\mathcal{L}) = 0 - 1 = -1$ which is equal to $1(1-1) - 1 = -1$.
- $z \to z+1$:
  In this case (since we assumed that y-axis is not tangent to $\mathfrak{z}$) a blowup $\sigma_q^{I}\colon (x,y) \mapsto (x,xy)$ of type I is applied to $\mathfrak{z}$ in the resolution sequence. Let $\mathcal{C}'$ and $\mathcal{P}'$ be the proper transforms of $\mathcal{C}$ and $\mathcal{P}$ w.r.t. σ_q^{I}. The exceptional divisor $\mathcal{E}'$ is given by $\mathcal{E}'\colon x^{m_0} = 0$ and we can easily check that $\mathcal{P}'$ is the polar curve of $\mathcal{C}'$ w.r.t. the point $(0,1)$. The line through the first neighboring point q' of q and $(0,1)$ is defined by $x = 0$, and therefore we are in a situation where we can apply the induction hypothesis

  $$\mathrm{mult}_{q'}(\mathfrak{z}',\mathcal{P}') - \mathrm{mult}_{q'}(\mathfrak{z}',\mathcal{L}') = \sum_{h=1}^{H} m_h(n_h - 1) - 1,$$

  where it holds that

  $$\mathrm{mult}_{q'}(\mathfrak{z}',\mathcal{L}') = m_0.$$

  So by Lemma 6 we get

  $$\begin{aligned}\mathrm{mult}_q(\mathfrak{z},\mathcal{P}) &= m_0(n_0 - 1) + \mathrm{mult}_{q'}(\mathfrak{z}',\mathcal{P}') = \\ &= m_0(n_0 - 1) + \sum_{h=1}^{H} m_h(n_h - 1) - 1 + m_0 = \\ &= \sum_{h=0}^{H} m_h(n_h - 1) - 1 + m_0.\end{aligned}$$

  Once again using the fact that p is a general point this is equal to

  $$\mathrm{mult}_q(\mathfrak{z},\mathcal{P}) - \mathrm{mult}_q(\mathfrak{z},\mathcal{L}) = \sum_{h=0}^{H} n_h(m_h - 1) - 1.$$

$\square$

We now once again want to replace in the intersection multiplicities by truncated intersection multiplicities (as we did in Lemma 9). To do so, we need the following technical result.

Lemma 12. *For any proper multiplicity sequence of the form*

$$[m_r]^{\mu_r}[m_{r-1}]^{\mu_{r-1}}\cdots[m_2]^{\mu_2}[m_1]^{\mu_1}[m_0]^{\mu_0}$$

it holds that

$$m_{i-1}+m_i \leq \sum_{j=0}^{i-1}\mu_j m_j \qquad \text{, for } 1\leq i\leq r.$$

Proof. We proof this Lemma by induction on i. For the induction base $i=1$ we know (from Theorem 6) that $m_0=m_1=1$ and $\mu_0=1$, $\mu_1\geq 1$. So it holds that

$$m_0+m_1=2\leq 1+\mu_1=\sum_{j=0}^{1}\mu_j m_j.$$

For the induction step $i\to i+1$ the multiplicity m_{i+1} is either equal to $m_{i+1}=\mu_i m_i$ or $m_{i+1}=\mu_i m_i+m_{i-1}$. In both cases we have

$$m_i+m_{i+1}\leq \mu_i m_i+m_{i-1}+m_i\leq \mu_i m_i+\sum_{j=0}^{i-1}\mu_j m_j=\sum_{j=0}^{i}\mu_j m_j.$$

□

Lemma 13. *Let q be a point of C, $\mathfrak{z}$ a curve-branch of C centered at q, and p_1, $p_2\in\mathbf{A}^2$ both not equal to q. We define $\mathcal{P}_1$, $\mathcal{P}_2$ as the polar curves of C w.r.t. p_1, p_2, and $\mathcal{L}_1$, $\mathcal{L}_2$ as the lines from q to p_1, p_2, respectively.*

Then for all $T\geq H$, where H is the proper resolution length of $\mathfrak{z}$ within C, it holds that

$$\mathbf{mult}_q^T(\mathfrak{z},\mathcal{P}_1)-\mathbf{mult}_q^T(\mathfrak{z},\mathcal{L}_1)=\mathbf{mult}_q^T(\mathfrak{z},\mathcal{P}_2)-\mathbf{mult}_q^T(\mathfrak{z},\mathcal{L}_2).$$

Proof. Let $f(x,y)\in k[x,y]$ be the defining polynomial of C. W.l.o.g. we can assume that $q=(0,0)$ and we denote the sequence of neighboring points of q by $(q_h)_{0\leq h}$. Let m_h and n_h be the multiplicities of q_h w.r.t. $\mathfrak{z}$ and C. From Theorem 6 we know that the proper multiplicity sequence of $\mathfrak{z}$ can be written as

$$[m_r]^{\mu_r}[m_{r-1}]^{\mu_{r-1}}\cdots[m_2]^{\mu_2}[m_1]^{\mu_1}[m_1]^{\mu_1}[m_0]^{\mu_0}.$$

Furthermore, we can assume that $\mathcal{L}_1$ and $\mathcal{L}_2$ meet transversely at q. (Otherwise, we introduce a third point c, line $\mathcal{L}_3$ and polar curve $\mathcal{P}_3$ which meets both $\mathcal{L}_1$ and $\mathcal{L}_2$ transversely and compare them.)

So we can choose the coordinate system in such a way, that $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{P}_1$, $\mathcal{P}_2$ are defined by $l_1(x,y)=y$, $l_2(x,y)=x$ and $p_1(x,y)=\frac{\partial}{\partial x}f(x,y)$, $p_2(x,y)=\frac{\partial}{\partial y}f(x,y)$, respectively.

Let $\mathfrak{z}_{T,h}$ be the curve-branch $\mathfrak{z}$ truncated at T with parameter $h \in k$. It is obvious that

$$\mathrm{mult}_q(\mathfrak{z}_{T,h}, \mathcal{C}) = \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{C}) = \sum_{k=0}^{T} m_k n_k = M.$$

Using Lemma 2 we get

$$\mathrm{ord}(\frac{d}{dt} f(A_{T,h}(t), B_{T,h}(t))) = M - 1$$

which is equal to

$$\mathrm{ord}(G(t) + H(t)) = M - 1,$$

where

$$\begin{aligned} G(t) &= \frac{\partial}{\partial x} f(A_{T,h}(t), B_{T,h}(t)) \frac{d}{dt} A_{T,h}(t), \\ H(t) &= \frac{\partial}{\partial y} f(A_{T,h}(t), B_{T,h}(t)) \frac{d}{dt} B_{T,h}(t). \end{aligned}$$

Now there are two possibilities

(i) $\mathrm{ord}(G(t)) < M - 1$ or $\mathrm{ord}(H(t)) < M - 1$, or
(ii) $\mathrm{ord}(G(t)) \geq M - 1$ and $\mathrm{ord}(H(t)) \geq M - 1$.

In the first case we then have

$$\mathrm{ord}(G(t)) = \mathrm{ord}(H(t))$$

which implies that

$$\begin{aligned} &\mathrm{ord}(\frac{\partial}{\partial x} f(A_{T,h}(t), B_{T,h}(t))) - \mathrm{ord}(\frac{d}{dt} B_{T,h}(t)) = \\ &\mathrm{ord}(\frac{\partial}{\partial y} f(A_{T,h}(t), B_{T,h}(t))) - \mathrm{ord}(\frac{d}{dt} A_{T,h}(t)). \end{aligned}$$

But this is equal to

$$\mathrm{mult}_q^T(\mathfrak{z}, \mathcal{P}_1) - \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{L}_1) = \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{P}_2) - \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{L}_2).$$

In the second case we have

$$\begin{aligned} \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{P}_1) + \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{L}_2) &\geq M \\ \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{P}_2) + \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{L}_1) &\geq M. \end{aligned}$$

Using the fact that $T \geq K$ it follows that $\mathrm{mult}_q^T(\mathfrak{z}, \mathcal{L}_1) = \mathrm{mult}_q(\mathfrak{z}, \mathcal{L}_1)$ and $\mathrm{mult}_q^T(\mathfrak{z}, \mathcal{L}_2) = \mathrm{mult}_q(\mathfrak{z}, \mathcal{L}_2)$. So we get

$$\mathrm{mult}_q(\mathfrak{z}, \mathcal{P}_1) \geq \mathrm{mult}_q^T(\mathfrak{z}, \mathcal{P}_1) \geq M - \mathrm{mult}_q(\mathfrak{z}, \mathcal{L}_2)$$

Furthermore, we know from Lemma 11 that

$$\mathrm{mult}_q(\mathfrak{z}, \mathcal{P}_1) = \sum_{k=0}^{K-1} m_k(n_k - 1) + \mathrm{mult}_q(\mathfrak{z}, \mathcal{L}_1) - 1$$

and hence

$$\mathrm{mult}_q(\mathfrak{z}, \mathcal{L}_1) + \mathrm{mult}_q(\mathfrak{z}, \mathcal{L}_2) \geq \sum_{k=0}^{K} m_k + (T - K) = \sum_{j=0}^{r} \mu_j m_j + (T - K).$$

Because not both $\mathcal{L}_1$ and $\mathcal{L}_2$ are tangent to $\mathfrak{z}$ this implies that

$$\mu_r m_r + m_{r-1} + m_r \geq \sum_{j=0}^{r} \mu_j m_j + (T - K) + 1$$

which is equivalent to

$$m_{r-1} + m_r \geq \sum_{j=0}^{r-1} \mu_j m_j + (T - K) + 1.$$

But this contradicts Lemma 12 and therefore the second case can never occur. □

Lemma 14. *Let $\mathcal{P} \subset \mathbf{A}^2$ be the polar-curve of $\mathcal{C}$ w.r.t. a generic point p. The numbers $d_{i,j}$ from Definition 52 are equal to*

$$d_{j,i} = \mathrm{mult}_{q_j}^{H_{j,i}}(\mathfrak{z}_{j,i}, \mathcal{P}) - \mathrm{mult}_{q_j}(\mathfrak{z}_{j,i}) + 1.$$

Proof. Using Lemma 13 one can prove this Lemma basically in the same way as Lemma 11. □

Putting these results together, we get the following definition for adjoint curves based on truncated intersection multiplicities.

Theorem 7. *Let $\mathcal{P}$ be the polar-curve of $\mathcal{C}$ w.r.t. a point generic point p. A curve $\mathcal{D}$ is adjoint to $\mathcal{C}$ if*

$$\mathrm{mult}_{q_j}^{H_{j,i}}(\mathfrak{z}, \mathcal{D}) \geq \mathrm{mult}_{q_j}^{H_{j,i}}(\mathfrak{z}, \mathcal{P}) - \mathrm{mult}_{q_j}(\mathfrak{z}) + 1$$

for $1 \leq j \leq J$, $1 \leq i \leq I_j$.

Proof. Follows from Lemma 9 and Lemma 14. □

CHAPTER 4

Newton–Puiseux Algorithm

Let q be a singular point of a curve $\mathcal{C}$. We saw in the last chapter that having computed the singular parts of the Puiseux parametrizations of all curve-branches of $\mathcal{C}$ centered at q is basically the same as having resolved $\mathcal{C}$ at q.

We now show how the singular parts can efficiently be computed by the well-known Newton–Puiseux algorithm. The main part of this chapter then deals with a complexity analysis of the algorithm.

The Newton–Puiseux Algorithm is described in the first section. In Section 2 we point out the strong correspondence between quadratic transformations and Newton transformations. From Section 3 on we concentrate on the computational aspects, namely the degree of algebraic extensions necessary to compute resolution sequences respectively Puiseux parametrizations. In Section 4 a detailed complexity analysis for the Newton–Puiseux algorithm is given which will be the basis of our complexity considerations in Chapter 5.

We define for the whole chapter: Let $f(x, y) \in k[x, y]$ be an irreducible polynomial of degree D defining the curve $\mathcal{C}$. Denoting the x- and y-degree of $f(x, y)$ by M and N respectively, we can write $f(x, y)$ as $f(x, y) = \sum_{i=0}^{M} \sum_{j=0}^{N} a_{i,j} x^i y^j$.

1. Algorithm

We start by describing a rational version of the Newton–Puiseux algorithm as formulated by D. Duval, see [**Duval, 1989**].

There are two basic concepts on which the Newton–Puiseux algorithm is based: Newton polygons and Newton transformations. In the following we define both of them and show how they build up the algorithm.

Definition 55. Let $M_0 = \min(i \mid a_{i,0} \neq 0)$, $M_N = \min(i \mid a_{i,N} \neq 0)$, and $N_0 = \min(j \mid a_{0,j} \neq 0)$. The *extended Newton polygon of* $f(x, y)$ is defined as the left part of the convex hull of the set of points $\{(i, j) \mid a_{i,j} \neq 0\}$ in the (i, j)-plane, going from (M_N, N) to $(M_0, 0)$. The *Newton polygon of* $f(x, y)$ is defined analogously except that it starts at $(0, N_0)$ and ends at $(M_0, 0)$.

Let L be the number of segments of the (extended) Newton-polygon of $f(x, y)$. For each segment S_l, for $1 \leq l \leq L$, we define the numbers p'_l and q'_l as shown in Figure 1. Let $d'_l = \gcd(p'_l, q'_l)$ and $p_l = p'_l/d'_l$, $q_l = q'_l/d'_l$.

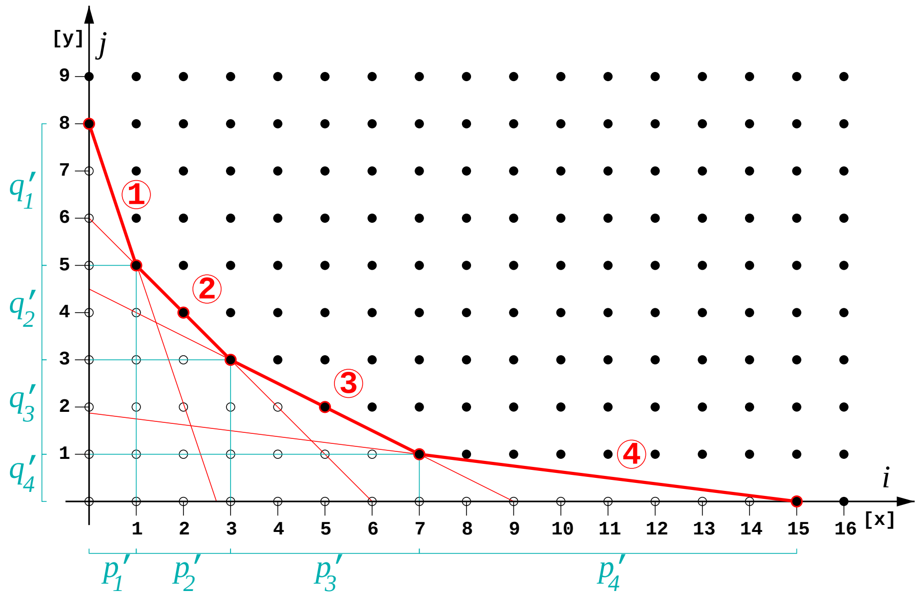


FIGURE 1. Newton polygon

The equation of the line $\mathcal{L}_l$ in the (i,j)-plane associated to S_l is then given by

$$\mathcal{L}_l \colon q_l i + p_l j = r_l.$$

Finally, we define for each segment S_l the numbers a_l and b_l as

$$a_l = \sum_{1 \le l' < l} p'_{l'},$$

$$b_l = \sum_{l < l' \le L} q'_{l'}.$$

As can easily be seen, it holds that $r_l = a_l + p_l + \frac{p_l}{q_l} b_l$, for all $1 \le l \le L$.

Definition 56. A *Newton transformation* τ is defined as a 6-tuple

$$\tau = [p, q, r;\ \zeta;\ u, v],$$

where $p,\ q,\ r \in \mathbb{Z}$, with $q > 0$, $\gcd(p,q) = 1$, $\zeta \in k$, and $u,\ v \in \mathbb{Z}$. The number ζ is called the *coefficient of the Newton transformation* τ. Let $g(x,y) \in k[x,y]$. The *Newton transform of* $g(x,y)$ *w.r.t.* τ, written as $\tau(g(x,y))$, is defined as

$$\tau(g(x,y)) = x^{-r} g(\zeta^{-v} x^q, x^p(\zeta^u + y)).$$

We are now able to define one step in the Newton–Puiseux algorithm. Let $g(x,y) \in k[x,y]$ be a polynomial and S a segment of its (extended) Newton polygon. We assume that the line in the (i,j)-plane associated to S is defined by $qj + pi = r$. We define $\Phi(z) \in k[z]$ as $\Phi(z) = \sum_{(i,j)\in S} a_{i,j} z^{(i-i_0)/q}$, where i_0 is the smallest value of i such that there is a point (i,j) in S. Let $\zeta \in k$ be a root of $\Phi(z)$.

Since $\gcd(p,q) = 1$ there exist $u, v \in \mathbb{Z}$ such that $uq + vp = 1$. The Newton transformation (w.r.t. S and ζ) is then defined as

$$\tau = [p, q, r;\ \zeta;\ u, v],$$

and the transformed polynomial $g_{\text{new}}(x,y)$ as

$$g_{\text{new}}(x,y) = \tau(g(x,y)).$$

We note, that the integers u and v are not uniquely determined. Disregarding the requirement $uq + vp = 1$ and always setting $u = 1$, $v = 0$ yields the classical Newton–Puiseux algorithm.

Example 9. To illustrate the above definitions we consider the Newton polygon of the polynomial

$$g(x,y) = y^8 - xy^5 + x^3y^3 + 2x^5y^2 + x^7y + x^{15}.$$

Its four segments and Newton transformations are defined by the following table (see also Figure 1).

l	d'_l	p_l	q_l	r_l	a_l	b_l	$\Phi_l(z)$	u_l	v_l
1	1	1	3	8	0	5	$z-1$	0	1
2	2	1	1	6	1	3	$-z^2+1$	1	0
3	1	2	1	9	3	1	z^2+2z+1	1	0
4	1	8	1	15	7	0	$z+1$	1	0

Definition 57. Let $T = (\tau_k)_{1 \le k \le K}$ be a sequence of Newton transformations. The *Puiseux parametrization* $\delta(t)$ *determined by* T is defined recursively by

$$\begin{aligned}
(A_K(t), B_K(t)) &= (t, 0) \\
(A_{k-1}(t), B_{k-1}(t)) &= (\zeta^{-v_k} A_k^{q_k}(t), (\zeta^{u_k} + B_k(t)) A_k^{p_k}(t), \text{ for } K \ge k \ge 1. \\
\delta(t) &= (A_0(t), B_0(t)).
\end{aligned}$$

Lemma 15. *The singular part of a Puiseux parametrization consists of all those terms which correspond to Newton transformations* $\tau = [p, q, r;\ \zeta;\ u, v]$*, where* ζ *is a multiple root.*

PROOF. See [**Duval, 1989**]. □

The location of the segment belonging to the first Newton transformations of a curve-branch determines where the curve-branch over $x = 0$ is centered. We can distinguish three types of segments, namely (see Figure 2):

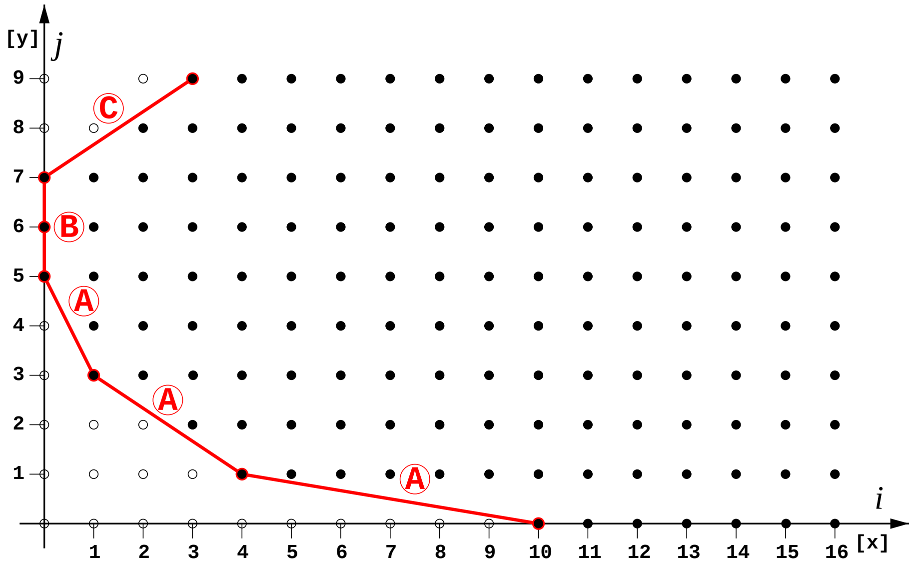


FIGURE 2. Segment-types in Newton polygon

A The curve-branch is centered at the origin.
B The curve-branch is centered at finite distance over $x = 0$ but not at the origin.
C The curve-branch is centered at infinity over $x = 0$.

Types B and C belong to the *extended Newton polygon* and are only of interest for the first Newton transformation.

Definition 58. Let k' be a subfield of k and $q \in \mathbf{A}^2$. A *system of rational Puiseux parametrizations of C centered over $x = 0$ w.r.t.* k' is a set $\{\delta_1(t), \ldots, \delta_I(t)\}$ of non-equivalent Puiseux parametrizations of C centered over $x = 0$ which is invariant under the action of the Galois group $\mathrm{Gal}(k/k')$.

The Puiseux parametrizations δ_i, for $1 \leq i \leq I$, are called *rational Puiseux parametrizations over* k'.

Theorem 8. *The algorithm described in Appendix A Section 3 computes a system of rational Puiseux expansions corresponding to all curve-branches centered at* $x = 0$.

PROOF. See [**Duval, 1989,** Theorem 3]. □

A basic property of a rational Newton–Puiseux algorithm is, that after having computed all singular parts of $f(x, y)$ there is a one-to-one correspondence between the curve-branches at $x = 0$ and the Puiseux parametrizations computed by the algorithm.

Of course, there are infinitely many different rational Puiseux series corresponding to one curve-branch (e.g. obtained by a linear parameter substitution). From now on, whenever we speak about *the* Puiseux parametrization of a curve-branch, we mean the one computed by the algorithm above.

From now on we assume that the curve-branches of $\mathcal{C}$ over $x = 0$ are given by $\mathfrak{z}_i$, for $1 \leq i \leq I = N$.

Definition 59. For each curve-branch $\mathfrak{z}_i$, for $1 \leq i \leq I$, of $\mathcal{C}$ over 0 we define:

$f_{i,k}(x, y)$	The polynomial defined as $\tau_{i,k-1}(f_{i,k-1}(x, y))$, where $f_{i,0}(x, y) = f(x, y)$.
$S_{i,k}$	The segment of the Newton-polygon of $f_{i,k}(x, y)$ belonging to $\mathfrak{z}_i$.
$p_{i,k}$, $q_{i,k}$, $r_{i,k}$, $a_{i,k}$, $b_{i,k}$	Defined as in Definition 55.
$\zeta_{i,k}$, $u_{i,k}$, $v_{i,k}$	Defined as in Definition 56.
$\tau_{i,k}$	The Newton transformation defined by $f_{i,k}(x, y)$ and $S_{i,k}$.
K_i	The number of Newton transformations (i.e. the number of steps in the Newton–Puiseux algorithm) necessary to compute the singular part of $\mathfrak{z}_i$.

It is a consequence of the one-to-one correspondence between curve-branches and rational Puiseux series that the above polynomials, segments, numbers, and Newton transformations are all well and uniquely defined.

2. Newton Transformations

As we saw in Chapter 3 Section 3 the terms of lowest order of a local parametrization determine the first blowups. On the other hand these terms are determined by the first Newton transformation. Therefore we can expect to get a correspondence between subsequences of the resolution sequence and Newton transformations. This is shown in the next Lemma.

Lemma 16. *Let* $\mathfrak{z}$ *be a curve-branch and* $(\tau_1, \ldots, \tau_K)$ *be the sequence of the Newton transformations corresponding to its singular part. Then* $\tau_k = [p_k, q_k, r_k;\ \zeta_k;\ u_k, v_k]$ *determines the multiplicities (and types) of the* $\mu_1 + \cdots + \mu_I$ *neighboring points (and QTs) as follows:*

$$[d_{k+1}m_1]^{\mu_1}[d_{k+1}m_2]^{\mu_2}\cdots[d_{k+1}m_I]^{\mu_I},$$

$$[\mathrm{I}]^{\mu_1}[\mathrm{II}]^{\mu_2}[\mathrm{I}]^{\mu_3}[\mathrm{II}]^{\mu_4}\cdots[a]^{\mu_I-1}[\mathrm{I}],$$

where the numbers μ_i *and* m_i *are defined by*

$$\begin{aligned} m_0 &= \mu_1 m_1 + m_2, \\ m_1 &= \mu_2 m_2 + m_3, \\ &\vdots \\ m_I &= \mu_I m_I, \end{aligned}$$

with $m_0 = p_k$, $m_1 = q_k$. *Further, let* $d_{K+1} = 1$, $d_k = q_k d_{k+1}$, *and* $a = \mathrm{I}$ *if* I *is odd and* $a = \mathrm{II}$ *otherwise.*

Proof. Recalling how τ_k defines the Puiseux parametrization of $\mathfrak{z}$ (Definition 57) we see that p_k and q_k can be seen as part of a characteristic pair (a, b) of $\mathfrak{z}$, in the sense that $a = p_k p'$, $b = q_k + q'$, with p', $q' \in \mathbb{N}$.

The proof follows then from Corollary 1 and Corollary 2 applied to (p_k, q_k). □

Let Σ be the proper resolution sequence of $\mathfrak{z}$ and T be a sequence of Newton transformations for the singular part of $\mathfrak{z}$. According to Lemma 16 we get the following partition for Σ

$$\begin{aligned} \Sigma &= (\sigma_1, \ldots, \sigma_H), \\ T &= (\tau_1, \ldots, \tau_K), \end{aligned}$$

and $\tau_k \sim (\sigma_{h_k}, \ldots, \sigma_{h_{k+1}-1})$, where $h_1 = 1$, $h_{K+1} = H + 1$.

This correspondence can also be used to compute the intersection multiplicity of two curve-branches and the sum over all multiplicities in the proper multiplicity sequence.

Definition 60. Two Newton transformations $\tau = [p, q, r;\ \zeta;\ u, v]$ and $\tau' = [p', q', r';\ \zeta';\ u', v']$ are of the *same type* if $p = p'$, $q = q'$, $\zeta = \zeta'$, $u = u'$, and $v = v'$.

Lemma 17. *Let $\mathfrak{z}$ and $\mathfrak{z}'$ be two curve-branches centered at q and let $(\tau_1, \dots, \tau_K)$ and $(\tau'_1, \dots, \tau'_{K'})$ be the sequences of the Newton transformations corresponding to their singular parts. As usual τ_k and τ'_k are of the forms $\tau_k = [p_k, q_k, r_k;\ \zeta_k;\ u_k, v_k]$ and $\tau'_k = [p'_k, q'_k, r'_k;\ \zeta'_k;\ u'_k, v'_k]$. We assume that the sequences are comparable in a sense that if $p_k = p'_k$ and $q_k = q'_k$ of some k then also $u_k = u'_k$ and $v_k = v'_k$. Let s be defined as the number for which τ_k and τ'_k are of the same type, for $1 \leq k < s$, whereas τ_s and τ'_s are not.*

The intersection multiplicity of $\mathfrak{z}$ and $\mathfrak{z}'$ at q is then equal to

$$\operatorname{mult}_q(\mathfrak{z}, \mathfrak{z}') = \sum_{k=1}^{s-1} d_{k+1} d'_{k+1} p_k q_k + d_{s+1} d'_{s+1} \min(p_s q'_s, p'_s q_s),$$

where the numbers d_k and d'_k are defined by $d_{K+1} = 1$, $d_k = q_k d_{k+1}$, $d'_{K+1} = 1$, $d'_k = q'_k d'_{k+1}$.

Proof. We prove the Lemma using the facts that Newton transformations define a partition of a resolution sequence and that the intersection multiplicity is summable w.r.t. resolution sequences (see Lemma 6 on page 41).

Let M_k be the partial intersection multiplicities of $\mathfrak{z}$ and $\mathfrak{z}'$ determined by τ_k and τ'_k, for $1 \leq k \leq s$. Then we have to show that

$$\operatorname{mult}_q(\mathfrak{z}, \mathfrak{z}') = \sum_{k=1}^{s-1} M_k + M_s,$$

where

$$\begin{aligned} M_k &= d_{k+1} d'_{k+1} p_k q_k = d_{k+1} d'_{k+1} p'_k q'_k, \quad \text{for } 1 \leq k < s, \text{ and} \\ M_s &= d_{s+1} d'_{s+1} \min(p_s q'_s, p'_s q_s). \end{aligned}$$

In Lemma 16 we showed that the partial resolution sequences determined by τ_k and τ'_k are ($m_0 = p_k$, $m_1 = q_k$, $m_{I+1} = 0$, $n_0 = p'_k$, $n_1 = q'_k$, $n_{J+1} = 0$):

$$\begin{aligned} m_0 &= \mu_1 m_1 + m_2 & n_0 &= \nu_1 n_1 + n_2 \\ m_1 &= \mu_2 m_2 + m_3 & n_1 &= \nu_2 n_2 + n_3 \\ &\ \vdots & &\ \vdots \\ m_{I-1} &= \mu_I m_I + m_{I+1} & n_{J-1} &= \nu_J n_J + n_{J+1} \end{aligned}$$

Hence, for $1 \leq k < s$ we get

$$M_k = \sum_{i=1}^{I} \mu_i d_{k+1} m_i d'_{k+1} n_i = d_{k+1} d'_{k+1} \sum_{i=1}^{I} (m_{i-1} - m_{i+1}) m_i = \\ d_{k+1} d'_{k+1} m_0 m_1 = d_{k+1} d'_{k+1} n_0 n_1.$$

For $k = s$, let L be defined such that $\mu_l = \nu_l$, for $1 \leq l < L$, and $\mu_L \neq \nu_L$ (for $I = J$ and $\mu_I = \nu_J$ we set L equal to I). Then $M_s = d_{s+1} d'_{s+1} \sum_{l=1}^{L} N_l$, where N_l is defined according to the following three cases:

$$\begin{aligned} \mu_l > \nu_l &: N_l = \nu_l m_l n_l + m_l n_{l+1} = m_l n_{l-1} \\ \mu_l < \nu_l &: N_l = \mu_l m_l n_l + m_{l+1} n_l = n_l m_{l-1} \\ \mu_l = \nu_l &: N_l = \mu_l m_l n_l = \nu_l m_l n_l \end{aligned}$$

Expanding the sum $\sum_{l=1}^{L} N_l$ we get $\sum_{l=1}^{L} N_l = \min(m_1 n_0, n_1 m_0)$. □

For computing certain invariants of curve-branches (e.g. its conductor number) the following formula will also be helpful.

Lemma 18. *We use the notation of Lemma 16. Let* $(m_h)_{0 \leq h \leq H}$ *be the proper multiplicity sequence of* $\mathfrak{z}$. *Then it holds that*

$$\sum_{h=0}^{H} m_h = \sum_{k=1}^{K} d_{k+1}(p_k + q_k - 1).$$

PROOF. This Lemma can be proved the same way as Lemma 17. □

Looking again at QTs we see that $\sigma_{h_k}, \ldots, \sigma_{h_{k+1}-1}$, for some $1 \leq k \leq K$, can be put together to one transformation, since the transformations σ_{h_k+1}, $\ldots$, $\sigma_{h_{k+1}-1}$ are all centered at the origin. This transformation is called toroidal transformation and in general of the following form.

Definition 61. Let $A = \left(\begin{smallmatrix} q & t \\ p & s \end{smallmatrix}\right)$ be a regular and integral, i.e. $\det(T) = \pm 1$, matrix. The *toroidal transformation (TT)* σ^A is defined as

$$\sigma^A : (x, y) \mapsto (x^q y^t, x^p y^s).$$

Remark: By definition toroidal transformations are birational and so they can be seen as birational counterparts to Newton transformations (Newton transformations are only birational in the case $q = 1$, i.e. no ramification occurs).

Rational Newton–Puiseux algorithms use this to avoid unnecessary field extensions for their coefficients in the Newton transformations.

Resolution algorithms based on toroidal transformations are described in [**Kozen, 1994**] and [**Oka, 1996**].

3. Algebraic Extensions

Before we analyze the computational complexity of the Newton–Puiseux algorithm we compare the algebraic extensions that are needed to compute the resolution sequence of a curve-branch and the singular part of its Puiseux parametrization (given a not necessarily algebraically closed ground field).

Given a curve-branch $\mathfrak{z}$, we will see that the degree of an algebraic extension needed to resolve $\mathfrak{z}$ by quadratic transformations is always less than or equal to the degree of an algebraic extension resulting from a Puiseux parametrization of $\mathfrak{z}$.

On the other hand, one can show there always a exists a (rational) Puiseux parametrization of $\mathfrak{z}$ where the degree of its algebraic extension is equal to the one of its resolution sequence. Note, that this is not true for classical Puiseux parametrizations, as can be seen from the example $f(x,y) = 8y^2 - x^3$. A classical Puiseux parametrization of $f(x,y)$ centered at the origin is $(t^2, \frac{1}{\sqrt{8}}t^3)$, whereas a rational one is given by $(2t^2, t^3)$.

For this section, we assume that the curve $\mathcal{C}$ is defined over k', i.e. $f(x,y) \in k'[x,y]$, where we explicitly do not require that k' is an algebraically closed field.

Let $\mathfrak{z}$ be a curve-branch of $\mathcal{C}$, where we assume w.l.o.g. that $\mathfrak{z}$ is centered at the origin. By $\Sigma = (\sigma_1, \dots, \sigma_H)$ we denote the resolution sequence of $\mathfrak{z}$ (as a curve-branch of $\mathcal{C}$) and by $T = (\tau_1, \dots, \tau_K)$ a sequence of Newton transformations necessary to compute the singular part of $\mathfrak{z}$ (w.r.t. $\mathcal{C}$). Further, let $\delta(t) = (\sum_{k>0} a_k t^k, \sum_{k>0} b_k t^k)$ be the Puiseux parametrization of $\mathfrak{z}$ corresponding to T. We define the coefficient fields of Σ, T, and $\delta(t)$ w.r.t. k' as follows.

Definition 62. The *coefficient fields* k^Σ*,* k^T*, and* k^δ are defined as

$$\begin{aligned} k^\Sigma &= k'(\eta_1, \dots, \eta_H), \\ k^T &= k'(\zeta_1, \dots, \zeta_K), \\ k^\delta &= k'(a_0, b_0, a_1, b_1, \dots). \end{aligned}$$

It clearly holds that $k^T = k^\delta$.

3.1. Quadratic Transformations

Lemma 19. *It holds that* $k^\Sigma \subseteq k^\delta$*.*

PROOF. Let $\mathfrak{z}_0 = \mathfrak{z}$ and let $\delta_0(t) = (A_0(t), B_0(t)) = \delta(t)$. We show that $\eta_h \in k^\delta$, for all $1 \leq h \leq H$, and therefore $k^\Sigma \subseteq k^\delta$. Using induction on h we assume that $\eta_1, \dots, \eta_{h-1} \in k^\delta$ and show that $\eta_h \in k^\delta$.

Since $\mathfrak{z}$ is centered at the origin it holds that $\eta_0 = 0 \in k^\delta$. For $h > 0$ we get the following: Depending on which type σ_h is, the proper transform $\mathfrak{z}_h$ of $\mathfrak{z}_{h-1}$ defined by $\delta_h(t) = (A_h(t), B_h(t))$ is of the form:

- The QT σ_h is of type I:

$$\begin{aligned} A_h(t) &= A_{h-1}(t), \\ B'_h(t) &= \frac{B_{h-1}(t)}{A_{h-1}(t)}, \\ B_h(t) &= B'_h(t) - B'_h(0). \end{aligned}$$

  It surely holds that $A_h(t)$, $B'_h(t)$, $B_h(t) \in k^\delta[[t]]$. Since $\eta_h = B'_h(0)$ we have $\eta_h \in k^\delta$.

- The QT σ_h is of type II:

$$\begin{aligned} A_h(t) &= \frac{A_{h-1}(t)}{B_{h-1}(t)}, \\ B_h(t) &= B_{h-1}(t). \end{aligned}$$

  In this case, since $\mathrm{ord}(A_{h-1}(t)) > \mathrm{ord}(B_{h-1}(t))$ we have $\eta_h = A_h(0) = 0 \in k^\delta$.

□

3.2. Puiseux Series Expansion

First prove a Lemma concerning reparametrizations of power-series over non-algebraically closed fields.

Lemma 20. *Let k^* be a subfield of k which is not necessarily algebraically closed. Given a power series $A(t) = \sum_{k\geq 0} a_k t^k \in k^*[[t]]$ with $\mathrm{ord}(A(t)) = m \geq 1$. There exists a power series $B(t) \in k^*[[t]]$ with $\mathrm{ord}(B(t)) = 1$ such that $A(B(t)) = a_m t^m$.*

Proof. Let $A(t) = \sum_{k\geq m} a_k t^k$, $a_m \neq 0$ and let $B(t) = \sum_{k\geq 1} b_k t^k$. For $A(B(t)) = \sum_{k\geq m} c_k t^k$ we require that

$$c_k = \begin{cases} a_k & \text{if } k = m, \\ 0 & \text{if } k > m. \end{cases}$$

For $k = m$, since $c_m = a_m b_1^m$ we can set $b_1 = 1$.

For $k > m$, the coefficients c_k are equal to

$$\begin{aligned} c_k &= \sum_{i=m}^{k} a_i\, f_{k,i} \quad \text{, where} \\ f_{k,i} &= \sum_{\substack{i_1+2\,i_2+\cdots+k\,i_k=k,\\ i_1+i_2+\cdots+i_k=i}} \left(\frac{i!}{i_1!\; i_2! \cdots i_k!}\right) b_1^{i_1} b_2^{i_2} \cdots b_k^{i_k}. \end{aligned}$$

From this it can be seen that $f_{k,i}$ depends on b_i, for $1 \leq i \leq k-i+1$, and therefore c_k depends on b_i, for $1 \leq i \leq k-m+1$. The coefficient b_{k-m+1} occurs exactly once in the sum c_k, namely as $a_m m b_1^{m-1} b_{k-m+1}$. Hence, setting $c_k = 0$ one gets a *linear* equation for b_{k-m+1}.

The coefficients of $B(t)$ can now be obtained iteratively by setting $c_k = 0$ and solving for b_{k-m+1}, for $k = m+1, m+2, \dots$. ☐

As can be seen from the proof of Lemma 20 in general there does not exist a power-series $B(t) \in k^*[[t]]$ such that $A(B(t)) = t^m$, because this would require setting $b_1 = (a_m)^{\frac{1}{m}}$ which may not be possible in k^*. This fact corresponds exactly to the difference of classical Puiseux parametrizations and Puiseux parametrizations.

Lemma 21. *There exists a Puiseux parametrization $\delta(t)$ of $\mathfrak{z}$ such that $k^\delta \subseteq k^\Sigma$.*

Proof. Let $\bar{f}(x, y)$ be the defining polynomial of the resolved curve-branch $\mathfrak{z}_H$. By definition it holds that $\bar{f}(x, y) \in k^\Sigma[x, y]$, $\bar{f}(x, y)$ has normal-crossings at the origin, and $\operatorname{ord}(\bar{f}(0, t)) = 1$. Hence a local parametrization $(A'_H(t), B'_H(t))$ of $\bar{f}_H$ can be computed using the *implicit function theorem.* So we know that $A'_H(t)$, $B'_H(t) \in k^\Sigma[[t]]$ and $\operatorname{ord}(A'_H(t)) = 1$. According to Lemma 20 there exists a reparametrization $T_H(t) \in k^\Sigma[[t]]$ such that

$$\begin{aligned} A_H(t) &= A'_H(T(t)) = a_{H,1} t, \\ B_H(t) &= B'_H(T(t)). \end{aligned}$$

But this means that we have found a Puiseux parametrization $\delta_H(t) = (A_H(t), B_H(t))$ of $\mathfrak{z}_H$ such that $k^{\delta_H} \subset k^\Sigma$.

Now assume, for some $1 \leq h \leq H$, we already computed a Puiseux parametrization $\delta_h(t) = (A_h(t), B_h(t))$ for $\mathfrak{z}_h$ such that $k^{\delta_h} \subseteq k^\Sigma$.

A local parametrization of $\mathfrak{z}_{h-1}$ is then either given by:

- The QT σ_h is of type I:

$$\begin{aligned} A'_{h-1}(t) &= A_h(t) \\ B'_{h-1}(t) &= A_h(t)(B_h(t) - \xi_h) \end{aligned}$$

- The QT σ_h is of type II:

$$\begin{aligned} A'_{h-1}(t) &= B_h(t)(A_h(t) - \xi_h) \\ B'_{h-1}(t) &= B_h(t). \end{aligned}$$

In both cases there exists a reparametrization $T_{h-1}(t) \in k^\gamma[[t]]$ such that $\delta_h(t) = (A'_{h-1}(T(t)), B'_{h-1}(T(t)))$ is a Puiseux parametrization of $\mathfrak{z}_h$ and $k^{\delta_h} \subseteq k^\Sigma$.

In particular this is true for $h = 0$ and therefore we have shown that there exists a Puiseux parametrization $\delta_0(t)$ of $\mathfrak{z}$ whose coefficient field is a subfield of k^Σ. □

The next corollary summarizes the results of the previous and this subsection.

Corollary 4. *There exists a Puiseux parametrization* $\delta(t)$ *of* $\mathfrak{z}$ *such that* $k^\delta = k^\Sigma$.

PROOF. Follows from Lemma 19 and Lemma 21. □

3.3. Degree of Algebraic Extension

Let G be the Galois group of k over k', i.e. $G = \mathrm{Gal}(k/k')$. For a system $\Delta = \{\delta_1(t), \dots, \delta_I(t)\}$ of rational Puiseux parametrizations of C over k' let k^{δ_i}, for $1 \le i \le I$, be the corresponding coefficient field of $\delta_i(t)$. We define G as the total number of G-orbits in Δ and the numbers d_i as $d_i = [k^{\delta_i} : k]$.

If we assume that δ_g, for $1 \le g \le G$, belong to different G-orbits, then it holds that $N = \sum_{1 \le g \le G} d_g e_g$, since curve-branches defined by parametrizations in the same G-orbit have the same ramification index.

Lemma 22. *The degree of an algebraic extension necessary to resolve the curve-branch* $\mathfrak{z}$ *of the curve* C *by QTs is less than or equal to the* y*-degree of the defining polynomial of* C.

PROOF. From the previous subsection we know that there exists a sequence T of Newton transformations such that $d = [k^\Sigma : k] = [k^T : k]$. Using the fact that $de \le N$, where e is the ramification index of $\mathfrak{z}$ and N is the y-degree of the defining polynomial of C, we have $d \le N$. □

A direct consequence of Lemma 22 is that at most $\log(N)$ field extensions are necessary to resolve a curve-branch (or to compute the singular part of its rational Puiseux parametrization).

4. Computational Complexity

In this section we give a detailed complexity analysis (in terms of operations in the ground field k') for the Newton–Puiseux algorithm described in Section 1.

There are three quantities which basically contribute to the complexity of the algorithm, namely:

- The number of Newton transformations necessary to compute the singular part of a Puiseux series of $f(x, y)$.
- The x- and y-degrees of the transformed polynomials.
- The degree of the algebraic extension necessary to represent the coefficients of a Puiseux series of $f(x, y)$.

We do not consider the size of the coefficients occurring during computation. A bound for the coefficients of Puiseux series and a bit-complexity analysis can be found in [**Walsh, 1998**] and [**Walsh, 1999**].

For each one of the above three quantities there are rather obvious bounds:

- The number of Newton transformations for computing the singular part is bounded by MN.
- The x- and y-degrees of the transformed polynomials are bounded by MN^2 and N, respectively.
- The degree of the algebraic extension is at most N.

Putting these bounds together it follows that the singular parts of all curve-branches of $\mathcal{C}$ over $x = 0$ can be computed requiring no more than M^2N^6 operations in the ground field k', see [**Duval, 1989**]. But this complexity bound has two disadvantages:

First, it is a global bound, which means that the complexity of the singularity, measured by some invariant as e.g. its Milnor-number, is neglected in the analysis. So we get the same complexity bound for an ordinary double point as for a very complicated singularity having a long resolution sequence. Complexity considerations based on local data can be found in [**Henry and Merle, 1987**] and [**Teitelbaum, 1990**]. In both articles the authors use the Milnor-number as a measure of the complexity of a singularity and resolve the singularity by quadratic transformations.

Second, it is not taken into consideration that not all three quantities mentioned above can contribute to the complexity at the same time. If a resolution of a singularity yields an algebraic extension of high degree, the number of Newton transformations to compute its singular parts has to be small.

Taking the above two points into account and making use of the close connection between quadratic transformations and Newton transformations, we are able to show that computing the singular parts of all curve-branches of $\mathcal{C}$ over centered at some point q takes no more than $N\mu^2$ operations in the ground field, where μ is again its Milnor-number.

We split the complexity analysis of the Newton–Puiseux algorithm into three parts: In Subsection 4.2, we derive an upper bound for the number of operations w.r.t. the algebraically closed field k, where we additionally assume that computing all roots of a univariate polynomial is a basic operation in k. Since we are interested in a complexity bound w.r.t. the number of operations in the ground field k' we also have to consider the additional complexity caused by computing with algebraic numbers. In Subsection 4.3 we show how this can be done using passive factorization. The results of these two subsections are then used in Subsection 4.4 to compute a complexity bound w.r.t. the ground field k'. During the analysis we neglect additions and subtractions, whereas equality tests play an essential role (due to the use of passive factorization – splitting of computation) and have to be taken into account.

In Subsection 4.5 we deal with the problem of how to make practical use of the complexity bounds derived in the previous subsections, since they are based on data (namely again the Milnor-number) we do not know in advance. As we will show, this can be solved using passive expansion in the algorithm.

Finally, in Subsection 4.6 we try to compare this complexity analysis to the work of other authors.

We start with answering the question of how many Newton transformations it may take at most to compute the singular part of a curve-branch.

4.1. Number of Newton Transformations

The following Lemma gives two upper bounds for the maximal number of Newton transformations necessary for computing the singular part of a Puiseux parametrization, namely a global one and a local one. We also show that both bounds are of optimal order.

Lemma 23. *The number* K *of Newton transformations necessary to compute the singular part of the Puiseux parametrization of a curve-branch* $\mathfrak{z}$ *of* C *is bounded by*

$$K \leq 2(N-1)M + 1 \tag{1}$$

or

$$K \leq \mu, \tag{2}$$

where μ *is the Milnor-number of* $\mathfrak{z}$.

Proof. The proof of (1) is given in [**Duval, 1989**]. Inequality (2) follows from the correspondence of Newton transformations and quadratic transformations. Let $(m_h)_{0\leq h}$ be the multiplicity sequence of $\mathfrak{z}$. As we have seen in Section 2 each Newton transformation in the singular part of the algorithm represents at least one quadratic transformation involving a neighboring point of multiplicity greater than or equal to 2. Hence $K < \sum_{0\leq h} m_h(m_h - 1) = \mu$. $\square$

A degree bound for the singular part of a curve-branch of $f(x, y)$ is also given in the paper of Duval, [**Duval, 1989**, page 151]. There it says that the singular part is at most of degree $2(N-1)M$.

A direct consequence of Lemma 23 is that K also bounded by $D(D-1)$.

We will show now, that these bounds are up to multiplicative constants and terms of smaller order optimal. To do so, we define two families of polynomials (n, $k \in \mathbb{N}$):

$$\begin{aligned} f_{n,k}(x, y) &= y^{4n+2} + 2xy^{2n+1} + x^2 + x^{2k+3} \\ g_n(x, y) &= f_{n,2n-1}(x, y) \end{aligned}$$

The polynomials (respectively the curves they define) have the following properties:

(i) The degrees of $f_{n,k}(x, y)$ and $g_n(x, y)$ are given by:

	M (x-degree)	N (x-degree)	D (total-degree)
$f_{n,k}(x, y)$	$2k + 3$	$4n + 2$	$\max(2k + 3, 4n + 2)$
$g_n(x, y)$	$4n + 1$	$4n + 2$	$4n + 2$

(ii) The Newton-transformations for $f_{n,k}(x, y)$ and $f_n(x, y)$ are

$$\begin{aligned} \tau_1 &= [1, 2n + 1, 4n + 2;\ -1, 0, 1], \\ \tau_2 &= [(2n + 1)(2k + 1), 2, (4n + 2)(2k + 1); \\ &\qquad (2n + 1)^{-2}, -(2nk + n + k), 1], \end{aligned}$$

and

$$\begin{aligned} \tau_1 &= [1, 2n + 1, 4n + 2;\ -1, 0, 1], \\ \tau_2 &= [8n^2 + 2n - 1, 2, 16n^2 + 4n - 2; \\ &\qquad (2n + 1)^{-2}, -(4n^2 + n - 1), 1]. \end{aligned}$$

(iii) The curves defined by $f_{n,k}(x, y)$ and $g_n(x, y)$ have one curve-branch centered at $(0, 0)$ whose Puiseux parametrizations are

$$\begin{aligned} f_{n,k}(x, y) : x(t) &= -(2n + 1)^{4n+2} t^{4n+2}, \\ y(t) &= (2n + 2)^2 t^2 + (2n + 1)^{(2n+1)(2k+1)+1} t^{(2n+1)(2k+1)+2}, \end{aligned}$$

and

$$\begin{aligned} g_n(x, y) : x(t) &= -(2n + 1)^{4n+2} t^{4n+2}, \\ y(t) &= (2n + 2)^2 t^2 + (2n + 1)^{2n(4n+1)} t^{2n(4n+1)+1}. \end{aligned}$$

(iv) The proper multiplicity sequences of the above curve-branches are

$$f_{n,k}(x,y) : [4n+2]^0[2]^{2n+1}[2]^{(2k+1)n+k}[1]^2,$$
$$g_n(x,y) : [4n+2]^0[2]^{2n+1}[2]^{n(4n+1)-1}[1]^2,$$

and the free and satellite neighboring points are given by

$$f_{n,k}(x,y) : [F][S]^{2n}[F]^{(2k+1)n+k+1}[S][F],$$
$$g_n(x,y) : [F][S]^{2n}[F]^{n(4n+1)}[S][F].$$

(v) The Milnor-numbers of the singularities at $(0,0)$ of $f_{n,k}(x,y)$ and $g_n(x,y)$ are

$$f_{n,k}(x,y) : \mu = (2n+1)k + 3n + 1,$$
$$g_n(x,y) : \mu = 4n^2 + 3n.$$

(vi) The curve defined by $g_n(x,y)$ has only one singularity, namely $(0,0)$, and its genus is

$$g = \frac{(4n+1)4n}{2} - (4n^2+3n) = 4n^2 - n.$$

If we now add some "suitable" terms of "suitable" high degree to $f_{n,k}(x,y)$ and $g_n(x,y)$ we can achieve that the free neighboring points get coordinates unequal to zero. But this results in non-zero coefficients for the Puiseux series and hence the number of Newton transformations is equal to the number of free neighboring points. A suitable modification term is

$$h_n(x,y) = (4n+2)(x^2y - xy^{2n+2}) + (2n+1)^2x^2y^2$$

and the modified polynomials are

$$\tilde{f}_{n,k}(x,y) = f_{n,k}(x,y) + h_n(x,y),$$
$$\tilde{g}_n(x,y) = g_n(x,y) + h_n(x,y).$$

So to summarize, we have constructed a family of curves with increasing genus where each curve has a singularity at $(0,0)$. Further, the number of Newton transformations necessary to resolve the singularity at $(0,0)$ is of order MN and μ respectively.

4.2. Number of Operations in the Algebraically Closed Field k

Let $q = (0,0)$ and let $\mathfrak{z}_i$, for $1 \leq i \leq I$, be the curve-branches of $\mathcal{C}$ centered at q. By μ_q we denote the Milnor-number of $\mathcal{C}$ at q. The sequence of proper neighboring points and the proper multiplicity sequence of $\mathfrak{z}_i$ are denoted by $(q_{i,h})_{0\leq h\leq H_i}$ and $(m_{i,h})_{0\leq h\leq H_i}$, for $1 \leq i \leq I$, respectively. For each curve-branch $\mathfrak{z}_i$ we also define the index-set of free- and satellite-neighboring points and denote them by H_i^F and H_i^S.

In the following we use the notation from Definition 59.

A key-point in the complexity analysis below is a bound for the x-degrees of the polynomials $f_{i,k}(x,y)$ necessary to compute the singular part of the curve-branches.

Lemma 24. *The singular part of the Puiseux series of $\mathfrak{z}_i$, for $1 \leq i \leq I$, remains unchanged if the polynomials $f_{i,k}(x,y)$, for $0 \leq k \leq K_i$, are truncated at x-degrees $M_{i,k}$, where*

$$M_{i,k} = r_{i,k} + \frac{r_{i,k+1}}{q_{i,k}} + \frac{r_{i,k+2}}{q_{i,k}q_{i,k+1}} + \frac{r_{i,k+3}}{q_{i,k}q_{i,k+1}q_{i,k+2}} + \cdots + \frac{r_{i,K_i}}{q_{i,k}\cdots q_{i,K_i-1}}.$$

Proof. The singular part of the Puiseux series of $\mathfrak{z}_i$ does not change if the polynomial $f_{i,K_i}(x,y)$ is truncated at x-degree $M_{i,K_i} = r_{i,K_i}$. Since terms of higher x-degree do not contribute to terms of lower x-degree when a Newton transformation is applied, it follows in general that $f_{i,k}(x,y)$ can be truncated at x-degree $r_{i,k} + M_{i,k+1}/q_{i,k}$ without changing the singular part of the Puiseux series. So we get

$$M_{i,k} = r_{i,k} + \frac{M_{i,k+1}}{q_{i,k}}.$$

Unfolding the recurrence concludes the proof. □

The next Lemma shows that the numbers $M_{i,k}$ are less than or equal to certain sums of multiplicities in the multiplicity-graph of q. This enables us to give an upper bound for the total number of coefficients occurring in the singular part computation in terms of μ and N.

First, we have to define certain index-subsets of the neighboring-points of q. For this, we recall that a Newton transformation can be viewed as a concatenation of one or more QTs. Hence, whenever we speak of a neighboring point belonging to a Newton transformation, we mean a neighboring point of a QT which is part of a Newton transformation.

Definition 63. For a curve-branch $\mathfrak{z}_i$, $1 \leq i \leq I$, and a Newton transformation $\tau_{i,k}$, $0 \leq k \leq K_i$, we define the *index-set of $\tau_{i,k}$-free neighboring* *points*, written as $H^F_{i,k}$, as

$$H^F_{i,k} = \{h \mid h \in H^F_i \text{ and } q_{i,h} \text{ belongs to } \tau_{i,k}\}.$$

Let i^* be such that $1 \le i^* \le I$. The *index-set of $\tau_{i,k}$-free neighboring points restricted to the curve-branch $\mathfrak{z}_{i^*}$*, written as $H^F_{i,k;i^*}$, is the set defined as

$$H^F_{i,k;i^*} = \{h \mid h \in H^F_{i,k} \text{ and } q_{i,h} = q_{i^*,h}\}.$$

Example 10. We investigate the singularity consisting of the five curve-branches gives by

$$\begin{aligned} z_1\colon\ x(t) &= t^6 & \qquad z_4\colon\ x(t) &= t^4\\ y(t) &= t^{15} + t^{23}, & y(t) &= t^4 + t^7,\\ z_2\colon\ x(t) &= t^6 & z_5\colon\ x(t) &= t^4\\ y(t) &= t^{15} + 2t^{23}, & y(t) &= t^4 + t^9.\\ z_3\colon\ x(t) &= t^6 & &\\ y(t) &= t^{15} + t^{19}, & & \end{aligned}$$

Its multiplicity-graph is shown in Figure 3.

The following table shows the index-sets of $\tau_{i,k}$-free neighboring points.

$H^F_{i,k}$	z_1	z_2	z_3	z_4	z_5
0	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0\}$	$\{0\}$
1	$\{4,5,6\}$	$\{4,5,6\}$	$\{4,5\}$	$\{1\}$	$\{1,2\}$

The index-sets of $\tau_{i,k}$-free neighboring points restricted to z_3 are given by the table below.

$H^F_{i,k;3}$	z_1	z_2	z_3	z_4	z_5
0	$\{0,1,2\}$	$\{0,1,2\}$	$\{0,1,2\}$	$\{0\}$	$\{0\}$
1	$\{4,5\}$	$\{4,5\}$	$\{4,5\}$	$\{\}$	$\{\}$

Lemma 25. *For each curve-branch $\mathfrak{z}_i$, $1 \le i \le I$, and each step k, $0 \le k \le K_i$, in the Newton–Puiseux algorithm it holds that*

$$r_{i,k} \le \sum_{i'=1}^{I} \sum_{h \in H^F_{i',k;i}} m_{i',h}.$$

Proof. We fix one curve-branch $\mathfrak{z}_{i_0}$ and one step k_0. To simplify notation, we define (see Definition 59): $p_i = p_{i,k_0}$, $q_i = q_{i,k_0}$, $r_i = r_{i,k_0}$, $a_i = a_{i,k_0}$, $b_i = b_{i,k_0}$, and $H_i = H^F_{i,k_0;i_0}$, for $1 \le i \le I$. It then holds that

$$r_{i_0} = a_{i_0} + p_{i_0} + \frac{p_{i_0}}{q_{i_0}} b_{i_0},$$

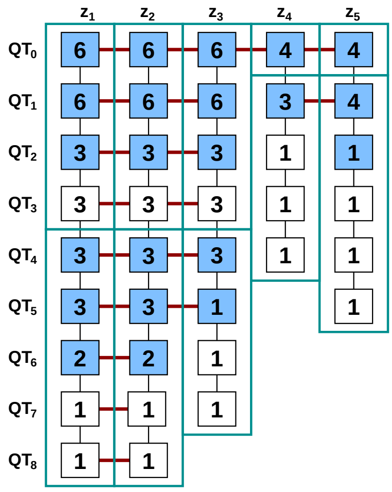


FIGURE 3. Multiplicity graph showing free and satellite points

and we have to show that

$$r_{i_0} \leq \sum_{i=1}^{I} \sum_{h \in H_i} m_{i,h}.$$

Furthermore, let $\mu_i = \left\lfloor \frac{p_i}{q_i} \right\rfloor$ and $s_i = p_i - \mu_i q_i$. We assume, that the curve-branches are labeled in such a way that the sequence $\left(\frac{p_i}{q_i}\right)_{1 \leq i \leq I}$ of rational numbers is increasing. This is equivalent to saying that the curve-branches are labeled according to the Newton-polygon.

Note, that we cannot require that the sequence $\left(\frac{p_i}{q_i}\right)_{1 \leq i \leq I}$ is strictly increasing since it may be the case, that two or more curve-branches still correspond to the same segment after k_0 steps in the algorithm.

In any case we have

$$\sum_{i=1}^{i_0-1} p_i \geq a_{i_0},$$
$$\sum_{i=i_0+1}^{J} q_i \geq b_{i_0}.$$

As one can conclude from Lemma 4 and Lemma 16 the multiplicities of the sequences of free neighboring points belonging to the Newton transformations τ_{i,k_0} are of the form

$$\begin{aligned} &[q_i]^{\mu_i} &&, \text{ if } s_i = 0, \\ &[q_i]^{\mu_i}[s_i] &&, \text{ if } s_i \neq 0. \end{aligned}$$

Because the μ_i form an ascending sequence, there exists an index I^* with $i_0 \le I^* \le I$ such that

$$\begin{aligned} \mu_i \le \mu_{i_0} &\quad , \text{for } 1 \le i \le I^*, \\ \mu_{i_0} < \mu_i &\quad , \text{for } I^* < i \le I. \end{aligned}$$

The multiplicities of the sequences of the neighboring points $q_{i,h}$, for $1 \le i \le I$, $h \in H_i$, are then of the form

$$\begin{aligned} [q_i]^{\mu_i} &\quad , \text{for } 1 \le i \le I^* \text{ and } s_i = 0, \\ [q_i]^{\mu_i}[s_i] &\quad , \text{for } 1 \le i \le I^* \text{ and } s_i \ne 0, \\ [q_i]^{\mu_{i_0}}[q_i] &\quad , \text{for } I^* < i \le I. \end{aligned}$$

Summing over all curve-branches yields

$$\begin{aligned} \sum_{i=1}^{I} \sum_{h \in H_i} m_{i,h} &= \sum_{i=1}^{I^*} (\mu_i q_i + s_i) + \sum_{i=I^*+1}^{I} (\mu_{i_0} q_i + q_i) \\ &= \sum_{i=1}^{I^*} p_i + (\mu_{i_0} + 1) \sum_{i=I^*+1}^{I} q_i \\ &= \sum_{i=1}^{i_0-1} p_i + p_{i_0} + \sum_{i=i_0+1}^{I^*} \frac{p_i}{q_i} q_i + (\mu_{i_0} + 1) \sum_{i=I^*+1}^{I} q_i. \end{aligned}$$

Using the inequalities $\frac{p_i}{q_i} \ge \frac{p_{i_0}}{q_{i_0}}$, for $i_0 < i \le I^*$, and $\mu_{i_0} + 1 \ge \frac{p_{i_0}}{q_{i_0}}$ we get

$$\begin{aligned} \sum_{i=1}^{i_0-1} p_i + p_{i_0} + \sum_{i=i_0+1}^{I^*} \frac{p_i}{q_i} q_i + (\mu_{i_0} + 1) \sum_{i=I^*+1}^{I} q_i &\ge \\ \sum_{i=1}^{i_0-1} p_i + p_{i_0} + \frac{p_{i_0}}{q_{i_0}} \sum_{i=i_0+1}^{I} q_i &\ge \\ a_{i_0} + p_{i_0} + \frac{p_{i_0}}{q_{i_0}} b_{i_0} &= r_{i_0}. \end{aligned}$$

□

Corollary 5. *For all* $1 \le i \le I$ *and* $0 \le k \le K_i$ *it holds that*

$$M_{i,k} \le \sum_{k'=k}^{K_i} \sum_{i'=1}^{I} \sum_{h \in H^{\mathbf{F}}_{i',k';i}} m_{i',h}$$

and

$$M_{i,k} \le \mu_q + 1,$$

where μ_q *is the Milnor-number of* q.

PROOF. We fix one curve-branch $\mathfrak{z}_{i_0}$ and one step k_0. Since all $q_{i_0,k} \geq 1$, for $0 \leq k \leq K_{i_0}$, we have

$$M_{i_0,k_0} \leq \sum_{k=k_0}^{K_{i_0}} r_{i_0,k} = \sum_{k=k_0}^{K_{i_0}} \sum_{i=1}^{I} \sum_{h \in H^F_{i,k;i_0}} m_{i,h}.$$

To prove the second inequality we recall that (using the notation of the proof of Lemma 25)

$$\sum_{k=k_0}^{K_{i_0}} \sum_{i=1}^{I} \sum_{h \in H^F_{i,k;i_0}} m_{i,h} \leq \sum_{k=k_0}^{K_{i_0}} \sum_{i=1}^{I} (\mu_{i,k} q_{i,k} + s_{i,k}),$$

Since

$$\mu_q \geq \sum_{k=k_0}^{K_{i_0}} \sum_{i=1}^{I} (\mu_{i,k} q_{i,k}(q_{i,k} - 1) + s_{i,k}(s_{i,k} - 1))$$

it follows from

$$\mu_{i,k} q_{i,k}(q_{i,k} - 1) + s_{i,k}(s_{i,k} - 1) \geq \mu_{i,k} q_{i,k} + s_{i,k},$$

for $q_{i,k} \geq 2$, $s_{i,k} \geq 2$, and

$$\mu_{i,k} q_{i,k}(q_{i,k} - 1) \geq \mu_{i,k} q_{i,k},$$

for $q_{i,k} \geq 2$, that

$$\mu_q \geq M_{i,k} - 1.$$

□

Disregarding any computational complexity due to algebraic extensions of the ground field as we do now, there is basically one subroutine in the algorithm which contributes to the complexity at each step, namely the procedure `new-polynomial` (see [**Duval, 1989**]) for a detailed description of the algorithm).

In `new-polynomial` there are at most $(M_{i,k} + 1)(N + 1)$ coefficient multiplications (and additions) to do. Therefore we get the following bound for the number of k-operations necessary to resolve one curve-branch.

Lemma 26. *To compute the singular part of the Puiseux expansion of one curve-branch $\mathfrak{z}_i$, $1 \leq i \leq I$, at most $\mu_q(\mu_q + 2)(N + 1)$ operations in k are needed. (Not considering any root computation of univariate polynomials.)*

PROOF. The number C of operations w.r.t. k is bounded by

$$C \leq \sum_{k=1}^{K_i} (M_{i,k} + 1)(N + 1).$$

From Lemma 23 and Corollary 5 we know that $K_i \leq \mu_q$ and $M_{i,k} \leq \mu_q + 1$, so we get

$$C \leq \mu_q(\mu_q + 2)(N+1).$$

$\square$

Restricting ourselves to the case of irreducible singularities the above Lemma would immediately also give a complexity bound for the number of ground field operations, since in this case no algebraic extensions can occur.

Taking into account that the sum of the multiplicities with which the x-degree bounds can be bounded, is non-overlapping for different curve-branches, the complexity bound for resolving one curve-branch at q directly extends to a complexity bound for resolving all curve-branches at q.

Lemma 27. *At most $2\mu_q(\mu_q+2)(N+1)$ operations in k are needed to compute the singular parts of all Puiseux expansion with origin q. (Not considering any root computation of univariate polynomials.)*

Proof. The lemma follows from the fact that the index-sets $H^F_{i,k;i'}$ are essentially non-overlapping in the following sense: If two such index-sets have an index in common, then also the computation in the Newton–Puiseux algorithm is the same for the corresponding neighboring-points (i.e. the curve-branches have not yet been separated and are hence treated together).

But this means, that each free neighboring point of q contributes to the sum of multiplicities only once. Therefore the sum of all x-degrees of the transformed polynomials is less than or equal to $\sum_{i=1}^{I} \sum_{h \in H_i^F} m_{i,h}$.

Using the same argumentation as in the proof of Corollary 5 one can show, that this sum is bounded by $2\mu_q$. $\square$

The next Corollary is not essential for the complexity analysis, but shows how the sum of multiplicities over all free neighboring points is related to the sum over all neighboring points.

Corollary 6. *For each curve-branch $\mathfrak{z}_i$, $1 \leq i \leq I$, it holds that*

$$\sum_{h \in H_i^F} m_{i,h} = \sum_{h=1}^{H_i} m_{i,h} + 1.$$

Proof. Follows from the fact that $m_{i,0} = \sum_{h \in H_i^S} m_{i,h} + 1$ (see Theorem 6 and Lemma 4). $\square$

A direct consequence of Lemma 25 is the following Corollary from deformation theory.

Corollary 7. *The singular part of the Puiseux series of* $\mathfrak{z}_i$*, for* $1 \leq i \leq i$*, remains unchanged if the polynomial* $f(x, y)$ *is truncated at total-degree* $\mu_q + 1$.

PROOF. As can easily be seen, terms of total-degree greater than $\mu_q + 1$ in the original polynomial do not contribute to terms of x-degree less than $\mu_q + 1$ in the transformed polynomials. Hence, the result follows from Corollary 5. $\square$

Using the result above we can reformulate Lemma 27 as: The number of k-operations can be bounded by $\mathbf{O}(\mu_q^3)$. But since μ_q may be of order $\mathbf{O}(N^2)$ and the method of passive expansion (described in Subsection 4.5) relays on the fact that *all* terms up to some given x-degree are known, we do not use this result.

4.3. Computing with Algebraic Numbers

As we saw in the previous section the coefficients of Newton transformations and hence the coefficients of Puiseux parametrizations are in general algebraic numbers (defined as the roots of a univariate polynomial, see page 52). Therefore, when computing the singular part of a curve-branch we have to deal with multiple algebraic extensions of the ground field.

There are at least two different ways to handle these multiple algebraic extensions: Either by repeatedly computing primitive elements or by introducing a new variable for each algebraic extension.

4.3.1. *Primitive Elements.* Let α be a root of $f_\alpha(x) \in k'[x]$ and let β be a root of $f_\beta(x) \in k'(\alpha)[x]$. We want to compute a minimal polynomial $f_\gamma(x) \in k'[x]$ of a primitive element γ of α and β w.r.t. k', i.e. $k'(\gamma) = k'(\alpha, \beta)$, together with representations of α and β in terms of γ, i.e. $\alpha = g_\alpha(\gamma)$, $\beta = g_\beta(\gamma)$, and $g_\alpha(x), g_\beta(x) \in k'[x]$.

An algorithm for computing $f_\gamma(x)$, $g_\alpha(x)$, and $g_\beta(x)$ can be found in [**van der Waerden, 1991**, Section 6.10] or [**Winkler, 1996**, Section 5.4]. One step in the algorithm is to compute a gcd in $k'(\gamma)$. It is known that this gcd-computation can be done in d^3 k'-operations, where d is the degree of the algebraic extension of γ. As far as we know, this is also a lower bound for all currently known algorithms for computing primitive elements.

As we will see below, the use of multiple variables allows us to replace one operation in $k'(\alpha, \beta)$ by d^2 operations in k' and therefore we will no longer persue the method of primitive elements.

Another reason why we will not use this method is the size of the coefficients of the primitive polynomials. Although we do not consider bit-complexity in our analysis, we just want to give one example of how the coeffient size may increase.

Example 11. Let α_1, α_2, and α_3 be roots of $f_1(x) = x^3 + 7x^2 + 9$, $f_2(x) = x^4 + 9x^3 + 5$, and $f_3(x) = x^2 + 7x + 3$. The polynomials $f_1(x)$, $f_2(x)$, and $f_3(x)$ are absolute irreducible w.r.t. to $\mathbb{Q}[x]$, $\mathbb{Q}(\alpha_1)[x]$, and $\mathbb{Q}(\alpha_1, \alpha_2)[x]$.

A primitive element $\alpha_{1.2}$ of α_1 and α_2 is defined e.g. by $f_{1.2}(x) = 1366725 - 1454669x + 773826x^2 + 1022803x^3 + 2122021x^4 + 1059700x^5 + 676122x^6 + 343090x^7 + 94621x^8 + 14422x^9 + 1230x^{10} + 55x^{11} + x^{12}$, which is not that bad. But a representation of α_1 in terms of $\alpha_{1.2}$ already contains fractions with numbers of the order 10^{27}.

Things really get worse if a primitive element $\alpha_{1.2.3}$ of α_1, α_2, and α_3 is computed. The coefficients of a minimial polynomial of $\alpha_{1.2.3}$ may be of order 10^{19}, whereas the ones of a representation of α_1 in terms of $\alpha_{1.2.3}$ may be of order 10^{91}.

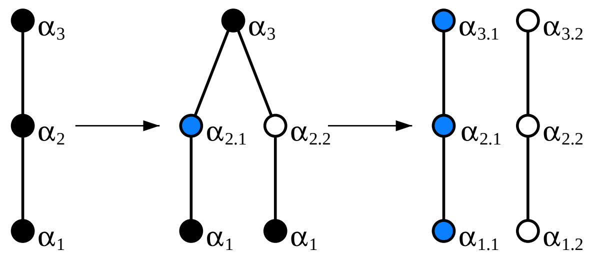


FIGURE 4. Case splitting during passive factorization

4.3.2. *Multiple Variables.* Another possibility to compute in the field $k'(\alpha_1, \ldots, \alpha_n)$ is to introduce for each α_i, for $1 \le i \le n$, a new variable and a minimal polynomial $p_i(x) \in k'(\alpha_1, \ldots, \alpha_{i-1})[x]$. Since $k'(\alpha_1, \ldots, \alpha_i)$ is isomorphic to $k'(\alpha_1, \ldots, \alpha_{i-1})[x]/p_i(x)$, computations in $k'(\alpha_1, \ldots, \alpha_n)$ can effectively be done. Let d be the degree of the algebraic extension, i.e. $d = [k'(\alpha_1, \ldots, \alpha_n) : k']$. Each operation in $k'(\alpha_1, \ldots, \alpha_n)$ can then be replaced by at most d^2 operations in k'.

As explained by D. Duval the above idea can be extended to the case where the polynomials p_i are no longer necessarily prime but just square-free. We therefore do not compute in a field, but in a ring with zero divisors. Since square-freeness can be checked by gcd-computation, polynomial factorization can be avoided at all. For details about this method we refer to **[Dicrescenzo and Duval, 1984]**, **[Dicrescenzo and Duval, 1985]**, or **[Duval, 1994]**.

What we additionally have to take into account from now on are zero-equality tests. The case that a term $a(x)$ contains the zero element, can again be decided by gcd-computation. If the gcd is non-trivial, we have to consider two different cases. At the same moment we also found a non-trivial factor of some $p_i(x)$. For this reason, this method is also called passive factorization.

We assume that at every step in our algorithm all minimal polynomials $p_i(x)$, $1 \le i \le n$, are monic and fully reduced w.r.t. $p_j(x)$, for $1 \le j < i$.

A typical case splitting situation is shown in Figure 4. If a non-trivial factorization of $p_i(x)$ is encountered, say $p_i(x) = p_{i.1}(x)p_{i.2}(x)$, all the polynomials $p_j(x)$, for $i < j \le n$, get reduced w.r.t. $p_{i.1}(x)$ and $p_{i.2}(x)$ yielding new minimal polynomials $p_{j.1}(x)$ and $p_{j.2}(x)$. Note, that during this reduction no new case splitting can occur, since we assumed that the $p_j(x)$, $i < j \le n$, are already monic and reduced w.r.t. $p_i(x)$. The polynomials $p_j(x)$, for $1 \le j < i$, are not affected by this splitting and therefore we have $p_{j.1}(x) = p_{j.2}(x) = p_j(x)$, for $1 \le j < i$.

Example 12. We define three algebraic numbers a, b, and c by $f_a(x) = x^2 - 4 \in \mathbb{Q}[x]$, $f_b(a, x) = x^2 - 1 \in \mathbb{Q}(a)[x]$, and $f_c(a, b, x) = x^2 - 7bx + 12 \in \mathbb{Q}(a, b)[x]$.

To check if the expression $(a+b-1)c-6$ is zero, we first compute

$$\begin{aligned} g_c(a,b,x) &= \gcd((a+b-1)x-6, x^2-7bx+12) \\ &= -5b^2-(3a+3)b-4a+2a^2+8. \end{aligned}$$

Now, to decide if $g_c(a,b,x)$ is zero, we compute a second gcd, namely

$$\begin{aligned} g_b(a,x) &= \gcd(-5x^2-(3a+3)x-4a+2a^2+8, x^2-1) = \\ &= 4a^4-16a^3+19a^2-6a. \end{aligned}$$

In a third step, since the gcd of $g_b(a,x)$ and $f_a(x)$ is non trivial,

$$g_a(x) = \gcd(4x^4-16x^3+19x^2-6x, x^2-4) = x-2,$$

we get a factorization of $f_a(x) = (x-2)(x+2)$. This factorization (for the case $a-2=0$) simplifies $g_c(a,b,x)$ to $g'_c(a,b,x) = -5b^2-3b+8$. We know that the gcd of $g'_c(a,b,x)$ and $f_b(a,b)$ has to be non trivial and it is equal to

$$\gcd(-5x^2-3x+8, x^2-1) = x-1.$$

Hence we can write $f_b(x) = (x-1)(x+1)$. This factorization (for the case $b-1=0$) makes the gcd of $(a+b-1)c-6$ and $f_c(a,b,x)$ non trivial, namely

$$\gcd((a+b-1)c-6, x^2-7bx+12) = x-3$$

and factors $f_c(a,b,x)$ into $(x-3)(x-4)$.

So the equality test has caused a splitting into eight cases

	1	2	3	4	5	6	7	8
c	$x-3$	$x-3$	$x+3$	$x+3$	$x-4$	$x-4$	$x+4$	$x+4$
b	$x-1$	$x-1$	$x+1$	$x+1$	$x-1$	$x-1$	$x+1$	$x+1$
a	$x-2$	$x+2$	$x-2$	$x+2$	$x-2$	$x+2$	$x-2$	$x+2$

where $(a+b-1)c-6$ is zero in the case 1.

4.4. Number of Operations in the Ground Field k'

If we handle algebraic numbers the way described in Subsection 4.3.2 we also have to take into account equality tests between two numbers. The additional advantage of the passive factorization method is, that we can completely neglect root computation, i.e. factorization, which is the reason we never considered it.

The only place, where equality tests are needed is within the procedure `newton-polygon`. Within this procedure we have to test if coefficients become zero or not. Hence at the k-th step there are at most $(M_{i,k}+1)(N+1)$ tests and therefore the total number of equality tests can again be bounded by $\mu_q(\mu_q+2)(N+1)$ (cf. Lemma 26).

Assuming that the degree of the algebraic extension necessary to compute the Puiseux series of $\mathfrak{z}_i$ is d_i we know from the previous subsection that each operation in k can be replaced by d_i^2 operations in k'. As shown in **[Duval, 1989]** $\sum_{i=1}^{g} d_i \leq N$. Thus a bound for the total number of operations in k' would be $\mu_q(\mu_q+2)(N+1)N^2$.

As we will show below, this bound can be improved by a factor of N^2, which means, that the additional complexity introduced by the algebraic extensions does not contribute at all to the total complexity of resolving one singularity.

Lemma 28. *The number K_i, for $1 \leq i \leq I$, of Newton transformations for computing the singular part of $\mathfrak{z}_i$ is less than or equal to $\frac{\mu_q}{d_i}$.*

PROOF. Let $\mathfrak{z}$ be one of the curve-branches $\mathfrak{z}_i$, $1 \leq i \leq I$, and let d be the degree of the algebraic extension necessary to compute its Puiseux series. Since we compute d conjugated curve-branches at once, with each Newton transformation at least d neighboring points of multiplicity greater or equal to 2 are computed. The number of these points is at most μ_q and therefore the number of steps in the Newton–Puiseux algorithm is at most $\frac{\mu_q}{d}$. □

Theorem 9. *At most $3\mu_q^2(N+1)$ operations in k' are needed to compute the singular parts of all Puiseux expansions with origin q.*

PROOF. The proof is basically the same as for Lemma 27. Using the fact, that the sum of multiplicities of conjugated curve-branches bounding the sum of x-degrees of the transformed polynomials is non-overlapping, the numbers $M_{i,k}$ have to be less than or equal to $\frac{\mu_q}{d_i}$. Together with Lemma 28 we get for the total number C_i of operations w.r.t. k' necessary to resolve one curve-branch $\mathfrak{z}_i$ the inequality

$$C_i \leq \frac{\mu_q}{d_i}\left(\frac{\mu_q}{d_i}+2\right)(N+1)\, d_i^2 = \mu_q(\mu_q+2d_i)(N+1).$$

This yields a bound for the total number C of operations w.r.t. k' necessary to resolve all curve-branches with origin q, which is given by

$$C \leq 3\mu_q^2(N+1).$$

□

We close this subsection stating two variations of Theorem 9 without proofs:

- The multiplicities used in the complexity analysis are fully determined by the numbers $p_{i,k}$, $q_{i,k}$ of the Newton transformations (see Lemma 18 on 58). Therefore it would be possible to formulate the complexity bound of Theorem 9 in terms of characteristic pairs.
- The multiplicities used in the complexity analysis are exactly those which result from the first step in the Euclidean division chain (Theorem of Enriques–Chisini). This means that the complexity bound of Theorem 9 could also be formulated using the highest exponent occurring in the singular part of the Puiseux parametrization.

4.5. Passive Expansion

In the previous subsections we have shown that a singularity q of $\mathcal{C}$ can be resolved using no more than $\mathbf{O}(\mu_q^2 N)$ operations in the ground field k'. When we implement the Newton–Puiseux algorithm in a straight forward way it would be necessary to explicitly compute μ_q first in order to be able make practical use of this complexity bound.

An algorithm for computing μ_q can be found e.g. in the article of Teitelbaum, see [**Teitelbaum, 1990**]. This algorithm (based on standard basis computation) requires $\mathbf{O}(\mu_q^4)$ operations in k'. So it may be the case that computing μ_q may require more operations than computing the whole resolution of $\mathcal{C}$ at q (again provided we know μ_q in advance).

In the following we show how we can circumvent this situation using passive expansion. The idea of passive expansion is somehow related to (although in detail quite different from) that of lazy-evaluation as known from power-series computations. Passive expansion strongly depends on the special property of Newton transformations, namely that they do not increase the y-degree of the transformed polynomials.

During the computation of a Puiseux parametrization we do not fully expand a transformed polynomials $f_{i,k}(x,y)$, but only those parts of $f_{i,k}(x,y)$ which are needed to compute the Newton polygon of $f_{i,k}(x,y)$. So the polynomials $f_{i,k}(x,y)$ contain at any time explicitly all their coefficients, but for terms of a certain high enough x-degree we allow a collected and nested representation.

As we will see later-on, this allows us to make practical use of the theoretical x-degree bounds derived in Subsection 4.2. And since passive expansion does not introduce any additional complexity, the number of operations in k' to resolve $\mathcal{C}$ at q is $\mathbf{O}(\mu_q^2 N)$ also for the case, that we do not know μ_q in advance.

To make things precise we define the following.

Definition 64. Let $g(x,y) \in k[x,y]$ be a polynomial. We say that $g(x,y)$ is x-*expanded up to order* m if $g(x,y)$ is of the form

$$g(x,y) = \sum_{i=0}^{\Delta} x^{\delta} g_{\delta}(x,y),$$

where $g_{\delta}(x,y) \in k[y]$, for $0 \le i \le m$, and $g_i(x,y) \in k[x,y]$, for $m < i \le \Delta$.

Recalling the definition of a Newton transformation it is easily seen, that a polynomial which is x-expanded up to a high enough order is, after applying the transformation to it, again x-expanded up to some order. Figure 5 and Figure 6 illustrate the forward and backward dependencies of the coefficients of a polynomial w.r.t. Newton transformations.

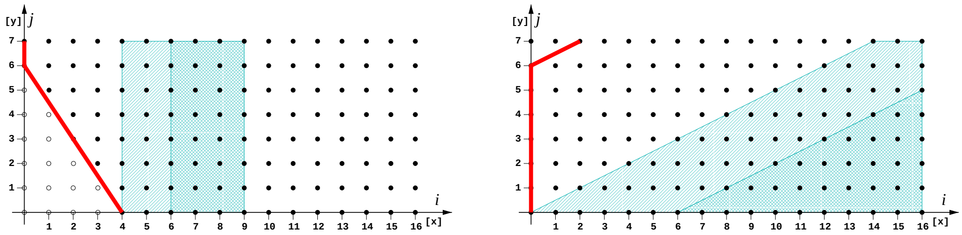


FIGURE 5. Forward coefficient dependency

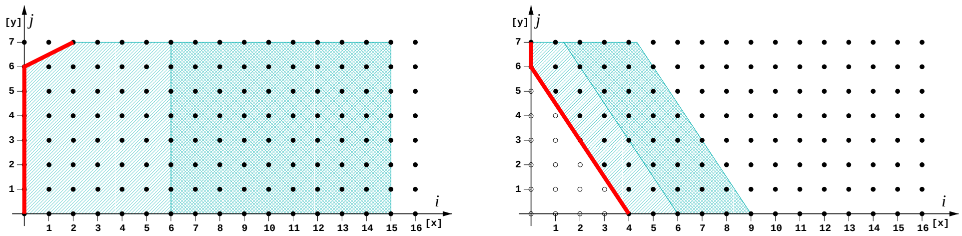


FIGURE 6. Backward coefficient dependency

Lemma 29. *Let* $g(x,y) = \sum_{\delta=0}^{\Delta} x^\delta g_\delta(x,y) \in k[x,y]$ *be a polynomial and let* $\tau = (p,q,r;\ \xi,u,v)$ *be a Newton transformation derived from the Newton-polygon of* $g(x,y)$*. If* $g(x,y)$ *is* x*-expanded up to order* $m+r$*, then* $\tau(g(x,y))$ *is* x*-expanded up to order* qm *just by substitution.*

PROOF. Let $g_\delta(x,y) = \sum_{j=0}^{N} c_{\delta,j} y^j$, for $0 \le i \le m+r$. It holds then that

$$
\begin{aligned}
\tau(g(x,y)) = & \\
& \sum_{\delta=0}^{\Delta} \tau(x^\delta g_\delta(x,y)) = \\
& \sum_{\delta=0}^{\Delta} \xi^{-v\delta} x^{q\delta - r} g_\delta(\xi^{-v} x^q, x^p(\xi^u + y)) = \\
& \sum_{\delta=0}^{m+r} \sum_{j=0}^{N} \xi^{-v\delta} c_{\delta,j} x^{q\delta+pj-r} (\xi^u + y)^j + \\
& \sum_{\delta=m+r+1}^{\Delta} \xi^{-v\delta} x^{q\delta - r} g_\delta(\xi^{-v} x^q, x^p(\xi^u + y)).
\end{aligned}
$$

As we can see, all x-degrees in the last sum are greater than qm and hence no expansion is necessary to x-expand $\tau(g(x,y))$ up to order k. $\square$

The next Lemma shows that it makes no difference if we compute the coefficients of the polynomials $f_{i,k}(x,y)$ (see Definition 59 on page 55) at each step up to some given x-degree (i.e. full expansion up to some x-degree and truncating all higher x-degree terms) or if we defer the expansion.

This means, passive expansion causes no additional computational work and makes it superfluous to know μ_q in advance, since the required order of the x-expansions is determined by the Newton-polygons of the $f_{i,k}(x,y)$'s themselves.

A Newton transformation τ can be decomposed into two parts $\tau = \tau_2 \circ \tau_1$, where

$$\tau_1 \colon (x,y) \mapsto (x^q, x^p y)$$
$$\tau_2 \colon (x,y) \mapsto (x, \zeta + y).$$

In this decomposition the transformation τ_1 involves no expansion at all and already yields the correct x-degrees.

Lemma 30. *Let*

$$f_{i,k}(x,y) = \sum_{\delta=0}^{\Delta_{i,k}} x^\delta f_{i,k,\delta}(x,y),$$

where $f_{i,k,\delta}(x,y) \in k[x,y]$, *for* $0 \le \delta \le \Delta_{i,k}$. *It needs no more than*

$$\left(\sum_{k'=0}^{k-1} (k'+1) l_{i,k'} + k\delta_0 \right) N$$

multiplications to expand all $f_{i,k,\delta}(x,y)$, *for* $\delta \le \delta_0$.

Proof. We prove this lemma by induction on k.

- $k = 0$:
  Since we assume that the input polynomial is given in expanded form nothing has to be shown.
- $k \to k+1$:
  To expand all $f_{i,k+1,\delta}(x,y)$, for $\delta \le \delta_0$ we have first to expand all $f_{i,k,\delta}(x,y)$, for $\delta \le l_{i,k} + \frac{\delta_0}{q_{i,k}}$. By the induction hypothesis this can be done using no more than

  $$\left(\sum_{k'=0}^{k-1} (k'+1) l_{i,k'} + k(l_{i,k} + \frac{\delta_0}{q_{i,k}}) \right) N$$

  multiplications. In addition to that $(l_{i,k} + \frac{\delta_0}{q_{i,k}})N$ multiplications are needed to expand all $f_{i,k+1,\delta}(x,y)$, for $\delta \le \delta_0$, once the $f_{i,k,\delta}(x,y)$, for $\delta \le l_{i,k} + \frac{\delta_0}{q_{i,k}}$, are already expanded. So we get in total

$$\left(\sum_{k'=0}^{k-1} (k'+1) l_{i,k'} + k(l_{i,k} + \frac{\delta_0}{q_{i,k}}) + l_{i,k} + \frac{\delta_0}{q_{i,k}} \right) N =$$

$$\left(\sum_{k'=0}^{k} (k'+1) l_{i,k'} + (k+1)\frac{\delta_0}{q_{i,k}} \right) N \le \left(\sum_{k'=0}^{k} (k'+1) l_{i,k'} + (k+1)\delta_0 \right) N.$$

$\square$

4.6. Other Resolution Algorithms

Other complexity bounds for the number of ground field operations necessary to resolve a singular point q with Milnor-number μ_q can be found in:

- [**Teitelbaum, 1990**]: As already said, in the article of Teitelbaum first μ_q is computed using standard bases. Once μ_q is known it can be used to find degree bounds for the intermediate results. The resolution sequences are computed using quadratic transformations and algebraic numbers are handled using primitive elements together with passive factorization. The overall complexity of this approach is $\mathbf{O}(\mu_q^6)$.
- [**Henry and Merle, 1987**]: In this article the authors use no a-priori degree bound. To avoid unnecessary computations with high degree terms lazy evaluation is used for all computations instead (see also [**Moreno Socías, 1990**]). Again, singularities are resolved using quadratic transformations. The problem of dealing with algebraic numbers is not addressed, since the analysis is restricted to irreducible singularities where algebraic extensions can not occur. A complexity bound for this algorithm is $\mathbf{O}(\mu_q^4)$.

We think our complexity bound of $\mathbf{O}(\mu_q^2 N)$ was made possible for the following reasons:

- Our algorithm is based on Newton transformations instead of quadratic transformations. This allows us to have a very strict control of the x-degrees of the intermediate polynomials. Since the y-degrees do not increase at all, we can find a rather low bound for the total number of coefficients of the intermediate polynomials occurring in the algorithm.
- The method of multiple variables, combined with passive factorization, seems to be computationally superior to the primitive element method.
- The additional complexity introduced by algebraic numbers occurring in the Newton–Puiseux algorithm is fully covered by the fact, that conjugated curve-branches and singularities can be treated together. Most probably this also should hold for the algorithm described in [**Henry and Merle, 1987**].
- Using deferred expansion, it is possible to make practical use of the theoretical degree bounds. It also circumvents the problem of having to know some a-priori complexity measure of the singularity, e.g. its Milnor-number, in order to simplify intermediate results without changing its resolution.

CHAPTER 5

Applications

In this chapter we show how the information we get from the Newton–Puiseux algorithm, namely sequences of Newton transformations respectively truncated Puiseux parametrizations for the curve-branches, can be used to answer some questions arising in the field of algebraic geometry, e.g. Milnor-number and genus computation, computing the system of adjoint curves, or computing parametrizations of rational curves.

As we will see, once the singular parts of all curve-branches of a given curve are known, these questions can be answered rather easily (also with respect to the computational complexity). The correctness of the methods described below is in most cases a more or less direct consequence of the theoretical results stated in Chapter 3.

For some of the algorithms we also mention the computational complexity. The analysis is not done in such detail as we did it for the Newton–Puiseux algorithm in Chapter 4, we rather point out which sub-algorithms additionally to the Newton–Puiseux algorithm are used. Examples of such sub-algorithms are: solving linear systems of equations, resultant- or gcd-computations. For a detailed complexity analysis of these algorithms we refer to [**Winkler, 1996**] and [**von zur Gathen and Gerhard, 1999**].

For experimental results the algorithms were implemented in Maple V (release 5). The computation times given in this chapter were measured under Linux (kernel 2.2) running on an Intel Pentium-II architecture with a CPU-speed of 350MHz and 128MB main memory.

In the following let $\mathcal{C}$ be an irreducible curve defined by $f(x, y) \in k'[x, y]$ of total degree D.

1. Resolution

We assume that $q = (0, 0)$ is a singular point of $\mathcal{C}$. Let $\mathfrak{z}_i$, for $1 \leq i \leq I$, be the curve-branches of $\mathcal{C}$ centered at q. Solutions for the following problems can easily be computed once the singular parts of the curve-branches $\mathfrak{z}_i$ are known:

- Compute the coordinates of the neighboring points of q.
- Compute the intersection semigroup for each curve-branch $\mathfrak{z}_i$.
- Compute the conductor number for each curve-branch $\mathfrak{z}_i$.
- Compute the Milnor-number of $\mathcal{C}$ at q.

1.1. Coordinates of Neighboring Points

We fix one of the curve-branches $\mathfrak{z}_i$ and denote it by $\mathfrak{z}$. The singular part of its Puiseux parametrization shall be given by $(A(t), B(t))$ and its proper multiplicity sequence by $(m_h)_{0\leq h\leq H}$.

As we explained thoroughly in Section 3 of Chapter 3 the coordinates of the neighboring points of q w.r.t. $\mathfrak{z}$ can be obtained by computing the sequence

$$
\begin{aligned}
(A_0(t), B_0(t)) &= (A(t), B(t)),\\
(A_{h+1}(t), B_{h+1}(t)) &= \begin{cases} (A'_h(t), B'_h(t)/A'_h(t)) & \text{if } a_h \leq b_h,\\ (A'_h(t)/B'_h(t), B'_h(t)) & \text{if } a_h > b_h, \end{cases}
\end{aligned}
$$

where

$$
\begin{aligned}
(A'_h(t), B'_h(t)) &= (A_h(t) - A_h(0), B_h(t) - B_h(0)),\\
(a_h, b_h) &= (\operatorname{ord}(A'_h(t)), \operatorname{ord}(B'_h(t))).
\end{aligned}
$$

We also define the numbers c_h and d_h as

$$
\begin{aligned}
(c_0, d_0) &= (\deg(B_0(t)), \deg(A_0(t))),\\
(c_{h+1}, d_{h+1}) &= \begin{cases} (c_h, d_h - m_h) & \text{if } a_h \leq b_h,\\ (c_h - m_h, d_h) & \text{if } a_h \leq b_h. \end{cases}
\end{aligned}
$$

From Corollary 3 it follows that already the singular part of the Puiseux parametrization determines the corodinates of all proper neighboring points of q and that the power series expansions for the polynomial quotients $A'_h(t)/B'_h(t)$ and $B'_h(t)/A'_h(t)$ can be truncated at $e_h = \max(c_h, d_h)$.

The coordinates of the neighboring points q_h of q are hence given by $B_h(0)$ or $A_h(0)$, depending whether $a_{h-1} \leq b_{h-1}$ or not. A detailed algorithmic description of the above procedure can be found in Appendix A Section 2.

1.2. Intersection Semigroup

Let $(k_g)_{0\leq g\leq G}$ be the k-sequence of a curve-branch $\mathfrak{z}$. Following Abhyankar in [**Abhyankar, 1990**, Section 21] we define the q-, s-, and r-sequences

$$
d_g = \gcd(k_0, \dots, k_{g-1}), \text{ for } 1 \leq g \leq G,
$$

$$
\begin{aligned}
q_0 &= k_0,\\
q_1 &= k_1,\\
q_g &= k_g - k_{g-1}, \text{ for } 2 \leq g \leq G,
\end{aligned}
$$

$$\begin{aligned} s_0 &= q_0, \\ s_g &= \sum_{g'=1}^{g} d_{g'} q_{g'}, \text{ for } 1 \le g \le G, \end{aligned}$$

and

$$\begin{aligned} r_0 &= s_0, \\ r_g &= s_g / d_g, \text{ for } 1 \le g \le G. \end{aligned}$$

It now holds that the intersection semigroup of $\mathfrak{z}$ is generated by the numbers $r_0, \ldots, r_G$. For more details we refer to [**Abhyankar, 1967**] and [**Abhyankar, 1977**].

1.3. Conductor number

The conductor number c_i of a curve-branch $\mathfrak{z}_i$ is equal to

$$\begin{aligned} c_i = \sum_{0 \le h \le H_i} m_{i,h} \left(\sum_{i' \sim_h i} m_{i',h} - 1 \right) = \\ \sum_{0 \le h \le H_i} \left(m_{i,h}(m_{i,h} - 1) + \sum_{\substack{i' \sim_h i, \\ i' \ne i}} m_{i,h} m_{i',h} \right) = \\ \sum_{0 \le h \le H_i} m_{i,h} m_{i,h} - \sum_{0 \le h \le H_i} m_{i,h} + \sum_{\substack{i' \sim_h i, \\ i' \ne i}} \mathbf{mult}_q(\mathfrak{z}_i, \mathfrak{z}_{i'}). \end{aligned}$$

In Chapter 4 Section 2 we showed how these sums can easily be computed from the Newton transformations of $\mathfrak{z}_i$ (see Lemma 17 and Lemma 18 on page 17). Therefore the conductor number can be computed along with the singular part of the Puiseux parametrization of $\mathfrak{z}_i$. This modification of the Newton–Puiseux algorithm does of course not change the order of its computational complexity.

1.4. Milnor number

Once we have computed the conductor numbers c_i for all curve-branches the Milnor number μ_q of $\mathcal{C}$ at q is also known, since

$$\mu_q = \sum_{i=1}^{I} (c_i - 1) + 1.$$

Lemma 31. *Let* $q = (0,0)$. *The Milnor-number* μ_q *of* $\mathcal{C}$ *at* q *can be computed in* $\mathbf{O}(\mu^2 N)$ k'*-operations.*

PROOF. Follows from the complexity bound given in Theorem 9. □

1.5. Experimental Results

In the following, three sets of examples are given, where the second one consists of a family of singularities with an increasing number of characteristic pairs, but not involving any algebraic extensions.

The procedure "conPa" is called once at the beginning of the computation and converts a polynomial into a form suitable for passive expansion. In our current implementation we do not use passive factorization as described in Subsection 4.3 of Chapter 4 to handle algebraic numbers. Instead, the procedure "solAE" performs a full factorization of the polynomial $\Phi(z)$ which defines the Newton coefficients (see page 52). Although this factorization increases the theoretical complexity, it has no significant impact on practical computation times.

The procedure "appSu" is a sub-procedure of "appTr" and involves no polynomial arithmetic but just applies a substitution without expanding the resulting polynomial. Therefore, using a sophisticated implementation the time necessary for "appSu" should become almost zero. Looking at the examples we see that in our implementation this is not always the case (in example $p_6(x, y)$ most of the time is actually spent in the sub-procedure "appSu"). Reasons for this could be some unwanted expansions or evaluations caused by Maple's subs-command.

Computation times are shown in Table 1. In Table 2 the degree of the defining polynomials, the conductor numbers, and the singular part of the Puiseux parametrizations are given.

Our implementation uses a refinement in the Newton–Puiseux algorithm for getting small rational Newton coefficients (as described on page 106). Omitting this improvement coefficients would become quite large, e.g. for $p_6(x, y)$ we would get

$$\begin{aligned}
&(-18446744073709551616t^{64}, 79228162514264337593543950336t^{96} - \\
&2596148429267413814265248164610048t^{112} + \\
&332306998946228968225951765070086144t^{120} - \\
&2658455991569831745807614120560689152t^{124} + \\
&5316911983139663491615228241121378304t^{126} + \\
&5316911983139663491615228241121378304t^{127}).
\end{aligned}$$

	conPa	solAE	appSu	appTr	comNP	comCN	total
$f_1(x,y)$	0.000	0.000	0.000	0.000	0.000	0.000	0.091
$f_2(x,y)$	0.000	0.000	0.000	0.000	0.000	0.000	0.131
$f_3(x,y)$	0.000	0.000	0.000	0.010	0.010	0.000	0.070
$f_4(x,y)$	0.011	0.010	0.000	0.009	0.000	0.000	0.070
$f_5(x,y)$	0.011	0.000	0.000	0.010	0.000	0.011	0.121
$f_6(x,y)$	0.021	0.000	0.019	0.049	0.019	0.010	0.370
$f_7(x,y)$	0.191	0.058	0.030	0.481	0.010	0.070	0.981
$p_1(x,y)$	0.000	0.000	0.000	0.000	0.000	0.000	0.030
$p_2(x,y)$	0.000	0.010	0.000	0.000	0.000	0.000	0.119
$p_3(x,y)$	0.000	0.010	0.010	0.020	0.010	0.000	0.149
$p_4(x,y)$	0.020	0.000	0.020	0.121	0.010	0.010	0.310
$p_5(x,y)$	0.040	0.000	0.158	0.280	0.071	0.000	0.660
$p_6(x,y)$	0.250	0.021	25.979	27.339	0.039	0.000	27.960
$q_1(x,y)$	0.140	0.009	0.511	0.908	0.011	0.000	1.329
$q_2(x,y)$	0.189	0.000	1.023	2.343	0.030	0.000	2.829
$a_1(x,y)$	0.011	0.028	0.000	0.010	0.000	0.009	0.240
$a_2(x,y)$	0.010	0.041	0.000	0.049	0.000	0.000	0.171
$a_3(x,y)$	0.030	0.030	0.020	0.131	0.000	0.010	0.361

Abbreviation: conPa ... convert to passive expansion form, solAE ... factor univariate polynomials over algebraic number field, appSu ... apply Newton substitutions, appTr ... apply Newton transformations, comNP ... compute Newton polygons, comCN ... compute conductor numbers.

TABLE 1. Computation times

	deg.	cond.	Puiseux parametrizations
$f_1(x,y)$	10	34	$(3t^7, 3t^5)$
		12	$(t^2, 3t^3)$
$f_2(x,y)$	10	18	(t^4, t^6+t^9)
$f_3(x,y)$	10	18	(t^4, t^6+t^9)
$f_4(x,y)$	20	48	$(\frac{t^4}{\alpha^2}, \frac{t^6}{\alpha^3}+\frac{t^9}{\alpha^4}),\ \alpha^2-2\alpha+2=0$
$f_5(x,y)$	20	48	$(\frac{t^4}{\alpha^2}, \frac{t^6}{\alpha^3}+\frac{t^9}{\alpha^4}),\ \alpha^2-2\alpha+2=0$
$f_6(x,y)$	30	74	$(3t^7, 3t^5)$
		36	$(t^2, 3t^3)$
		81	$(t^4, t^6+t^9+t^{10})$
		81	$(t^4, t^6+t^9+2t^{10})$
$f_7(x,y)$	60	169	$(\frac{t^4}{\alpha^2}, \frac{t^6}{\alpha^3}+\frac{t^9}{\alpha^4}+\frac{(1-\alpha)t^{10}}{\alpha^3}),\ \alpha^2-2\alpha+2=0$
		169	$(\frac{t^4}{\alpha^2}, \frac{t^6}{\alpha^3}+\frac{t^9}{\alpha^4}+\frac{(1-\alpha)t^{10}}{2\alpha^3}),\ \alpha^2-2\alpha+2=0$
		169	$(t^4, t^6+t^9+t^{10})$
		169	$(t^4, t^6+t^9+2t^{10})$
$p_1(x,y)$	3	2	$(-t^2, -t^3)$
$p_2(x,y)$	6	16	$(-t^4, t^6-\frac{1}{2}t^7)$
$p_3(x,y)$	13	84	$(-t^8, t^{12}-\frac{1}{2}t^{14}+\frac{1}{4}t^{15})$
$p_4(x,y)$	26	380	$(-t^{16}, t^{24}-\frac{1}{2}t^{28}+\frac{1}{4}t^{30}+\frac{1}{8}t^{31})$
$p_5(x,y)$	53	1612	$(-t^{32}, t^{48}-\frac{1}{2}t^{56}+\frac{1}{4}t^{60}-\frac{1}{8}t^{62}-\frac{1}{16}t^{63})$
$p_6(x,y)$	106	6636	$(-t^{64}, t^{96}-\frac{1}{2}t^{112}+\frac{1}{4}t^{120}-\frac{1}{8}t^{124}+\frac{1}{16}t^{126}-\frac{1}{32}t^{127})$
$q_1(x,y)$	66	2038	$(-t^{32}, t^{48}-\frac{1}{2}t^{56}+\frac{1}{4}t^{60}-\frac{1}{8}t^{62}-\frac{1}{16}t^{63})$
		510	$(-t^8, t^{12}-\frac{1}{2}t^{14}+\frac{1}{4}t^{15}+\frac{11}{128}t^{17})$
$q_2(x,y)$	79	2465	$(-t^{32}, t^{48}-\frac{1}{2}t^{56}+\frac{1}{4}t^{60}-\frac{1}{8}t^{62}-\frac{1}{16}t^{63})$
		1233	$(-t^{16}, t^{24}-\frac{1}{2}t^{28}+\frac{1}{4}t^{30}+\frac{1}{8}t^{31}+\frac{11}{256}t^{33})$
$a_1(x,y)$	14	40	$(t^4, 2t^6+t^7)$
		40	(t^4, t^6+t^7)
$a_2(x,y)$	25	113	$(\frac{t^5}{\alpha}, \frac{t^6}{\alpha}+\beta t^7),\ \alpha^2+1=0,\ \beta^2+\alpha=0$
$a_3(x,y)$	36	446	$(t^{12}, t^{18}+\frac{1}{2}t^{21}+\frac{1}{4}t^{23})$
		450	$(t^{12}, t^{18}+\frac{1}{2}t^{21}+\frac{1}{4}t^{24}+\frac{1}{4}t^{25})$

TABLE 2. Results

$$
\begin{aligned}
f_1(x,y) &= (9x^5 - y^7)(9x^3 - y^2),\\
f_2(x,y) &= y^4 - 4x^6y - 4x^7y - 2x^3y^2 - 4x^4y^2 - 2y^2x^5 + 3x^9 + x^6 +\\
&\quad 4x^7 + 6x^8 + x^{10},\\
f_3(x,y) &= y^4 - 4x^6y - 8x^7y - 2x^3y^2 - 8x^4y^2 - 8y^2x^5 + 31x^9 + x^6 +\\
&\quad 8x^7 + 24x^8 + 16x^{10},\\
f_4(x,y) &= 25x^{18} + y^8 + 52x^{15} + x^{12} + 36y^4x^8 + 24y^4x^7 - 4y^6x^5 + 6y^4x^{10} +\\
&\quad 20x^{13}y^2 - 60x^{11}y^2 - 24x^{10}y^2 + 16x^{12}y^3 + 16y^3x^9 - 8x^{17}y +\\
&\quad 4x^{19} - 8y^5x^6 + 8x^{13} + 28x^{14} - 40y^2x^{12} + 54x^{16} + 32x^{17} - 8x^7y^5 -\\
&\quad 8y^6x^4 - 8yx^{12} + 16x^{14}y^2 + x^{20} - 80yx^{14} - 4x^{15}y^2 + 48y^3x^{11} -\\
&\quad 40yx^{13} - 40yx^{15} - 4x^3y^6 + 6x^6y^4 - 4x^9y^2 + 20y^4x^9 + 48y^3x^{10},\\
f_5(x,y) &= 1677x^{18} + y^8 + 444x^{15} + x^{12} + 144y^4x^8 + 48y^4x^7 - 16y^6x^5 +\\
&\quad 96y^4x^{10} - 800x^{13}y^2 - 240x^{11}y^2 - 48x^{10}y^2 + 128x^{12}y^3 + 16y^3x^9 -\\
&\quad 256x^{17}y + 960x^{19} - 8y^5x^6 + 16x^{13} + 112x^{14} - 600y^2x^{12} + 1088x^{16} +\\
&\quad 1696x^{17} - 16x^7y^5 - 16y^6x^4 - 8yx^{12} - 608x^{14}y^2 - 560yx^{16} +\\
&\quad 256x^{20} - 320yx^{14} - 256x^{15}y^2 + 192y^3x^{11} - 80yx^{13} - 600yx^{15} -\\
&\quad 4x^3y^6 + 6x^6y^4 - 4x^9y^2 + 188y^4x^9 + 96y^3x^{10},\\
f_6(x,y) &= f_1(x,y)f_2(x,y)f_3(x,y),\\
f_7(x,y) &= f_2(x,y)f_3(x,y)f_4(x,y)f_5(x,y).
\end{aligned}
$$

TABLE 3. Examples

$$
\begin{aligned}
p_1(x,y) &= y^2 + x^3,\\
p_2(x,y) &= p_1^2(x,y) + x^5y,\\
p_3(x,y) &= p_2^2(x,y) + x^{10}p_1(x,y),\\
p_4(x,y) &= p_3^2(x,y) + x^{20}p_2(x,y),\\
p_5(x,y) &= p_4^2(x,y) + x^{40}p_3(x,y),\\
p_6(x,y) &= p_5^2(x,y) + x^{80}p_4(x,y),
\end{aligned}
$$

$$
\begin{aligned}
q_1(x,y) &= p_3(x,y)p_5(x,y),\\
q_2(x,y) &= p_4(x,y)p_5(x,y).
\end{aligned}
$$

TABLE 4. Examples from M. Merle

$$
\begin{aligned}
a_1(x,y) &= x^{14} - 17x^{13} + x^{12}(12y+16) - 72yx^{11} + 42y^2x^{10} - 40y^2x^9 + \\
&\quad 48y^3x^8 - 2y^4x^7 + 33y^4x^6 - 12y^5x^5 - 10y^6x^3 + y^8, \\
a_2(x,y) &= 100yx^{24} + x^{24} + 2y^{10}x^{12} + y^{20}, \\
a_{3.1}(x,y) &= y^{12} - 3x^2y^{11} - 6x^3y^{10} - x^3y^{11} + 3x^4y^{10} + 12x^5y^9 + 15x^6y^8 + \\
&\quad 3x^6y^9 - 6x^7y^8 - 18x^8y^7 - 20x^9y^6 - 6x^9y^7 + 3x^{10}y^6 + 12x^{11}y^5 + \\
&\quad 15x^{12}y^4 + 4x^{12}y^5 - 3x^{14}y^3 - 6x^{15}y^2 - x^{15}y^3 + x^{18}, \\
a_{3.2}(x,y) &= y^{12} - 3x^2y^{11} - 6x^3y^{10} + 2x^4y^{10} + 12x^5y^9 + 15x^6y^8 - x^6y^9 - \\
&\quad 2x^7y^8 - 18x^8y^7 - 20x^9y^6 - 3x^{10}y^6 + 12x^{11}y^5 + 15x^{12}y^4 + \\
&\quad 4x^{13}y^4 - 3x^{14}y^3 - 6x^{15}y^2 - x^{16}y^2 + x^{18}, \\
a_3(x,y) &= a_{3.1}(x,y)a_{3.2}(x,y).
\end{aligned}
$$

TABLE 5. Examples from V. Baladi

2. Genus

Again, let μ_q be the Milnor-numbers of $\mathcal{C}$ at q. The genus g of the curve $\mathcal{C}$ is then equal to

$$g = \sum_{q \text{ singular point of } \mathcal{C}} (\mu_q + I_q - 1),$$

where I_q is the number of curve-branches of $\mathcal{C}$ centered at q.

We already showed in the previous section how to compute μ_q. For the case $q = (0,0)$ we also showed that this requires no more than $\mathbf{O}(\mu^2 N)$ operations in the ground field k'.

Let q now be an arbitrary singular point of the curve $\mathcal{C}$ whose coordinates are algebraic numbers of degree d. Moving q by a linear transformation to the origin of a new coordinate system the defining polynomial of $\mathcal{C}$ (w.r.t. the new coordinate system) has in general coefficients of algebraic degree d.

Of course, all conjugated points of q are also singular points of $\mathcal{C}$ and they have the same Milnor-number. Computing the singular parts of the curve-branches centered at only one representative of the family of conjugated singularities and using the fact that the sum of the Milnor-numbers over all singular points is bounded by $(D-1)(D-2)$ we get the following result.

Lemma 32. *To compute the singular parts of the Puiseux parametrizations of one (and hence all) of the* d *conjugated singularities of* $\mathcal{C}$ *at* q *no more than* $3(D-1)^2(D-2)^2(N+1)$ *operations in* k' *are needed.*

Proof. Using again passive factorization to deal with the algebraic numbers introduced by the coordinates of q (as described in Subsection 4.3 of Chapter 4) it follows from Theorem 9 on page 77 that the computational complexity to compute the singular parts of all curve-branches centered at q is less than $3\mu_q^2(N+1)d^2$.

Since the sum of the Milnor-numbers of all the conjugated singularities of q is at most $(D-1)(D-2)$ (see e.g. [**Abhyankar, 1990**]), the Milnor-number μ_q has to be less than or equal to $\frac{(D-1)(D-2)}{d}$. $\square$

If we assume that we have already computed all singular points of $\mathcal{C}$ (which is a standard problem that can be solved by resultant- and gcd-computation), the previous Lemma extends in a straight forward way to a complexity bound for resolving all singularities of $\mathcal{C}$.

Theorem 10. *Assuming that all singular points of* $\mathcal{C}$ *are known, the number* C *of* k'*-operations necessary to compute the singular parts of all curve-branches of all singularities of* $\mathcal{C}$ *is bounded by*

$$C \leq 3(D-1)^2(D-2)^2(N+1).$$

	comSi	conPa	solAE	appSu	appTr	comPP	total
$f_1(x,y)$	0.010	0.020	0.020	0.000	0.000	0.181	0.211
$f_2(x,y)$	0.740	0.000	0.010	0.000	0.000	0.180	0.920
$f_3(x,y)$	0.840	1.019	0.211	0.000	0.101	0.680	2.550
$f_4(x,y)$	422.871	2703.319	130.500	0.000	0.000	551.141	3677.341
$f_5(x,y)$	23.480	81.572	5.178	0.000	0.000	5.499	110.551
$f_6(x,y)$	6.980	0.010	0.029	0.000	0.000	0.120	7.110

Abbreviation: comSi ... compute singularities, conPa ... convert to passive expansion form, solAE ... factor univariate polynomials over algebraic number field, appSu ... apply Newton substitutions, appTr ... apply Newton transformations, comPP ... compute singular part of Puiseux parametrizations.

TABLE 6. Computation times

PROOF. Similar to the proof of Lemma 32 using the fact that the sum of the Milnor-numbers over all singular points of $\mathcal{C}$ is also bounded by $(D-1)(D-2)$. □

So we get the following upper bound for the complexity of the genus computation of $\mathcal{C}$.

Theorem 11. *The genus* g *of* $\mathcal{C}$ *can be computed using no more than* $\mathbf{O}(D^5)$ k'*-operations.*

PROOF. From Subsection 1.4 we know that having computed the singular parts of all curve-branches of all singular points of $\mathcal{C}$ we basically also have computed the Milnor-numbers of all singularities of $\mathcal{C}$.

Hence the upper bound for the number of k'-operations necessary to compute the genus of $\mathcal{C}$ follows from Theorem 10 together with an analysis of the complexity of resultant- and gcd-computations which can be found e.g. in [**von zur Gathen and Gerhard, 1999**]. □

2.1. Experimental Results

The singular points of a curve are computed using resultants (done in in the procedure "comSi"). The procedure "conPa" moves singular points to the origin and converts the resulting polynomial into a passive expansion form. During this conversion the polynomial is fully expanded, which may become the most expensive operation of the whole genus computation.

Table 6 shows the computation times and in Table 7 the degree of the curves, their genus and the Milnor numbers of their singularities are given.

	deg.	genus	Milnor numbers
$f_1(x,y)$	6	0	13, 2
$f_2(x,y)$	10	13	44
$f_3(x,y)$	10	10	2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2
$f_4(x,y)$	9	0	6, 1
$f_5(x,y)$	27	0	32, 1, 1, 5, 7, 2
$f_6(x,y)$	16	45	106, 4

TABLE 7. Results

$$
\begin{aligned}
f_1(x,y) &= (x^2+y^2)^3 - 4x^2y^2,\\
f_2(x,y) &= (y^5 - x^2y^2 - xy^2 + x^3y + x^5)y^5 + (x^5 - y^2)x^5,\\
f_3(x,y) &= y^{10} + (-2494x^2 + 474)y^8 + (84366 + 2042158x^4 - 660492x^2)y^6 +\\
&\quad (-761328152x^6 + 128361096x^4 - 47970216x^2 + 6697080)y^4 +\\
&\quad (134266087241x^8 + 15052058268x^6 - 12024807786x^4 - 506101284x^2 +\\
&\quad 202172841)y^2 - 9127158539954x^{10} - 3212722859346x^8 -\\
&\quad 228715574724x^6 + 34263110700x^4 + 5431439286x^2 + 201803238,\\
f_4(x,y) &= 12xy^4 + 20yx^4 + 33x^2y^5 - 17x^9 + 20x^8 - 13x^7 + 4y^4 + 2yx^5 +\\
&\quad 37y^3x^5 - 37yx^7 - 18x^6y^2 - 48x^4y^4 + 132x^3y^3 - 64x^3y^2 -\\
&\quad 55x^4y^2 + 97yx^6 - 24xy^5 - 10x^2y^6 + 18xy^6 - 3x^7y^2 - 9x^6y^3 +\\
&\quad 18x^3y^5 + 16yx^8 - 116x^3y^4 - 84x^5y^2 + 71x^4y^3 + 9y^8 - 12y^7 -\\
&\quad 33x^2y^4 - 6xy^7 + 12x^3y + 48x^2y^3 - 28x^2y^2 + 8y^6 - 19x^5 -\\
&\quad 6x^6 - 4x^3 + 16x^4 - 8y^5,\\
f_5(x,y) &= -y^4 + 36x^{10}y + 120x^{13}y + 100yx^{16} + 2x^5y^2 - x^{10} + 81x^{15} +\\
&\quad 540x^{18} + 1350x^{21} + 1500x^{24} + 625x^{27},\\
f_6(x,y) &= y^{14} + 32x^5y^{11} - 241x^7y^8 + 2x^3y^8 + 128x^7y^4 + 12x^{10}y +\\
&\quad 35x^{14}y + x^{16}.
\end{aligned}
$$

TABLE 8. Examples

3. Linear System of Adjoint Curves

In this section we show how to compute the linear system of adjoint curves of degree d of $\mathcal{C}$.

For the sake of convenience we assume for this section that the curve $\mathcal{C}$ is given in projective space and defined by $f(x, y, z) \in k'[x, y, z]$. Let J be the number of singular points of $\mathcal{C}$. For each singular point q_j, for $1 \le j \le J$, let $\mathfrak{z}_{j,i}$, for $1 \le i \le I_j$, be the curve-branches of $\mathcal{C}$ which are centered at q_j. The Puiseux parametrization of $\mathfrak{z}_{j,i}$ is denoted by $(A_{j,i}(t), B_{j,i}(t), C_{j,i}(t))$.

3.1. Linear System of Curves

A curve $\mathcal{D} \subset \mathbf{P}^2$ of degree d is defined by a polynomial

$$g(x, y, z) = \sum_{i+j+k=d} a_{i,j,k} x^i y^j z^k.$$

So we can set up correspondence between the curves of degree d and the coefficients $a_{i,j,k}$. This correspondence is an isomorphism up to a common factor of the $a_{i,j,k}$'s. Hence the space of all curves of degree d can be identified with the projective space $\mathbf{P}^{N_d}$, where $N_d = \frac{d(d+3)}{2}$.

Definition 65. A linear subspace L of $\mathbf{P}^{N_d}$ is called *a linear system of curves of degree* d. If a point p is a point of multiplicity greater than or equal to m for every curve in the linear system L, p is called a *base point of multiplicity* m *of the linear system* L.

Definition 66. The linear system of curves of degree d consisting of all curves adjoint to $\mathcal{C}$ is called the *(linear) system of adjoint curves of degree* d *of* $\mathcal{C}$.

3.2. Computing the System of Adjoint Curves

The algorithm for computing the system of adjoints is based on the following idea (see Section 5 of Chapter 3):

(i) We fix a generic point $p \notin \mathcal{C}$. Let $\mathcal{P}$ be the polar curve of $\mathcal{C}$ w.r.t. p and let $\mathcal{L}_j$ be the line through p and q_j. The defining polynomials of $\mathcal{P}$ and $\mathcal{L}_j$ are denoted by $p(x, y, z)$ and $l_j(x, y, z)$, respectively.

(ii) Setting $d_{j,i}$ equal to

$$\begin{aligned} d_{j,i} = \mathbf{mult}_{q_j}(\mathfrak{z}_{j,i}, \mathcal{P}) - \mathbf{mult}_{q_j}(\mathfrak{z}_{j,i}, \mathcal{L}_j) + 1 = \\ \mathrm{ord}(p(A_{j,i}(t), B_{j,i}(t), C_{j,i}(t))) - \\ \mathrm{ord}(l_j(A_{j,i}(t), B_{j,i}(t), C_{j,i}(t))) + 1 \end{aligned}$$

(iii) and requiring that

$$\mathbf{mult}_{q_j}(\mathfrak{z}_{j,i}, \mathcal{D}) \ge d_{j,i}$$

which is equivalent to

$$\operatorname{ord}(g(A_{j,i}(t), B_{j,i}(t), C_{j,i}(t))) \geq d_{j,i}$$

yields linear conditions for the undetermined coefficients $a_{i,j,k}$ of $g(x, y, z)$.

(iv) Solving the resulting linear system we get a representation for the system of adjoint curves of degree d of $\mathcal{C}$.

Example 13. Let $\mathcal{C}$ be the curve defined by

$$f(x, y, z) = y^5 z - 2x^2y^2z^2 + 3yx^4z - x^6.$$

At the only singular point $q = (0, 0, 1)$ the curve decomposes into 3 branches, namely

$$\begin{aligned}
\mathfrak{z}_1 : A_1(t) &= t,\\
B_1(t) &= t^2 + t^6 + 3t^{10} + 13t^{14} + \cdots,\\
C_1(t) &= 1,\\
\mathfrak{z}_2 : A_2(t) &= t,\\
B_2(t) &= \frac{1}{2}t^2 - \frac{1}{32}t^6 + \frac{3}{256}t^{10} - \frac{13}{2048}t^{13} + \cdots,\\
C_1(t) &= 1,\\
\mathfrak{z}_3 : A_3(t) &= 2t^3,\\
B_3(t) &= 2t^2 - 2t^6 - \frac{8}{3}t^{10} - \frac{20}{3}t^{14} + \cdots,\\
C_1(t) &= 1.
\end{aligned}$$

and we get for $d_1 = 4$, $d_2 = 4$, and $d_3 = 6$.

A general curve $\mathcal{D}$ of degree 3 is defined by

$$\begin{aligned}
g(x, y, z) = a_{0,0,3}z^3 + a_{1,0,2}xz^2 + a_{0,1,2}yz^2 + a_{2,0,1}x^2z + a_{1,1,1}xyz +\\
a_{0,2,1}y^2z + a_{3,0,0}x^3 + a_{2,1,0}x^2y + a_{1,2,0}xy^2 + a_{0,3,0}y^3
\end{aligned}$$

and adjoint to $\mathcal{C}$ iff

$$\begin{aligned}
\operatorname{mult}_q(\mathfrak{z}_1, \mathcal{D}) = \operatorname{ord}(g(A_1(t), B_1(t), C_1(t))) =\\
\operatorname{ord}(a_{0,0} + a_{1,0}t + (a_{0,1} + a_{2,0})t^2 + (a_{1,1} + a_{3,0})t^3 + \cdots) \geq 4,
\end{aligned}$$

$$\begin{aligned}
\operatorname{mult}_q(\mathfrak{z}_2, \mathcal{D}) = \operatorname{ord}(g(A_2(t), B_2(t), C_2(t))) =\\
\operatorname{ord}(a_{0,0} + a_{1,0}t + (\frac{1}{2}a_{0,1} + a_{2,0})t^2 + (\frac{1}{2}a_{1,1} + a_{3,0})t^3 + \cdots) \geq 4,
\end{aligned}$$

$$\begin{aligned}
\operatorname{mult}_q(\mathfrak{z}_3, \mathcal{D}) = \operatorname{ord}(g(A_3(t), B_3(t), C_3(t))) =\\
\operatorname{ord}(a_{0,0} + 2a_{0,1}t^2 + 2a_{1,0}t^3 + 4a_{0,2}t^4 + 4a_{1,1}t^5 + \cdots) \geq 6.
\end{aligned}$$

Hence, the system of adjoint curves is given by

$$a_{2,1,0}x^2y + a_{1,2,0}xy^2 + a_{0,3,0}y^3.$$

We just remark, that the numbers $d_{j,i}$ (see Definition 52 on page 44) can also easily be computed within the Newton–Puiseux algorithm using Lemma 17 and Lemma 18 of Chapter 4 Section 2.

In Chapter 3 Section 5 we proved an alternative definition of adjoints based on truncated intersection multiplicities. This allows us to replace the Puiseux parametrizations by their singular parts which yields an effective version of the above adjoint algorithm requiring only polynomial arithmetic (no power series computations are needed). So we can state the following theorem.

Theorem 12. *The Algorithm in Appendix A Section 4 computes the system of adjoint curves of degree* d *using no more that* $\mathbf{O}(D^6)$ *operations in the ground field* k'.

PROOF. The correctness of the algorithm is a consequence of Theorem 7 on page 50. We just additionally remark that the singular part of the Puiseux parametrization computed by the Newton–Puiseux algorithm for a curve-branch $\mathfrak{z}$ is a local parametrization of $\mathfrak{z}$ truncated at T, where T is the length of the proper resolution sequence of $\mathfrak{z}$ within $\mathcal{C}$. Also the last coefficient in the Puiseux parametrization is already generic w.r.t. $\mathcal{C}$.

As we proved in Theorem 11 the singular parts of all curve-branches of all singular points of $\mathcal{C}$ can be computed in $\mathbf{O}(D^5)$ k'-operations. We can assume that during this computation also the numbers $d_{j,i}$ are calculated. So it remains to substitute the singular parts into the undetermined polynomial $g(x, y, z)$ and solve the resulting linear system. Using the facts that the sum over all $d_{j,i}$ is bounded by $(D-1)(D-2)$ (which is therefore an upper bound for the number of equations) and that there are at most $\frac{D(D+3)}{3}$ unknowns, we can conclude that building and solving the linear system requires no more than $\mathbf{O}(D^6)$ k'-operations. □

3.3. Experimental Results

In Table 9 the computation times are shown. Since the systems of adjoint curves may become quite complicated we do not list them in Table 10.

	comSi	conPa	solAE	comPP	subPP	solLE	total
$f_1(x,y)$	0.179	0.021	0.080	0.300	0.011	0.170	0.711
$f_2(x,y)$	23.030	67.471	9.040	21.179	5.511	1.019	125.029
$f_3(x,y)$	33.030	84.459	6.161	16.031	29.039	3.170	199.919
$f_4(x,y)$	1.231	0.250	0.119	0.420	0.058	0.661	2.870
$f_5(x,y)$	1.370	0.010	0.000	0.240	0.400	0.920	3.120
$f_6(x,y)$	1.060	1.160	0.317	0.900	0.197	0.339	3.799

Abbreviation: comSi ... compute singularities, conPa ... convert to passive expansion form, solAE ... factor univariate polynomials over algebraic number field, comPP ... compute singular part of Puiseux parametrizations, subPP ... substitute Puiseux parametrizations, solLE ... solve linear system of equations.

TABLE 9. Computation times

	deg.	genus	deg. of system of adjoint curves
$f_1(x,y)$	5	0	3
$f_2(x,y)$	6	0	4
$f_3(x,y)$	8	0	6
$f_4(x,y)$	8	0	6
$f_5(x,y)$	18	0	16
$f_6(x,y)$	10	10	7

TABLE 10. Results

$$
\begin{aligned}
f_1(x,y) &= y^5 + 2xy^2 + 2xy^3 + x^2y - 4x^3y + 2x^5,\\
f_2(x,y) &= 9186x^4y^2 + 175830y^3x^3 - 50376x^5y + 620067x^2 + 8675y^2 -\\
&\quad 821835x^2y^3 + 44073x^4y - 532159x^3y^2 - 6270x + 3600y + 87528x^5 +\\
&\quad 13199x^6 - 9750y^3 - 580706x^3 - 1512525x^2y + 1477245x^2y^2 -\\
&\quad 30099xy^2 - 179175y^3x + 865881x^3y - 25233x^4 + 141219xy +\\
&\quad 230100x^2y^4 + 79075xy^4 + 9750y^5x + 599,\\
f_3(x,y) &= -932960x^4y + 106624xy^2 + 2689120x^7 + 31360x^6y + 8704x + 13824y -\\
&\quad 9216y^2 - 60928yx - 100352x^6 - 59904x^3 - 6912 + 2048y^3 + 66816x^3y -\\
&\quad 17920x^3y^2 - 14000487x^8 - 268912x^7y - 186592x^4,\\
f_4(x,y) &= 4xy^6 + 2x^5y^3 - 16x^5y - x^4y^4 - 8x^2y^5 + 4x^7 - 32x^2y^3 - 20xy^4 -\\
&\quad 40x^3y - 16x^3y^4 + 6x^2y^4 - 4xy^7 - 12x^2y^2 - 4x^5y^2 - 20x^4y^2 +\\
&\quad x^8 - 32x^4 - 8y^6 + y^4 + 2xy^3 + y^8,\\
f_5(x,y) &= y^7 - 7y^6x^2 + 21y^5x^4 - 35y^4x^6 + 35x^8y^3 - 21y^2x^{10} + 7x^{12}y -\\
&\quad x^{14} + 78125x^{18},\\
f_6(x,y) &= y^{10} + (-2494x^2 + 474)y^8 + (84366 + 2042158x^4 - 660492x^2)y^6 +\\
&\quad (-761328152x^6 + 128361096x^4 - 47970216x^2 + 6697080)y^4 +\\
&\quad (134266087241x^8 + 15052058268x^6 - 12024807786x^4 -\\
&\quad 506101284x^2 + 202172841)y^2 - 9127158539954x^{10} - 3212722859346x^8 -\\
&\quad 228715574724x^6 + 34263110700x^4 + 5431439286x^2 + 201803238.
\end{aligned}
$$

TABLE 11. Examples

4. Parametrization

An often encountered problem in algorithmic algebraic geometry is the problem of parametrizing rational curves. An algebraic curve is defined as the zero set of a bivariate polynomial. Some curves, namely exactly the rational ones, can also be defined by a parametrization.

An algorithm for computing a parametrization of a rational curve can be found in [**Sendra and Winkler, 1991**]. From a computational point of view the most expensive operation in this algorithm is the computation of a system of adjoint curves of some given degree. In the following we will briefly describe this algorithm.

Definition 67. The curve $\mathcal{C}$ is called *rational* or *parametrizable* if there exist two rational functions $x(t)$, $y(t) \in k(t)$ not both constant such that $f(x(t), y(t)) = 0$.

A criterion for checking whether a curve is rational or not is given by the genus. It holds that an irreducible plane algebraic curve is rational if and only if its genus is zero.

The algorithm described in [**Sendra and Winkler, 1991**] consists of the following three main steps:

(i) Compute the linear system L of adjoint curves of degree $D-2$ of $\mathcal{C}$; the degrees $D-1$ or D would also be possible.
(ii) Find additional points on the curve $\mathcal{C}$ and make it base points of L to lower the dimension of L to 1.
(iii) Determine the one free intersection point of L and $\mathcal{C}$ depending on the only free parameter of L.

The algorithm above can be extended to compute parametrizations over optimal field extensions, as described in [**Sendra and Winkler, 1997**].

Remark: Step (ii) in the parametrization algorithm does nothing else then to increase the intersection multiplicity of a generic curve in L and $\mathcal{C}$. Having computed Puiseux parametrizations of the curve-branches this can be done directly by increasing the numbers $d_{j,i}$ in the adjoint algorithm. Doing so one has to take care, that now also some terms of regular parts of some curve-branches are required (but we know exactly how many). Hence, the complexity for computing the linear system L is basically the same as the one for computing the system of adjoints – disregarding optimal field extensions.

A complexity analysis for the parametrization problem (not taking into account resolution of singularities) can be found in [**Mňuk et al., 1996**].

APPENDIX A

Algorithms

1. Resolution by Quadratic Transformations

■ `resolveQT(f(x,y))`

In: A polynomial $f(x,y) \in k[x,y]$ defining a curve $\mathcal{C}$.

Out: The resolution graph of $\mathcal{C}$ at the origin.

begin
(1) **return** `applyQT(f(x,y), 0)`
end

■ `applyQT(f(x,y), η)`

In: A polynomial $f(x,y) \in k[x,y]$ defining a curve $\mathcal{C}$ and $\eta \in k$.

Out: The resolution graph of $\mathcal{C}$ at $(0,\eta)$.

begin
(1) $G \leftarrow [\,]$

(2) $f(x,y) \leftarrow f(x,y+\eta)$
(3) $m \leftarrow \mathbf{mult}_0(f)$

(4) **if** $((\eta = 0)$ **or** $(m > 1))$ **then**

(5) $f^{I}(x,y) \leftarrow x^{-m} f(x,xy)$
(6) **for** η^{I} **root of** $f^{I}(0,y)$ **do**
(6.1) **for** S **in** `applyQT(`$f^{I}(x,y)$`, `η^{I}`)` **do**
(6.1.1) $G \leftarrow G \oplus ([\text{'I'}, m, \eta] \oplus S)$
(6.2) **od**
(7) **od**

(8) $f^{II}(x,y) \leftarrow y^{-m} f(xy,y)$
(9) **if** $(f^{II}(0,0) = 0)$ **then**
(9.1) **for** S **in** `applyQT(`$f^{II}(x,y)$`, 0)` **do**
(9.1.1) $G \leftarrow G \oplus ([\text{'II'}, m, \eta] \oplus S)$
(9.2) **od**
(10) **fi**

(11) **else**

(12) $\quad G \leftarrow G \oplus [[\text{`I`}, m, \eta]]$
(13) **fi**

(14) **return** G
end

Remark:

Quadratic transformations of type I and type II are treated in steps (5)–(7) and (8)–(10) in the procodure "`applyQT`", respectively.

2. Resolution of Curve-branches

■ `resolvePE(A(t), B(t))`

In: The singular part $(A(t), B(t))$, where $A(t)$, $B(t) \in k[t]$, of a Puiseux parametrization of a curve-branch $\mathfrak{z}$ centered at the origin.

Out: The resolution sequence of $\mathfrak{z}$.

begin
(1) $S \leftarrow [\,]$
(2) $d_x \leftarrow \deg(A(t))$
(3) $d_y \leftarrow \deg(B(t))$
(4) $d \leftarrow \max(d_x, d_y)$

(5) **while** $((d_x \geq 0$ **and** $d_y > 0)$ **or** $(d_x > 0$ **and** $d_y \geq 0))$ **do**
(5.1) $\eta \leftarrow B(0)$
(5.2) $B(t) \leftarrow B(t) - eta$
(5.3) $m_x \leftarrow \mathrm{ord}(A(t))$
(5.4) $m_y \leftarrow \mathrm{ord}(B(t))$
(5.5) $m \leftarrow \min(m_x, m_y)$

(5.6) **if** $(m_x \leq m_y)$ **then**
(5.6.1) $t \leftarrow$ 'I'
(5.6.2) $B(t) \leftarrow B(t)/A(t)$
(5.6.3) $d_y \leftarrow d_y - m$
(5.7) **else**
(5.7.1) $t \leftarrow$ 'II'
(5.7.2) $A(t) \leftarrow A(t)/B(t)$
(5.7.3) $d_x \leftarrow d_x - m$
(5.8) **fi**

(5.9) $d \leftarrow \max(d_x, d_y)$
(5.10) $S \leftarrow S \oplus [t, m, \eta]$

(6) **od**

(7) **return** S
end

Remark:

In general, the power series quotients in steps (5.6.2) and (5.7.2) are infinite power series. But since we only need to know the series $A(t)$ and $B(t)$ up to order $d+1$, see Subsection 1.1 in Chapter 5, and d is decreasing, any lazy evaluation of the coefficients of $A(t)$ and $B(t)$ can be avoided.

3. Newton–Puiseux Algorithm

■ `newton-puiseux(f(x, y))`

In: A polynomial $f(x,y) \in k'[x,y]$ defining a curve $\mathcal{C}$.

Out: The singular parts of a system of rational Puiseux parametrizations of $\mathcal{C}$ centered at the origin w.r.t. k'.

begin
(1) $S \leftarrow$ `computeSingularPart(f, [])`
(2) $P \leftarrow [\,]$

(3) **for** T **in** S **do**
(3.1) $P \leftarrow P \oplus$ `constructExpansion(T)`
(4) **od**

(5) **return** P
end

■ `computeSingularPart(f(x, y), T)`

In: A polynomial $f(x,y) \in k[x,y]$ defining a transformed curve $\mathcal{C}$, a sequence T of Newton transformations.

Out: A list of sequences of Newton transformations corresponding to the resolution sequences of the curve-branches of $\mathcal{C}$ centered at the origin.

begin
(1) $S \leftarrow [\,]$
(2) **for** (τ, m) **in** `computeNewtonTransformation(f(x, y))` **do**
(2.1) $T' \leftarrow T \oplus \tau$
(2.2) $f'(x,y) \leftarrow$ `applyNewtonTransformation(f(x, y), τ)`

(2.3) **if** $(m = 1)$ **then**
(2.3.1) $S \leftarrow S \oplus T'$
(2.4) **else**
(2.4.1) $S \leftarrow S \oplus$ `computeSingularPart(f'(x, y), T')`
(2.5) **fi**

(3) **od**

(4) **return** S
end

■ `computeNewtonTransformation(f(x, y))`

In: A polynomial $f(x,y) \in k[x,y]$.

Out: A list of Newton transformations corresponding to $f(x,y)$.

begin
(1) $S \leftarrow [\,]$
(2) $P \leftarrow$ `computeNewtonPolygon(f(x,y))`

(3) **for** $(p, q, r, \Phi(z))$ **in** P **do**
(3.1) **for** $(\Psi(z), m)$ **in** `factorSquarefree(Φ(z))` **do**
(3.1.1) **for** ζ **root of** $\Psi(z)$ **do**
(3.1.1.1) $(a, b) \leftarrow$ `computeNewtonCoefficient(p, q, ζ)`
(3.1.1.2) $\tau \leftarrow [p, q, r, a, b]$
(3.1.1.3) $S \leftarrow S \oplus [\tau, m]$
(3.1.2) **od**
(3.2) **od**
(4) **od**

(5) **return** S
end

■ `applyNewtonTransformation(f(x,y), τ)`

In: A polynomial $f(x, y) \in k[x, y]$, a Newton transformation τ of the form $\tau = [p, q, r, a, b]$.

Out: The polynomial $\tau(f(x, y)) = x^{-r} f(bx^q, x^p(a + y))$.

■ `computeNewtonPolygon(f(x,y))`

In: A polynomial $f(x, y) = \sum_{i=0}^{M} \sum_{j=0}^{N} a_{i,j} x^i y^j \in k[x, y]$.

Out: A list of 4-tuples $(p, q, r, \Phi(z))$, where for each segment S of the Newton polygon of $f(x, y)$ the numbers p, q, $r \in \mathbb{Z}$ are defined according to Definition 55 on page 51. The polynomial $\Phi(z) \in k[z]$ is equal to $\Phi(z) = \sum_{(i,j)\in S} a_{i,j} z^{(i-i_0)/q}$, where i_0 is the smallest value of i such that there is a point (i, j) in S.

■ `factorSquarefree(Φ(z))`

In: A polynomial $\Phi(z) \in k[z]$.

Out: A list of pairs $(\Psi_i(z), m_i)_{1 \le i \le n}$, with $\Psi_i(z) \in k[z]$, $m_i \in \mathbb{N}$, such that $\Phi(z) = \prod_{i=1}^{n} \Psi_i^{m_i}(z)$ and $\Psi_i(z)$ is squarefree, for $1 \le i \le n$.

■ `computeNewtonCoefficient(p, q, ζ)`

In: Two numbers p, $q \in \mathbb{Z}$, with $q > 0$, and $\zeta \in k$. For the case $p = 0$ we require that $q = 1$.

Out: A pair (a, b) with a, $b \in k$, such that $\zeta = a^q b^{-p}$.

begin
(1) **if** $(p = 0)$ **then**
(1.1) $(a, b) \leftarrow (\zeta, 1)$
(2) **else**
(2.1) $(u, v) \leftarrow$ `extGCD(p, q)`
(2.2) $(a, b) \leftarrow (\zeta^u, \zeta^{-v})$
(3) **fi**

(4) **return** (a, b)
end

■ `extCGD(p, q)`

In: Two numbers p, $q \in \mathbb{Z}$, with $p \neq 0$, $q > 0$.

Out: Two numbers u, $v \in \mathbb{Z}$, such that $uq + vp = 1$. Note that u and v are not uniquely determined.

■ `constructExpansion(T)`

In: A sequence of Newton transformations.

Out: A Puiseux parametrization corresponding to T.

begin
(1) $T' \leftarrow$ `reverseSequence(T)`
(2) $A(t) = t$
(3) $B(t) = 0$

(4) **for** (p, q, r, a, b) **in** T' **do**
(4.1) $B(t) \leftarrow A^p(t)(a + B(t))$
(4.2) $A(t) \leftarrow bA^q(t)$
(5) **od**

(6) **return** $(A(t), B(t))$
end

Remark:

In order to get for the Puiseux parametrizations small rational number coefficients we can improve the procedure `computeNewtonCoefficient` as follows.

Let $\tau = [p, q, r, a, b]$ be a rational Newton transformation, i.e. $a = \zeta^u$, $b = \zeta^{-v}$, and $uq + vp = 1$.

As mentioned above the numbers u, v are *not uniquely determined.* Since we are interested in small coefficients in the Puiseuz parametrizations we will choose u and v as small as possible. In the case where $\zeta = \eta^d$, for $\eta \in \mathbb{Q}$ we solve instead of the equation $uq + vp = 1$ the equation

$u'q + v'p = d$. Again with try to find “smallest possible” solutions u' and v'.

The coefficients for the Puiseux parametrization are then $a' = \eta^{u'}$ and $b' = \eta^{-v'}$. In the case where ζ is negative and d is even we have change the sign of either a or b depending on whether q is odd or p is odd.

To show that a' and b' are valid Newton coefficients we define

$$\begin{aligned} a &= \zeta^{u} & a' &= s_a \eta^{u'} \\ b &= \zeta^{-v} & b' &= s_b \eta^{-v'}, \end{aligned}$$

with s_a, $s_b \in \{-1, 1\}$.

Let $\Phi(z) = \sum_j c_j z^j$ be the defining equation for ζ, where z^j stands for $x^{-p}y^{q}$. Of course a and b are valid Newton coefficients since it holds that (recall the definition of Newton transformation: $x \to bx^q$, $y \to x^p(a+y)$)

$$\sum_j \gamma_j (b^{-p}a^{q})^j = \sum_j \gamma_j \left((\zeta^{-v})^{-p}(\zeta^{u})^{q}\right)^j = \sum_j \gamma_j \zeta^{(vp+uq)j} = \sum_j \gamma_j \zeta^j = 0.$$

If we now replace ζ by η we get

$$\begin{aligned} \sum_j \gamma_j (b'^{-p}a'^{q})^j &= \sum_j \gamma_j \left((s_b\eta^{-v'})^{-p}(s_a\eta^{u'})^{q}\right)^j \\ &= \sum_j \gamma_j \left(s_b^{-p} s_a^{q} \eta^{v'p+u'q}\right)^j = \sum_j \gamma_j (s\eta^{d})^j. \end{aligned}$$

Hence we have to chose s_a and s_b such that

$$\zeta = s\eta^{d} = s\eta^{v'p+u'q} = (s_b\eta^{-v'})^{-p}(s_a\eta^{u'})^{q} = (b')^{-p}(a')^{q},$$

which is always possible since not both p and q can be even.

4. Linear System of Adjoint Curves

■ `adjoints(f(x,y,z), d)`

In: A polynomial $f(x,y,z) \in k'[x,y,z]$ defining a curve $\mathcal{C}$ and a natural number d.

Out: The linear system of adjoint curves of degree d of $\mathcal{C}$.

begin

(1) $g(x,y,z) \leftarrow \sum_{i+j+k=d} a_{i,j,k} x^i y^j z^k$

(2) $E \leftarrow \{\,\}$

(3) $Q \leftarrow$ `singularPoints(f(x,y,z))`

(4) **for** q **in** Q **do**

(4.1) $P \leftarrow$ `newton-puiseux(f(x,y,z), q)`

(4.2) **for** $(A(t), B(t), C(t), c)$ **in** P **do**

(4.2.1) $g(t) \leftarrow g(A(t), B(t), C(t))$

(4.2.2) **for** i **from** 0 **to** c **do**

(4.2.2.1) $E \leftarrow E \cup$ `setCoeffZero(g(t), i)`

(4.2.3) **od**

(4.3) **od**

(5) **od**

(6) $A \leftarrow$ `makeLinearSystem(E, d)`

(7) **return** A

end

■ `singularPoints(f(x,y,z))`

In: A polynomial $f(x,y,z) \in k'[x,y,z]$ defining a curve $\mathcal{C}$.

Out: The set of singular points of $\mathcal{C}$.

■ `newton-puiseux(f(x,y,z), q)`

In: A polynomial $f(x,y,z) \in k'[x,y,z]$ defining a curve $\mathcal{C}$ and a point $q \in \mathbf{P}^2$.

Out: The singular parts of a system of rational Puiseux parametrizations of $\mathcal{C}$ centered at q w.r.t. k' together with its conductor-numbers.

■ `setCoeffZero(g(t), i)`

In: A polynomial $g(t) \in k'[t]$ and a non negative integer i.

Out: The set of equations such that the coefficient of t^i in $g(t)$ is zero.

■ `makeLinearSystem(E, d)`

In: A set of equations E in the variables $a_{i,j,k}$ and a natural number d.

Out: The linear system of curves of degree d of the form $g(x, y, z) = \sum_{i+j+k=d} a_{i,j,k} x^i y^j z^k$ such that the $a_{i,j,k}$'s are in the null-space of E.

Bibliography

[Abhyankar, 1967] Abhyankar, S. S. (1967). Inversion and invariance of characteristic pairs. *Amer. J. Math.*, 89:363–372.

[Abhyankar, 1976] Abhyankar, S. S. (1976). Historical ramblings in algebraic geometry and related algebra. *American Mathematical Monthly*, 83:409–448.

[Abhyankar, 1977] Abhyankar, S. S. (1977). On the semigroup of a meromorphic curve, part i. In *Proceedings of the International Symposium on Algebraic Geometry, Kyoto*, pages 240–414.

[Abhyankar, 1990] Abhyankar, S. S. (1990). *Algebraic Geometry for Scientists and Engineers.* American Mathematical Society.

[Bliss, 1933] Bliss, G. A. (1933). *Algebraic functions.* American Mathematical Society.

[Brieskorn and Knörrer, 1986] Brieskorn, E. and Knörrer, H. (1986). *Plane Algebraic Curves.* Birkhäuser.

[Dicrescenzo and Duval, 1984] Dicrescenzo, C. and Duval, D. (1984). Computations on curves. In *Lecture Notes in Comput. Sci.*, volume 174, pages 100–107. Springer.

[Dicrescenzo and Duval, 1985] Dicrescenzo, C. and Duval, D. (1985). Algebraic computations on algebraic numbers. In *Informatique et Calcul*, pages 54–61. Wiley-Masson.

[Duval, 1989] Duval, D. (1989). Rational puiseux expansion. *Compositio Math.*, 70:119–154.

[Duval, 1994] Duval, D. (1994). Algebraic numbers: An example of dynamic evaluation. *J. Symbolic Computation*, 18:429–445.

[Hartshorne, 1977] Hartshorne, R. (1977). *Algebraic Geometry.* Springer.

[Hemmecke et al., 2000] Hemmecke, R., Hillgarter, E., and Winkler, F. (2000). The casa system. In Grabmeier, J., Kaltofen, E., and Weispfenning, V., editors, *Handbook of Computer Algebra: Foundations, Applications, Systems.* Springer.

[Henry and Merle, 1987] Henry, J. P. G. and Merle, M. (1987). Complexity of computation of embedded resolution of algebraic curves. In *EUROCAL '87*, LNCS 378, pages 381–390. Springer.

[Keller, 1974] Keller, O.-H. (1974). *Vorlesung über algebraische Geometrie.* Akademische Verlagsgesellschaft Geest und Portig, Leipzig.

[Kozen, 1994] Kozen, D. (1994). Efficient resolution of singularities of plane curves. In *Foundations of software technology and theoretical computer science, Madras, India*, LNCS 880, pages 1–11. Springer.

[Mňuk, 1995] Mňuk, M. (1995). *Algebraic and Geometric Approach to Parametrization of Rational Curves.* PhD thesis, Johannes Kepler Universität, 4040 Linz, Austria, Europe.

[Mňuk et al., 1996] Mňuk, M., Sendra, J. R., and Winkler, F. (1996). On the complexity of parametrizing curves. *Beiträge zur Algebra und Geometrie*, 37(2):309–327.

[Mňuk and Winkler, 1996] Mňuk, M. and Winkler, F. (1996). Casa – a system for computer aided constructive algebraic geometry. In *International Symposium on Design and Implementation of Symbolic Computation Systems (DISCO'96), Karlsruhe, Germany*, LNCS 1128, pages 297–307. Springer.

[Moreno Socías, 1990] Moreno Socías, G. (1990). Lazy resolution of plane curves. Technical report, Équipe de Calcul Formel, Centre de Mathématiques & LIX, École polytechnique, F-91128 Palaiseau cedex, France.

[Mumford, 1976] Mumford, D. (1976). *Algebraic Geometry I, Complex Projective Varieties*. Springer.

[Newton, 1660] Newton, I. (1660). *Geometria Analitica*. Publisher unknown.

[Oka, 1996] Oka, M. (1996). Geometry of plane curves via toroidal resolution. In López, A. C. and Macarro, L. N., editors, *Algebraic Geometry and Singularities*, volume 134 of *Progress in Mathematics*, pages 95–121. Birkhäuser.

[Puiseux, 1850] Puiseux, V. A. (1850). Recherches sur les fonctions algebriques. *Journal de Mathématique pures et appliquées*, 15(9):365–480.

[Schicho, 1995] Schicho, J. (1995). *Rational Parametrization of Algebraic Surfaces*. PhD thesis, Johannes Kepler Universität, 4040 Linz, Austria, Europe.

[Sendra and Winkler, 1991] Sendra, J. R. and Winkler, F. (1991). Symbolic parametrization of curves. *J. Symbolic Computation*, 12:607–631.

[Sendra and Winkler, 1997] Sendra, J. R. and Winkler, F. (1997). Parametrization of algebraic curves over optimal field extensions. *J. Symbolic Computation*, 23:191–208.

[Shafarevich, 1994] Shafarevich, I. R. (1994). *Basic Algebraic Geometry 1, Varieties in Projective Space*. Springer-Verlag.

[Teitelbaum, 1990] Teitelbaum, J. (1990). The computational complexity of the resolution of plane curve singularities. *Mathematics of Computation*, pages 797–837.

[van der Waerden, 1973] van der Waerden, B. L. (1973). *Einführung in die algebraische Geometrie*. Springer-Verlag.

[van der Waerden, 1991] van der Waerden, B. L. (1991). *Algebra, Volume I*. Springer-Verlag.

[von zur Gathen and Gerhard, 1999] von zur Gathen, J. and Gerhard, J. (1999). *Modern Computer Algebra*. Cambridge University Press.

[Walker, 1978] Walker, R. J. (1978). *Algebraic curves*. Springer.

[Walsh, 1998] Walsh, P. G. (1998). A polynomial complexity bound for computations on curves. *SIAM J. Comput.*, 28(2):704–708.

[Walsh, 1999] Walsh, P. G. (1999). On the complexity of rational puiseux expansions. *Pacific J. of Mathematics*, 188(2):369–387.

[Winkler, 1996] Winkler, F. (1996). *Polynomial Algorithms in Computer Algebra*. Springer.

Curriculum Vitae

Personal

Name	Peter Stadelmeyer
Date of birth	August 20, 1970
Place of birth	Wels, Austria
Nationality	Austria

Education

1977 – 1981	Dr. Renner Volkschule, Marchtrenk.
1981 – 1989	Bundesrealgymnasium Anton-Bruckner Straße, Wels.
1989 – 1996	Diploma studies in Mathematics: Johannes Kepler Universität, Linz. Diploma thesis: "Ordered Completion with Selection for Horn Clauses". M.Sc. with honors.
1996 – 2000	PhD studies in Mathematics: RISC–Linz, Johannes Kepler Universität, Linz. PhD thesis: "On the Computational Complexity of Resolving Curve Singularities and Related Problems".

Remarks

1998 – 1999	Civil service (12 months)

Publications

Stadelmeyer, P. "MGrobner: A Maple Package for Computing Groebnerbases in Modules", RISC-rep 93-60.

Stadelmeyer, P. "Ordered Completion with Selection for Horn Clauses", RISC-rep 95-52.

Stadelmeyer, P. and Winkler, F. "Computing the System of Adjoint Plane Curves by Puiseux Expansion", RISC-rep 97-38.

Stadelmeyer, P. "Computing the System of Adjoint Plane Curves by Puiseux Expansion, Part II: Algorithm", RISC-rep 00-04.

Stadelmeyer, P. "Algebraic Extensions in the Resolution of Curve Singularities", RISC-rep 00-10.

Stadelmeyer, P. "Computing the System of Adjoint Plane Curves by Puiseux Expansion, Part III: Complexity", RISC-rep 00-20.

Conferences

54. Arbeitstagung Allgemeine Algebra, Klagenfurt, 1997. Stadelmeyer, P. and Winkler, F. "Computing Adjoints of Rational Curves".

Working Week on Resolution of Singularities, Tirol, 1997.

DMV-Tagung 2000, Dresden, 2000. Stadelmeyer, P. "Ein schneller Resolutionsalgorithmus für Kurvensingularitäten".