June 2017

Computing Limit Points of Quasi-components of Regular Chains and its Applications

Parisa Alvandi
The University of Western Ontario

Supervisor
Dr. Marc Moreno Maza
The University of Western Ontario

Graduate Program in Computer Science

A thesis submitted in partial fulfillment of the requirements for the degree in Doctor of Philosophy

Abstract

Computing limits is a fundamental task in mathematics and different mathematical concepts are defined in terms of limit computations. Among these mathematical concepts, we are interested in three different types of limit computations: first, computing the limit points of solutions of polynomial systems represented by regular chains, second, computing tangent cones of space curves at their singular points which can be viewed as computing limit of secant lines, and third, computing the limit of real multivariate rational functions.

For computing the limit of solutions of polynomial systems represented by regular chains, we present two different methods based on Puiseux series expansions and linear changes of coordinates. The first method, which is based on Puiseux series expansions, addresses the problem of computing real and complex limit points corresponding to regular chains of dimension one. The second method studies regular chains under changes of coordinates. It especially computes the limit points corresponding to regular chains of dimension higher than one for some cases. we consider strategies where these changes of coordinates can be either generic or guided by the input.

For computing the Puiseux parametrizations corresponding to regular chains of dimension one, we rely on extended Hensel construction (EHC). The Extended Hensel Construction is a procedure which, for an input bivariate polynomial with complex coefficients, can serve the same purpose as the Newton-Puiseux algorithm, and, for the multivariate case, can be seen as an effective variant of Jung-Abhyankar Theorem. We show that the EHC requires only linear algebra and univariate polynomial arithmetic. We deduce complexity estimates and report on a software implementation together with experimental results.

We also outline a method for computing the tangent cone of a space curve at any of its points. We rely on the theory of regular chains and Puiseux series expansions. Our approach is novel in that it explicitly constructs the tangent cone at arbitrary and possibly irrational points without using a standard basis.

We also present an algorithm for determining the existence of the limit of a real multivariate rational function q at a given point which is an isolated zero of the denominator of q. When the limit exists, the algorithm computes it, without making any assumption on the number of variables. A process, which extends the work of Cadavid, Molina and Velez, reduces the multivariate setting to computing limits of bivariate rational functions. By using regular chain theory and triangular decomposition of semi-algebraic systems, we avoid the computation of singular loci and the decomposition of algebraic sets into irreducible components.

Keywords: Regular chains, quasi-components, limit points, tangent cone, limit of multivariate rational functions, extended Hensel construction.

Acknowlegements

I would like to thank all the people who contributed in some way to the work described in this thesis. First, I would like to express my sincere appreciation and gratitude to professor Marc Moreno Maza for his guidance during my research. His support and inspiring suggestions have been precious for the accomplishments of this thesis content. During my study at The University of Western Ontario, he contributed to a rewarding graduate school experience by supporting my attendance at various conferences, engaging me in new ideas, and demanding a high quality of work in all my endeavors.

Additionally, I would like to thank my committee members Professor Agnes Szanto, Professor Jan Minac, Professor Lila Kari, and Professor Olga Veksler for their interest in my work.

Every result described in this thesis was accomplished with the help and support of fellow labmates and collaborators. I feel honoured to collaborate with my brilliant, insightful co-authors: Professor Amir Hashemi, Professor Éric Schost, Dr. Changbo Chen, Dr. Paul Vrbik, Dr. Masoud Ataei, and Mahsa Kazemi.

Finally, I would like to acknowledge friends and family who supported me during my time here. First and foremost, I would like to thank my mother and father , Fariba and Hossein, and my brothers, Mostafa, Mohammad, and Reza, for their great love, support and sacrifices. I also would like to thank my friends Sugi Magesan and Elham Karami for supporting me all along this way. I am also grateful for their help in proofreading some chapters of my thesis.

Contents

List of Algorithms

List of Figures

List of Tables

Chapter 1

Overview

In computational mathematics, computer algebra, also known as symbolic computation, is an area that develops algorithms and software for solving mathematical problems by means of exact computation. Here, exact computation implies that mathematical entities are represented by their formal definitions instead of using approximations. For instance, computer algebra algorithms may represent $\sqrt{2}$ as the positive solution of $x^2 = 2$ while numerical methods would generally use a floating point approximation like 1.414213562.

The software that perform symbolic computations are called computer algebra systems. Some of the most famous computer algebra systems include `Singular`, `Aldor`, MAPLE, `Mathematica`, `AXIOM`, `CoCoA`, `MAGMA`, and many others. Development of computer algebra systems began in 1960s mainly to serve the two fields of theoretical physics and Artificial Intelligence[1].

Nowadays, general-purpose computer algebra systems can solve a wide range of mathematical problems from factoring an integer number, solving linear systems, determining the real roots of a univariate polynomial to solving a system of non-linear equations, and solving a system of partial differential equations (see Figure 1.1). Hence, computer algebra systems manipulate numbers, matrices, polynomials and other algebraic objects. These capabilities stems from the fact that many algebraic properties can be stated in terms of closed-form expressions and established by means of rewriting systems.

While computer algebra systems can perform highly sophisticated algebraic tasks, such as solving (formally) a system of partial differential equations, they are much less equipped for solving problems from analysis in a symbolic, or exact manner. Some problems in analysis, like the manipulation of Taylor series and the calculation of limits of univariate functions, that is essentially undergraduate univariate calculus, is available (with some limitation) in general-purpose computer algebra systems such as MAPLE and `Mathematica`. However, limits of multivariate functions and more advanced notions of limits, e.g. topological closures, are almost absent from such systems. For instance, MAPLE is not capable of computing limits of rational functions in more than two variables.

Many fundamental concepts in mathematics are defined in terms of limits and it is desirable for computer algebra to implement those concepts. However, they are, by essence, hard to compute, or even not computable, in an algorithmic fashion, say by doing finitely many

[1] See `https://en.wikipedia.org/wiki/Computer_algebra_system` for more details about computer algebra systems, as of May 2017.

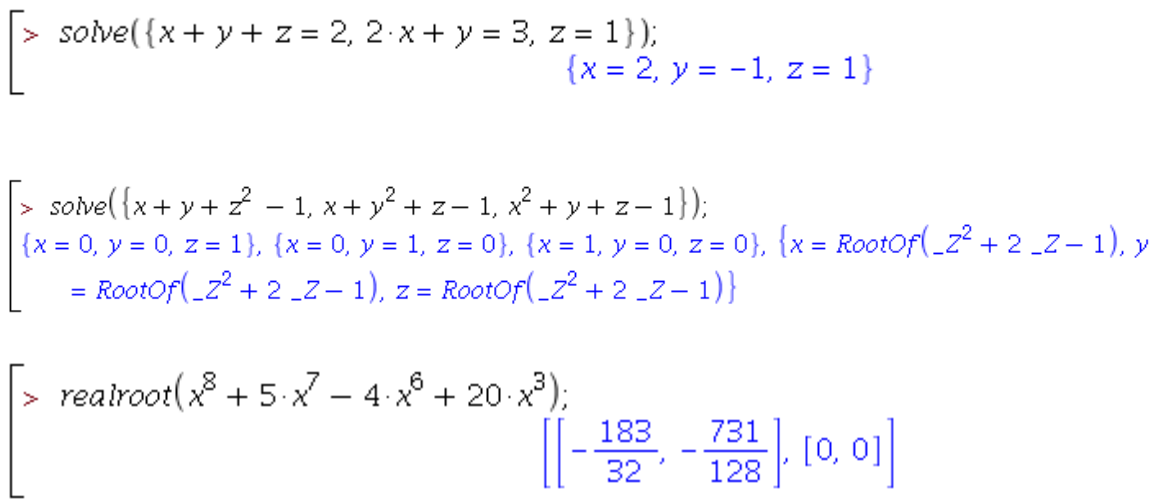


Figure 1.1: The computer algebra system MAPLE uses commands `solve` and `RealRoot` to solve polynomial systems; the command `solve` can solve both linear and non-linear systems; the command `RealRoot` isolates the real roots of univariate polynomials in intervals.

rational operations on polynomials or matrices, over the usual coefficient fields of symbolic computation.

Let us give an example of a limit computation process by means of computer algebra tools. This example is taken from Computer Aided Geometric Design and uses an important question raised in studying algebraic surfaces. Given an algebraic surface $S \subseteq \mathbb{R}^3$ and a parametrization of S described by the following mapping:

$$\begin{array}{rccc} \boldsymbol{r}: & \mathbb{R}^2 & \rightarrow & \mathbb{R}^3 \\ & (s,t) & \mapsto & \boldsymbol{r}(s,t) \end{array}$$

determine whether Image($\boldsymbol{r}$) = S holds or not. That is, determine whether every point of S can be reached with the parametrization $\boldsymbol{r}$.

Consider the Roman surface or Steiner surface[2](see Figure 1.2) with implicit formula $f = 0$, where f is the following polynomials in the variables x, y, z:

$$\begin{array}{l} f := 4\,x^4 - 8\,yx^3 + 9\,x^2y^2 - 8\,yzx^2 - 5\,y^3x + 8\,y^2zx + y^4 \\ -2\,y^3z + 3\,y^2z^2 - 2\,yz^3 + z^4 - 8\,yx^2 + 8\,zx^2 + 8\,y^2x \\ -8\,xyz - 2\,y^3 + 2\,y^2z - 2\,yz^2 + 4\,x^2 - 4\,yx + y^2. \end{array} \tag{1.1}$$

With $q(s,t) := s^2 + t^2 + s - t + 1$, consider also the following map

$$\begin{array}{rccc} \boldsymbol{r}: & \mathbb{R}^2 & \rightarrow & \mathbb{R}^3 \\ & (s,t) & \mapsto & \left(\frac{s^2}{q(s,t)}, \frac{s^2+t^2}{q(s,t)}, \frac{s^2+s\,t+s+t}{q(s,t)}\right), \end{array} \tag{1.2}$$

[2]see `https://en.wikipedia.org/wiki/Roman_surface` for more information about Steiner, also called Roman, surface, as of May 2017.

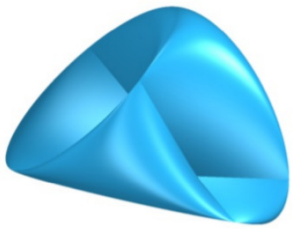


Figure 1.2: The Steiner surface S (this image is taken from [86]).

We want to determine whether Image($\boldsymbol{r}$) $= S$ holds or not. The Maple session shown on Figure 1.3 proves that $q(s,t)$ does not vanish over the reals while the Maple session shown on Figure 1.4 implies that Image($\boldsymbol{r}$) $\subseteq S$ holds.

```
> R := PolynomialRing([s, t, x, y, z]) : q := s^2 + t^2 + s - t + 1 :
  RealTriangularize([q], R);

                                [ ]
```

Figure 1.3: $q(s,t) := s^2 + t^2 + s - t + 1 = 0$ does not have any real solutions.

We now verify that the equality Image($\boldsymbol{r}$) $= S$ does not hold. To do so, we shall compare the points of Image($\boldsymbol{r}$) satisfying $y = 1$ with the points of S satisfying $y = 1$.

First, we compute the intersection of the image of the parametrization $\boldsymbol{r}$ with the plane $y = 1$. This can be determined to be the ellipse calculated in the Maple session shown on Figure 1.5, namely the plane curve with Cartesian equation:

$$2\,x^2 + 2\,x\,z + z^2 - 3\,x - 2\,z + 1 = 0.$$

Next, if we substitute $y = 1$ in the implicit formula $f = 0$ of Steiner surface (see Figure 1.6), then we obtain

$$(2\,x^2 - 2\,x\,z + z^2 - x)\,(2\,x^2 + 2\,x\,z + z^2 - 3\,x - 2\,z + 1) = 0$$

yielding two different ellipses E_1 and E_2, see Figure (1.7).

Thus the parametrization $\boldsymbol{r}$ does not cover the ellipse E_1 related to the first factor, see Figure 1.8. In fact, more advanced calculations can show that the points that are missed by the parametrization $\boldsymbol{r}$ are exactly those points belonging to E_1 and not to E_2.

Deciding whether a parametrization is surjective, or equivalently, determining its missing points are questions that can be phrased in terms of topological closures, and thus in terms of limit computations. To see this on our example, we associate the map $\boldsymbol{r}$ with the polynomial

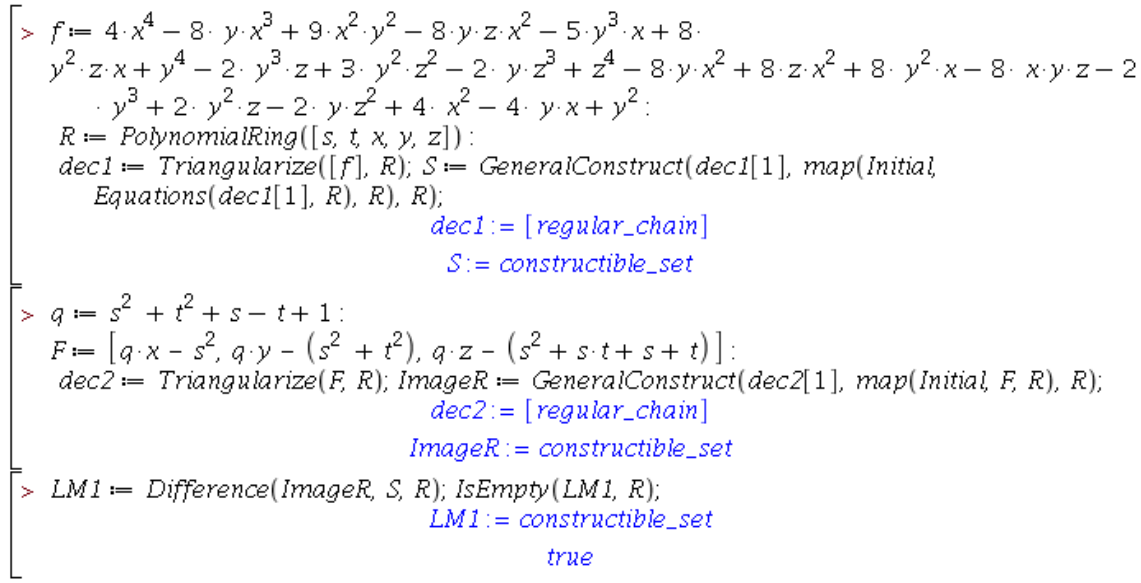


Figure 1.4: The command `Difference` computes the points in the image of $\boldsymbol{r}$ that do not belong to surface S, which is empty.

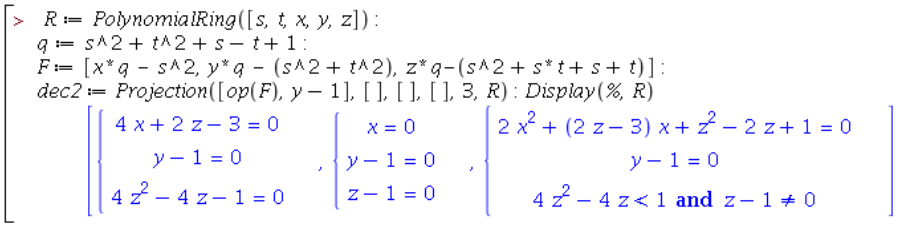


Figure 1.5: The intersection of the image of the parametrization $\boldsymbol{r}$ with plane $y = 1$.

system R shown in Equation (1.3)

$$R := \begin{cases} q(s,t)\,x - s^2 = 0 \\ q(s,t)\,y - (s^2 + t^2) = 0 \\ q(s,t)\,z - (s^2 + s\,t + s + t) = 0 \\ q(s,t) \neq 0 \end{cases} \tag{1.3}$$

Broadly speaking, Theorem 2 in [29] implies that eliminating s, t from the polynomial system R yields a polynomial g such that the topological closure of Image($\boldsymbol{r}$) is exactly the zero set of g. The Maple session of Figure 1.9 shows that this polynomial g is precisely the polynomial f introduced in Equation (1.1).

Therefore, our original question, that is, checking whether Image($\boldsymbol{r}$) $\subseteq S$ holds or not, can be answered by

1. computing the topological closure of Image($\boldsymbol{r}$), and
2. comparing it against S.

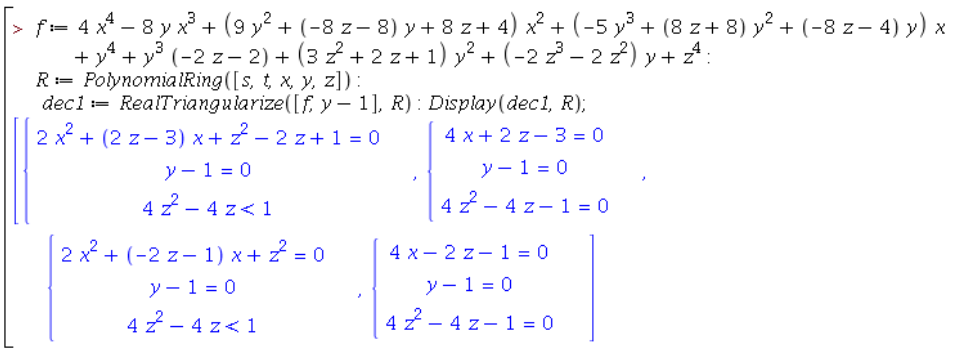


Figure 1.6: The intersection of Steiner surface S with plane $y = 1$.

Note that comparing the topological closure of Image($\boldsymbol{r}$) against Image($\boldsymbol{r}$) will produce the points missed by the parametrization r.

Computing topological closures of solution sets of polynomial systems, and thus computing missing points of parametrizations, are the questions motivating this thesis.

To see how those latter questions can be addressed in terms of limit computations, we introduce some notations before considering two examples. Let W be the zero set of a polynomial system R and let $\overline{W}$ be the topological closure of W in the Euclidean topology. Often W is known while $\overline{W}$ is to be computed. In the previous example, the set W was Image($\boldsymbol{r}$). In the next, and simpler example, we shall see that the set-theoretic difference $\overline{W} \setminus W$ can be obtained via a limit computation process. Consider the following system R

$$R := \begin{cases} z\,x - y^2 = 0 \\ y^5 - z^4 = 0 \\ z \neq 0 \end{cases} . \tag{1.4}$$

Let us denote by W the set of all $(x, y, z) \in \mathbb{C}^3$ solving R. A natural parametrization[3] of W can be given by

$$\begin{cases} x = \frac{y^2}{z} \\ y = z^{4/5} \end{cases} ,$$

where z now can play the role of a parameter. To be more formal, we can write this parametrization as

$$\begin{cases} x = \frac{t^{8/5}}{t} \\ y = t^{4/5} \\ z = t \end{cases} ,$$

where t is a parameter. For this parametrization, z (and consequently t) can accept any complex value but zero in order to be able to determine the values of x and y.

To compute $\overline{W} \setminus W$, that is, the possible missing points of that parametrization, we compute the limit of x, y, z, regarded as functions of t, at $t = 0$. Thus we have:

[3]Such parametrization will be defined formally later with the notion of Puiseux Series.

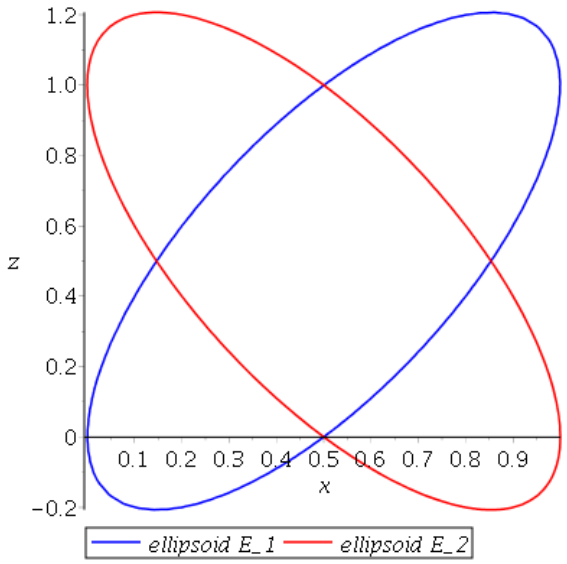


Figure 1.7: The intersection of Steiner surface S with plane $y = 1$.

$$\begin{aligned}
&\lim_{t\to 0} z = \lim_{t\to 0} t = 0,\\
&\lim_{t\to 0} y = \lim_{t\to 0} t^{4/5} = 0,\\
&\lim_{t\to 0} x = \lim_{t\to 0} \frac{y^2}{t} = \lim_{t\to 0} \frac{(t^{4/5})^2}{t} = \lim_{t\to 0} t^{\frac{3}{5}} = 0.
\end{aligned}$$

Therefore, $(0, 0, 0)$ is a missing point, or in other words, a *limit point* of the set W in the Euclidean topology.

Let us consider now this other system, which is a variation of (1.4) in which the term z^4 in the second equation is replaced with z^2:

$$\begin{cases} z\,x - y^2 = 0 \\ y^5 - z^2 = 0 \\ z \neq 0 \end{cases}. \tag{1.5}$$

The following is a parametrization of the above system:

$$\begin{cases} x = \frac{t^{4/5}}{t} \\ y = t^{2/5} \\ z = t \end{cases}.$$

Therefore,

$$\begin{aligned}
&\lim_{t\to 0} z = \lim_{t\to 0} t = 0,\\
&\lim_{t\to 0} y = \lim_{x_1\to 0} t^{2/5} = 0,\\
&\lim_{t\to 0} x = \lim_{t\to 0} \frac{y^2}{t} = \lim_{t\to 0} \frac{(t^{2/5})^2}{t} = \lim_{t\to 0} \frac{1}{t^{\frac{1}{5}}} = \pm\infty.
\end{aligned}$$

Since there is no finite limit for x at $t = 0$, there is no limit point for the system of Equation (1.5) in $\mathbb{R}^3$.

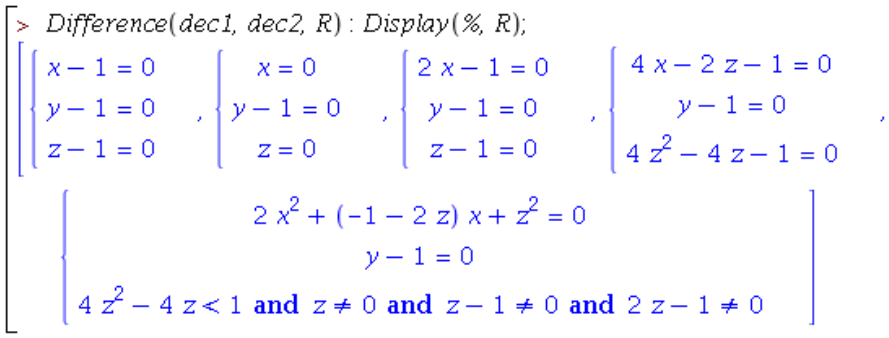


Figure 1.8: The points on Steiner surface S and the plane $y = 1$ which do not belong to the intersection of the image of the parametrization $\boldsymbol{r}$ given by (1.2) and the plane $y = 1$.

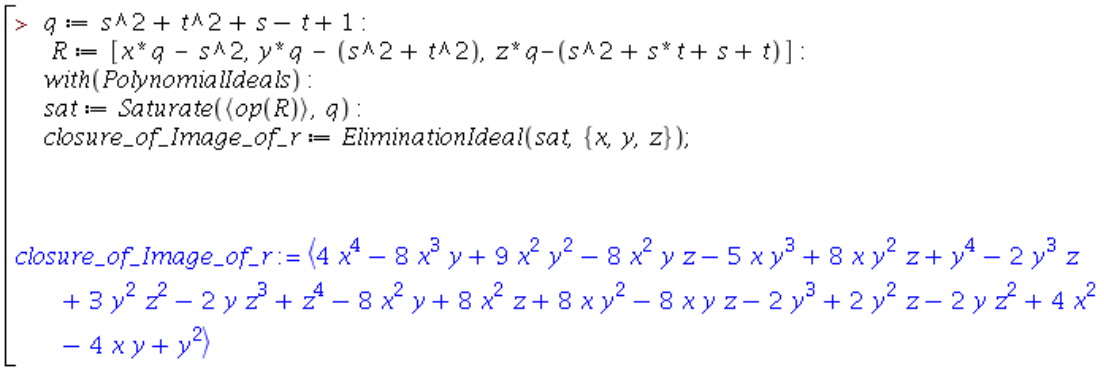


Figure 1.9: Closure of Image($\boldsymbol{r}$).

The above approach, which is based on computing the limits of different variables, is an inspiring method. However, there is a difficulty in applying this approach to more advanced cases. To see this, let us have a look at a slightly more complicated example and consider the solution set W of the system R in the variables t, y, x, given by:

$$R := \begin{cases} (t+2)\,t\,x^2 + (y+1)\,(x+1) = 0 \\ t\,y^5 - y + 1 = 0 \\ (t+2)\,t \neq 0 \end{cases}, \tag{1.6}$$

where the variable t is regarded as a parameter. In order to apply the same process as before, one needs to compute x and y as functions in t. For the last two examples, this was really easy. But now the obstacle is how to find such functions in general.

In this thesis, we aim at answering the question of how to compute both *real* and *complex* limit points of the solution sets of polynomial systems. The computation of missing points of rational parametrizations is one application, as we have seen above. Other applications, discussed in this thesis, are:

- the computation of the tangent cone of a space curve at one of its singular points, and
- the computation of limits of multivariate rational functions.

In the rest of this chapter, the main goals of this thesis are presented followed by our accomplishments towards those goals. We end this chapter by giving a brief overview of all chapters presented in this thesis.

1.1 Goals

In this thesis, we have three main objectives that we explain in this section.

Our first and most important goal is to compute the topological closures, or equivalently *limit points*, of solution sets of polynomial systems. The examples that we used above suggest to use parametric representations of such sets. To be technically more precise, they suggest to use representations given by rational functions. Unfortunately, this is not always possible and one needs to use a weaker notion of "parametrization", which is given by that of a *regular chain*[4].

Using the fact that the solution set of every polynomial system (over the complex number as well as over the reals) can be decomposed into the solution sets of finitely many regular chains, we restate our main objective as follows, where all technical terms are defined in Chapter 2.

Given a regular chain $R \subset \mathbb{Q}[x_1, \ldots, x_n]$, denoting by h_R the product of its initials, our goal is to compute the (non-trivial) *limit points* of both

1. the set $Z_{\mathbb{R}}(R)$ consisting of the real zeros of R which do not cancel h_R, and
2. the set $W(R)$ consisting of the complex zeros of R which do not cancel h_R.

In other words, denoting by $\overline{Z_{\mathbb{R}}(R)}$ and $\overline{W(R)}$ the closures of $Z_{\mathbb{R}}(R)$ and $W(R)$ in the Euclidean topology and Zariski topology, respectively, we want to determine the set $\lim(Z_{\mathbb{R}}(R)) := \overline{Z_{\mathbb{R}}(R)} \setminus Z_{\mathbb{R}}(R)$ and $\lim(W(R)) := \overline{W(R)} \setminus W(R)$.

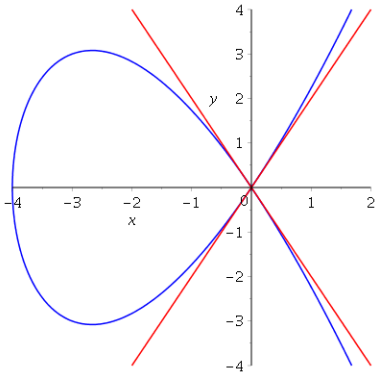


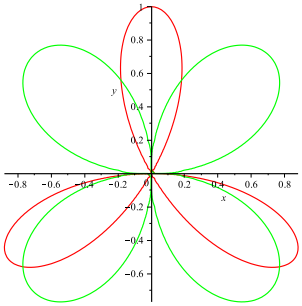
Figure 1.10: The tangent cone of the "fish" given by $f := y^2 - x^2(x+4) = 0$ at the origin consists of two tangent lines: $y = 2x$ and $y = -2x$.

Our second objective is to compute the tangent cone[5] of a space curve at one of its singular

[4]The notion of regular chain will be defined formally in Chapter 2.

[5]The tangent cone of a space curve at one of its points is a linear approximation of that curve around that point, see Figure 1.10 for an illustration.

$$> F := \left[(x^2+y^2)^2 + 3x^2y - y^3, (x^2+y^2)^3 - 4x^2y^2\right] :$$
$$> \text{plots[implicitplot]}(Fs, x = -2..2, y = -2..2) :$$



$$> R := \text{PolynomialRing}([x, y], 101) :$$
$$> \text{TriangularizeWithMultiplicity}(F, R);$$

$$\left[\left[1, \left\{ \begin{array}{c} x-1=0 \\ y+14=0 \end{array} \right\}\right], \left[\left[1, \left\{ \begin{array}{c} x+1=0 \\ y+14=0 \end{array} \right\}\right], \left[\left[1, \left\{ \begin{array}{c} x-47=0 \\ y-14=0 \end{array} \right\}\right], \left[\left[1, \left\{ \begin{array}{c} x+47=0 \\ y-14=0 \end{array} \right\}\right], \left[\left[14, \left\{ \begin{array}{c} x=0 \\ y=0 \end{array} \right\}\right]\right]\right]\right]\right]$$

Figure 1.11: In the `RegularChains` library in Maple, the command TriangularizeWithMultiplicity computes the intersection multiplicity of $V(F)$ for all the points $p \in V(F)$. In the above Maple session, computations are performed modulo a prime number for the only reason of keeping output expressions small. The same calculations can be performed with the TriangularizeWithMultiplicity command over the reals.

points without relying on standard basis computations. This topic is a follow-up on a preliminary work by S. Marcus, M. Moreno Maza and P. Vrbik on computing intersection multiplicities of zero-dimensional algebraic sets by means of regular chain theory [62]. Figure 1.11 illustrates how the `RegularChains` library in Maple computes triangular decomposition of zero-dimensional algebraic sets together with intersection multiplicities. In [62], the authors reduce their problem to that of computing the tangent cone of a space curve at one of its singular points. Therefore, the second objective of this thesis aims at completing the project initiated in [62].

Our third and final objective deals with the computations of limits of real multivariate rational functions. For example, for the multivariate rational function $q = \frac{x^4+3x^2y-x^2-y^2}{x^2+y^2}$, we are interested in determining whether $\lim_{(x,y)\to(0,0)} q$ exists or not, and, if it exists, to compute it, see Figure 1.12.

1.2 Thesis accomplishments

In this section, we would like to explain which achievements were obtained towards the objectives presented in Section 1.1.

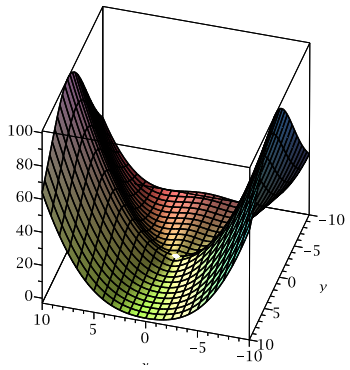


Figure 1.12: The surface defined by $q = z$ around the origin.

1.2.1 Computing limit points of quasi-components of regular chains of dimension one.

Regarding the first objective, that is, computing the limit points of solution sets of polynomial systems, we have established a method for computing limit points of solution sets of regular chains in dimension one, see Chapter 4 and the article [6] co-authored with C. Chen and M. Moreno Maza. Puiseux expansions play a central role in our results. The work presented in Chapter 4 focuses on limit points corresponding to the regular chains of dimension one with respect to Zariski topology.

1.2.2 Improving the extended Hensel construction.

For computing Puiseux series expansions, we were initially relying on the puiseux command of the algcurve package of MAPLE. This command only accepts input polynomials with two variables and, consequently, cannot be used for computing Puiseux parametrizations of regular chains with dimension higher than one. Therefore, a complete implementation computing Puiseux expansions of algebraic functions in several variables was needed. Up to our knowledge, there is only one practical method meeting our needs, namely the so-called *extended Hensel construction* (EHC, for short) a method introduced by T. Sasaki and F. Kako in [82]. Our current implementation of the EHC is integrated into the PowerSeries[6] package of MAPLE. In Chapter 3, a short review of the EHC is followed by our new techniques for improving the computational efficiency of that algorithm. We show that the EHC requires only linear algebra and univariate polynomial arithmetic. We deduce complexity estimates and report on a software implementation together with experimental results. This is a joint work with Masoud Ataei and Marc Moreno Maza.

[6]This package can be downloaded from `http://www.regularchains.org/downloads.html`

1.2.3 Computing the real limit points of the quasi-component of a regular chain of dimension one.

One of the interesting facts about the EHC is the following. Suppose that, for a bivariate polynomial $F \in K[X, Y]$, where K is a field extension of $\mathbb{Q}$ (the field of rational numbers) we want to factor F into linear factors in X over the field of Puiseux series of Y. Then, the EHC will not only determine the algebraic extension L of K necessary to do so, but it will also, for each linear factor of F, characterize the sub-field of L which is required to express the coefficients of that factor. Thanks to this fact, we have proposed a method for computing the "*real*" limit points of regular chains of dimension one in Chapter 5; see also Section 5 in [3].

1.2.4 Studying regular chains under changes of coordinates.

As mentioned above, the ideas presented in Chapters 4 and 5 (respectively for computing the complex and real limit points of the quasi-components of regular chains) are only applicable to regular chains with dimension one. Therefore, a complete algorithm is still needed to compute the limit points of quasi-components of regular chains in higher dimension. That is why we take the second approach presented in Chapter 6 and [4]. Broadly speaking, the intention is to map the limit point computation from a coordinate system where it is difficult to perform to another coordinate system where it is easy to do. We do not propose a general method achieving this intention, but we propose criteria which appear to be helpful in practice.

1.2.5 Introducing new tools for computing tangent cones of space curves.

Computing limit points of quasi-components of regular chains can be applied to perform other "*limit computations*". The first example, discussed in Chapter 7, is the computation of the tangent cone of a space curve C (and more generally algebraic set) at one of its points, say P. Indeed, the tangent cone of C at P is the set of the limits of the secants PQ where Q is a point on C approaching P. Taking advantage of this method, in Chapter 7, we have proposed a method for computing the tangent cone of algebraic curves, which is the main tool for computing the *intersection multiplicity* of algebraic curves at any point. This means we have also met our second objective of this proposal for computing the tangent cones of space curves at their singular points.

1.2.6 Computing limit of real multivariate rational functions.

Regarding the last objective of this dissertation, we have answered the questions of

1. how to determine whether the limit of a real multivariate rational function at the origin exists or not, and
2. if it exists, how to compute it,

provided that the origin is an isolated zero of the denominator. Our proposed method reduces to computing the real limit points of regular chains of dimension one, which in turn demonstrates another application of the methods presented in Chapters 4 and 5. Our proposed technique for computing the limit of real multivariate rational functions at the origin is explained in Chapter 8 and also [8]; these techniques extend and enhance the papers [21] and [98] for

respectively computing the limit of real bivariate and trivariate rational functions at the origin when the origin is an isolated zero of the denominator. The key point in this method is that for computing the limit of a multivariate rational function q at the origin, it is enough to compute the limit of q at the origin along a finite number of special paths, represented by a so-called *discriminant variety*. For example, for the real multivariate rational function $q := \frac{x^4+3\,x^2\,y-x^2-y^2}{x^2+y^2}$, see Figure 1.13, its discriminant variety consists of three different paths through the origin. Following each of them, one can verify that the limit of q at the origin exists and is equal to -1.

For covering the case when the origin is not an isolated solution of the denominator, we suggest possible directions, in particular by following the paper [54] in which the L'Hospital's rule for multivariate rational functions is introduced. It is worth noting that in 2014, S.J. Xiao and G.X. Zeng, in [104], proposed a first algorithm that decides whether $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q$ is zero or not, without any assumptions on the denominator of q. However, the results of our experimentation shows that our implementation outperforms the one in [104].

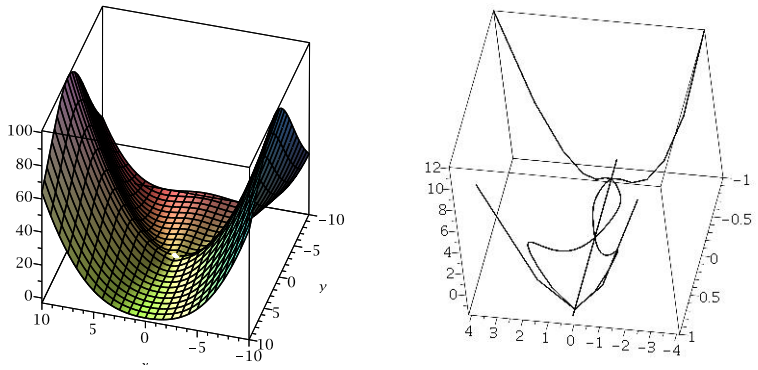
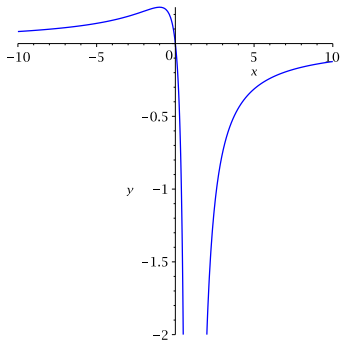


Figure 1.13: On the left: the surface defined by $q := \frac{x^4+3\,x^2\,y-x^2-y^2}{x^2+y^2} = z$ around the origin. On the right: the three paths of discriminant variety of q going through the point (0,0,-1).

1.2.7 Separating the real and complex branches of space curves.

For computing the limit of a real multivariate rational function q at the origin, one needs to compute local parametrizations of all the paths represented by the discriminant variety (in the sense of the papers [21] and [98]) of q around the origin. In the set up of the method described in Chapter 8 and also [8], we represent the discriminant variety of q as the union of finitely many *regular semi-algebraic sets*. Thus for computing and separating all of those special real paths, we compute real Puiseux parametrizations corresponding to all of the regular chain parts of the regular semi-algebraic sets representing the discriminant variety of q at the origin, see Figure1.13. In fact, we propose a method for separating the real and complex branches of space curves given by regular chains from the ones that escape to infinity when the parameter corresponding to the given curve approaches zero, see Figure 1.14.

The command `RegularChainBranches` which is integrated in RegularChains library, implements this method for an input regular chain with dimension one. Note that the input regular




```
> R:= PolynomialRing([x,y]):
  rc:= Chain([f],Empty(R),R):
  RegularChainBranches(rc,R, [y],[5]);
     [[y = T, x = -4 T (112 T^3 + 20 T^2 + 4 T + 1)]]
```

Figure 1.14: On the right: Close to the origin, the irreducible polynomial $f := y(x-2)^2 + x$ contains two different paths: one passing through the origin and the other one not. On the left: RegularChainBranches returns a parametrization for the path of f going through the origin.

chain might have more branches than the ones printed in the output of `RegularChainBranches`. In fact, those branches corresponding to the given regular chain that escape to infinity, when the parameter of the given regular chain approaches zero, are excluded from the output of the command `RegularChainBranches` (see Figure 1.15).

```
> R := PolynomialRing([x, y, z]) :
  rc := Chain([y^(3)-2* y^(3) + y^(2) + z^(5), z^(4)* x + y^(3)-y^(2)], Empty(R), R) : Display(rc, R);
  br := RegularChainBranches(rc, R, [z], coefficient = complex);
```

$$\begin{cases} z^4 x + y^3 - y^2 = 0 \\ -y^3 + y^2 + z^5 = 0 \\ z^4 \neq 0 \end{cases}$$

$$br := \left[\left[z = T^2, y = \frac{1}{2} T^5 \left(-T^5 + 2\, RootOf(\_Z^2 + 1)\right), x = -\frac{1}{8} T^2 \left(-T^{20} + 6\, T^{15}\, RootOf(\_Z^2 + 1) + 10\, T^{10} + 8\right)\right], \left[z = T^2, y = -\frac{1}{2} T^5 \left(T^5 + 2\, RootOf(\_Z^2 + 1)\right), x = \frac{1}{8} T^2 \left(T^{20} + 6\, T^{15}\, RootOf(\_Z^2 + 1) - 10\, T^{10} - 8\right)\right], [z = T, y = T^5 + 1, x = -T \left(T^{10} + 2\, T^5 + 1\right)]\right]$$

```
> br := RegularChainBranches(rc, R, [z], coefficient = real);
```

$$br := [[z = T, y = T^5 + 1, x = -T \left(T^{10} + 2\, T^5 + 1\right)]]$$

Figure 1.15: The command `RegularChainBranches` computes a parametrization for the complex and real paths of the quasi-component defined by *rc*. When coefficient argument is set as real, then the command `RegularChainBranches` computes the real branches.

1.2.8 Thesis contribution in `RegularChains` and `PowerSeries` libraries.

The recent progress on regular chain theory and all the new features added to `RegularChains` library, described in this thesis, are presented in [5] and in [7] with a focus on limit point computations. All of these features are integrated to `AlgebraicGeometryTools` package of the `RegularChains` library. Furthermore, the `RegularChains` library has a new companion, called the `PowerSeries` library, to support computations with power series and Puiseux series. The command ExtendedHenselConstruction, which computes the Puiseux expansions of an

input multivariate polynomial with respect to its main variable, is integrated into `PowerSeries` library. The source code of both the `RegularChains` and `PowerSeries` libraries are available at `http://www.regularchains.org/`.

1.3 Contribution statement

Each chapter of the present thesis, except Chapter 2, is based on a refereed publication. In the sequel, we discuss the relations between those publications and chapters, as well as co-authors' contributions.

The work presented in Chapter 3, about the extended Hensel construction, is accepted at the ISSAC 2017 conference [3]. This is a joint work with Marc Moreno Maza and Masoud Ataei.

The method presented in Chapter 4, for computing complex limit points of regular chains of dimension one with respect to Zariski topology, is a joint work with Marc Moreno Maza and Changbo Chen. It was published in 2013 in the CASC conference [6].

The method for finding the real limit points corresponding to the regular chains of dimension one, which forms Chapter 5, is mainly the work of the author of the present thesis; it is also presented in Section 5 of [3].

The materials presented in Chapter 6, except Section 6.3.1, were published in 2015 in the CASC conference [4]; this is a joint work with Marc Moreno Maza, Amir Hashemi, and Changbo Chen. Section 6.3.1 in Chapter 6 is about the PALGIE algorithm, which was originally introduced in [18, 20] by François Boulier, François Lemaire and Marc Moreno Maza for differential regular chains. We present this algorithm in a purely algebraic setting and extensions of his specifications are proposed; this work was done by Marc Moreno Maza and the author of the present thesis.

The work presented in Chapter 7, based on the CASC 2015 article [9] by the author of the present thesis with Marc Moreno Maza, Éric Schost, and Paul Vrbik. In fact, this work builds upon the PhD thesis of Paul Vrbik, see [100] and is an application of computing limit of quasi-components of regular chains.

Finally, the materials forming Chapter 8 are mainly taken from the ISSAC 2016 article [8]. This is a joint work of the author of the present paper with Marc Moreno Maza and Mahsa Kazemi.

1.4 Thesis outline

This thesis contains seven different chapters, in addition to the current one.

Chapter 2 provides definitions and notations used throughout this thesis about our main tools, namely regular chains and Puiseux series expansions. It also provides some information about the related work.

Chapter 3 is devoted to the extended Hensel construction (EHC) and its implementation. In Section 3.2, we give a brief review of the EHC for bivariate and multivariate polynomials. For computing Puiseux expansions of bivariate polynomials using EHC, there are two main steps. The first one is to compute the so-called *Yun-Moses polynomials* corresponding to the

initial factors of the Newton polynomial. To do so, we suggest an efficient algorithm together with complexity estimates in Section 3.3. The second step is to compute the product of the lifted factors at each iteration. In Section 3.4, we explain our method for multiplying the lifted factors, efficiently; in particular, we demonstrate how to recycle the computations performed in previous iterations. Finally, in Section 8.7, we present experimental results comparing our improved EHC against the original one.

In Chapter 4, we address the problem of computing (non-trivial) limit points induced by regular chains of dimension one with respect to Zariski topology. The method described in Chapter 4 for computing such limits is via Puiseux series expansions. We first establish the notion of Puiseux parametrizations for regular chains of dimension one around a point vanishing the product of the initials of the polynomials contained in the given regular chain. By using a few of the first terms of such Puiseux series appearing in the Puiseux parametrizations of a regular chain, one can compute the desired limit points. Later on, we give several results on how to truncate the series appearing in the Puiseux parametrizations while having sufficiently many terms to calculate the limit points. We conclude Chapter 4 by an application of finding limit points corresponding to regular chains in removing redundant components in triangular decompositions of polynomial systems.

In Chapter 6, we discuss how to compute limit points of quasi-components of regular chains via changes of coordinates. The results in this chapter cover the case where regular chains have positive dimension and not necessarily dimension one. One of the main challenges in this method is that after applying a change of coordinates to a regular chain, the resulted polynomial set might not be a regular chain. Our goal is, then, to replace this polynomial set by one regular chain at a "cheap computational cost. We achieve this by employing the PALGIE algorithm [18, 20] We also consider Noether normalization and its effect on regular chains structure. Finally, we present several results for computing the limit points of quasi-components of the regular chains with respect to Zariski topology.

Chapter 7 describes our method for computing the tangent cones of a space curve at one of its singular points. As explained in Section 1.1, the main idea is to compute limits of families of secant lines; this trick permits a reduction to the computations of limit points of quasi-components of regular chains in dimension one.

In Chapter 8, we discuss how to compute the limit of real multivariate rational functions at the origin. We first present our method for computing such limits when this latter point is an isolated zero of the denominator. This work was published in [8]. Section 8.3 states fundamental lemmas as the main ingredients to the proof of the correctness of Algorithm 20, in Section 8.4. Then, in Section 8.6, we suggest how to compute the limits of real multivariate rational functions at the origin when this latter point is not an isolated zero of the denominator.

Finally, Chapter 9 summarizes the accomplishments of the present thesis for addressing the main goals of this thesis. Chapter 9 also discusses the problems that have remained unsolved with respect to our goals.

All the algorithms presented in this thesis are implemented in MAPLE and are integrated into `RegularChains` and `PowerSeries` libraries.

Chapter 2

Background and Related Work

In this chapter, we gather definitions and notations used throughout this thesis. The problem of computing limit points of regular chains is also explained more formally. Moreover, related works are also discussed.

2.1 Solving polynomial systems

Solving linear polynomial systems is a cornerstone in mathematical sciences. This problem has been at the centre of attention of many researchers in both the academia and the industry, yielding fast algorithms and efficient implementation. However, replacing linear systems with nonlinear ones results in a dramatic decline in what can be achieved in practice, due to the inherent higher complexity of the problem. This is especially true when exactness and completeness are required features for the computed solution sets. Therefore, it is desirable for symbolic methods, which aim at providing these features, to emphasize the development of better and better algorithms for solving polynomial systems.

Among all the avenues by which such systems can be solved, the one significant method is *triangular decomposition*, see [19, 25, 22, 11], which decomposes each nonlinear system into several components, with special shapes and rich properties, called *regular chains*.

Before stating formal definitions, we start with an example. Consider the polynomial set $F := \{f_1, f_2, f_3\}$ where the polynomials

$$f_1 = x^2 + y + z - 10, f_2 = y^2 + x + z - 1, f_3 = z^2 + x + y - 1$$

have rational number coefficients and three variables z, y, x, that we order as $z < y < x$. The solution set of F, that is, the set of the points (x, y, z) with complex coordinates satisfying

$$f_1 = f_2 = f_3 = 0,$$

can be decomposed as follows into four components:

$$R_1 := \begin{cases} x - z = 0 \\ y - z = 0 \\ z^2 + 2z - 1 = 0 \end{cases}, \quad R_2 := \begin{cases} x = 0 \\ y = 0 \\ z - 1 = 0 \end{cases}, \quad R_3 := \begin{cases} x = 0 \\ y - 1 = 0 \\ z = 0 \end{cases}, \quad R_4 := \begin{cases} x - 1 = 0 \\ y = 0 \\ z = 0 \end{cases}.$$

Each of the polynomial sets defining those components has a triangular shape and is called a *regular chains*. In our example, we denote these four regular chains by R_1, R_2, R_3, R_4 respectively and we say that $\{R_1, R_2, R_3, R_4\}$ forms a triangular decomposition for F. As one can see, it is easy to read the solutions for each of the regular chains R_1, R_2, R_3, R_4, even if, for R_1, a bit of work is needed (solving the equation in z and substituting in the other two equations).

Let $X_1, \ldots, X_n$ be n independent variables. A *monomial* in variables $X_1, \ldots, X_n$ is of the form $X_1^{\alpha_1} \cdots X_n^{\alpha_n}$, where $\alpha_i \in \mathbb{Z}_{\geq 0}$, for $i = 1, \ldots, n$. For convenience, we denote a monomial as X^α. The degree of this monomial is defined as $\alpha_1 + \cdots + \alpha_n$ and denoted by $\deg(X^\alpha)$.

The set all of the polynomials in $X_1, \ldots, X_n$ with coefficients in the field $\mathbf{k}$ is denoted as $\mathbf{k}[X_1, \ldots, X_n]$, or $\mathbf{k}[X]$, and is called a *polynomial ring* over $\mathbf{k}$. In this thesis, we are mainly interested in polynomial rings over the real and complex numbers, that is, when $\mathbf{k}$ is the field of real or complex numbers.

Let $X_1 < \cdots < X_n$ be an order on these variables. For a non-constant polynomial f, the greatest variable appearing in f is called the *main variable* of f, denoted by $\mathrm{mvar}(f)$, and the leading coefficient of f w.r.t. $\mathrm{mvar}(f)$ is called the *initial* of f, denoted by $\mathrm{init}(f)$. These notations support algorithms where the polynomial f is considered as a univariate polynomial in its main variable.

Example 1. *For $f = 2\,X_1X_2^2 + X_2 \in \mathbb{Q}[X_1, X_2]$, its main variable is X_2 and its initial is $2\,X_1$.*

Let $\mathbf{k}$ be a field and $\overline{\mathbf{k}}$ be its algebraic closure[1]. Since in this thesis, the field $\mathbf{k}$ will often be the field $\mathbb{R}$ of the real numbers or the field $\mathbb{C}$ of the complex numbers, we have $\overline{\mathbf{k}} = \mathbb{C}$. For a polynomial set $F \subseteq \mathbf{k}[X]$, the set of all common solutions (or zeros) of the polynomials in F is denoted by $V(F)$ and called the *algebraic set*, or *algebraic variety*, of F. Hence, we have:

$$V(F) := \{(a_1, \ldots, a_n) \in \overline{\mathbf{k}}^n \mid f(a_1, \ldots, a_n) = 0 \text{ for each } f \in F\},$$

.

Example 2. *Let $F = \{X_1 + X_2^2, X_1^2 + X_2\} \subset \mathbb{C}[X_1, X_2]$. Then*

$$V(F) = \left\{(0,0), (-1,-1), \left(\frac{1+i\sqrt{3}}{2}, \frac{1-i\sqrt{3}}{2}\right), \left(\frac{1-i\sqrt{3}}{2}, \frac{1+i\sqrt{3}}{2}\right)\right\}.$$

It is worth mentioning that the solutions of a polynomial system depend on the field over which the solutions are searched for. For instance, in Example 2, if we solve F over $\mathbb{Q}$ then we have

$$V_{\mathbb{Q}}(F) = \{(0,0), (-1,-1)\}.$$

Definition 1. *A set R of non-constant polynomials in $\mathbf{k}[X]$ is said to be a* triangular set*, if for all $f, g \in R$, with $f \neq g$, we have* $\mathrm{mvar}(f) \neq \mathrm{mvar}(g)$. *A variable X_i is said* free *w.r.t. R if there is no $f \in R$ such that* $\mathrm{mvar}(f) = X_i$*, otherwise, it is said* algebraic.

Example 3. *Consider the set R defined as following:*

$$R = \left\{ \begin{array}{l} r_2 := X_1\,X_3 - X_2^2 \\ r_1 := X_2^4 - X_1^5 \end{array} \right. \subseteq \mathbb{Q}[X_1, X_2, X_3].$$

[1] See the wikipedia page `https://en.wikipedia.org/wiki/Algebraic_closure`, as of May 2017.

Then, R is a triangular set since $\mathrm{mvar}(r_2) = X_3 \neq \mathrm{mvar}(r_1) = X_2$. *Moreover, X_1 is the only free variable of this set.*

Definition 2. *A set $\mathcal{I} \subset \mathbf{k}[X_1, \ldots, X_n]$ is an ideal if it satisfies the following conditions:*

1. $0 \in \mathcal{I}$,
2. *if $f, g \in \mathcal{I}$, then $f + g \in \mathcal{I}$, and*
3. *if $f \in \mathcal{I}$ and $h \in \mathbf{k}[X_1, \ldots, X_n]$, then $hf \in \mathcal{I}$.*

Lemma 1 (See [29], Chapter 1, Lemma 3)**.** *Let $f_1, \ldots, f_s$ be polynomials in $\mathbf{k}[X_1, \ldots, X_n]$. Consider the set defined as follows:*

$$\langle f_1, \ldots, f_s \rangle := \left\{ \sum_{i=1}^{s} h_i f_i \mid h_1, \ldots, h_s \in \mathbf{k}[X_1, \ldots, X_n] \right\}.$$

Then, the set $\langle f_1, \ldots, f_s \rangle$ is an ideal and is called the ideal generated by $f_1, \ldots, f_s$.

Definition 3. *For a nonempty triangular set R, we define the saturated ideal of R, which is denoted by* $\mathrm{sat}(R)$*, to be the ideal*

$$\langle R \rangle : h_R^\infty := \{ f \in \mathbf{k}[X] \mid \exists m \in \mathbb{N} : h_R^m f \in \langle R \rangle \},$$

where h_R is the product of the initials of the polynomials in R. The saturated ideal of the empty triangular set is defined as the trivial ideal.

The ideal $\mathrm{sat}(R)$ has several properties, in particular it is unmixed (see [19]). We denote the number of polynomials in R by e. It turns out that $\mathrm{sat}(R)$ has dimension $n - e$ (see Theorem 1.6 in [19]). Moreover, writing $R = \{r_1, \ldots, r_e\}$ and assuming $\mathrm{mvar}(r_i) = X_{n-e+i}$, for $1 \leq i \leq e$, the intersection $\mathbf{k}[X_1, \ldots, X_{n-e}] \cap \mathrm{sat}(R)$ is the trivial ideal $\langle 0 \rangle$.

Definition 4. *Let $\mathcal{I} \subset \mathbf{k}[X_1, \ldots, X_n]$ be an ideal. A polynomial is regular modulo $\mathcal{I}$, if it is neither zero nor a zero-divisor*[2] *modulo $\mathcal{I}$.*

Definition 5. *We say that the triangular set $R = \{r_1, \ldots, r_e\}$ is a regular chain whenever R is empty or $\{r_1, \ldots, r_{e-1}\}$ is a regular chain and the initial of r_e is regular modulo the saturated ideal* $\mathrm{sat}(\{r_1, \ldots, r_{e-1}\})$*. The regular chain R is said to be* strongly normalized *whenever no algebraic variables appear in the initials of the polynomials of R,*

Example 4. *One more time consider the triangular set*

$$R := \begin{cases} X_1 X_3 - X_2^2 \\ X_2^4 - X_1^5 \end{cases}.$$

Then R is a regular chain because $h_R = X_1$ is regular modulo $\mathrm{sat}(\{r_1\}) = \langle X_2^4 - X_1^5 \rangle$*, where $r_1 := X_2^4 - X_1^5$. Indeed, the regularity of a polynomial modulo the saturated ideal of a regular chain can be checked using pseudo division (see [25]).*

[2] See the wikipedia page `https://en.wikipedia.org/wiki/Zero_divisor`, as of May 2017.

Definition 6. *We denote by* $W(R) := V(R) \setminus V(h_R)$ *the* quasi-component *of* R*, that is, the common zeros of* R *that do not cancel* h_R*.*

Definition 7. *Let* $S \subset \overline{\mathbf{k}}^n$*. The* Zariski closure *of the set* S*, denoted as* $\overline{S}$*, is defined to be* $V(\mathcal{I}(S))$ *where*

$$\mathcal{I}(S) = \{f \in \mathbf{k}[X_1, \ldots, X_n] \mid f(a) = 0 \text{ for all } a \in S\}.$$

It can be proved that $\overline{S}$ *is also the intersection of all algebraic sets containing* S*, that is, the smallest algebraic set containing* S*.*

Regular chains enjoy many properties. In particular, the quasi-component $W(R)$ of the regular chain is not empty. Moreover, the Zariski closure of $W(R)$ satisfies the following: $\overline{W(R)} = V(\mathrm{sat}(R))$.

Definition 8. *Let* $F \subset \mathbf{k}[X]$*. The regular chains* $R_1, \ldots, R_s$ *of* $\mathbf{k}[X]$ *form a* triangular decomposition *of* $V(F)$ *in the sense of Kalkbrener (resp. Wu and Lazard) whenever we have* $V(F) = \cup_{i=1}^{s} \overline{W(R_i)}$ *(resp.* $V(F) = \cup_{i=1}^{s} W(R_i)$*). We denote by* Triangularize *an algorithm, such as the one in [25], computing a Kalkbrener triangular decomposition.*

Regular chains are defined in multivariate polynomial rings where coefficients are themselves in a (commutative) ring. In practice, coefficients are often rational numbers while the values of the unknowns are either complex or real numbers. In the latter scenario, it is desirable to have an algorithm decomposing the solutions of polynomial systems over $\mathbb{R}$ rather than over $\mathbb{C}$. This is done by another algorithm called RealTriangularize introduced in [22] by C. Chen, J. H. Davenport, J. P. May, M. Moreno Maza, B. Xia and R. Xiao.

Example 5. *Let* $F := \{5\,y^6 - 15\,y^5 + 15\,y^4 + 2\,x\,z^2 - 5\,y^3 + 5\,x^2 + 5\,z^2\} \subset \mathbb{Q}[z < y < x]$*. The command* `RealTriangularize` *in the* `RegularChains` *library in* MAPLE *can solve for the real solutions of* F*, see Figure 2.2. These real solutions form the so-called* Eve *surface displayed on Figure 2.1*[3]*. On the other hand, the command* `Triangularize` *can solve the complex solutions of the above system as illustrated on Figure 2.3.*

Regular chains can be used to represent the solution sets of systems of equations, inequations and inequalities. More generally, they support the implementation of set-theoretic operations (union, intersection, difference) of algebraic sets, constructible sets, and semi-algebraic sets. To illustrate the difference between triangular decompositions in the sense of Kalkbrener, and triangular decompositions in the sense of Lazard and Wu (see Definition 8) we consider the following example.

Example 6. *For the variable order* $b < a < y < x$*, consider the polynomial set* $F = \{a\,x + b, b\,x + y\}$ *and the regular chain* $R_1 = \{b\,x + y, a\,y - b^2\}$*. Using calculations in* Maple*, one can verify that we have*

$$V(F) = \overline{W(R)}, \tag{2.1}$$

[3]This image is taken from the gallery of algebraic surfaces at `http://homepage.univie.ac.at/herwig.hauser/bildergalerie/gallery.html`

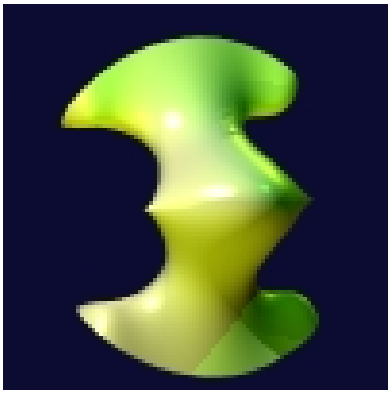


Figure 2.1: The eve surface

```
> R := PolynomialRing([x, y, z]) : F := [5·x^2 + 2·x·z^2 + 5·y^6 + 15·y^4 + 5·z^2 − 15·y^5 − 5·y^3] :
  RealTriangularize(F, R, output = record);
```

$$\left\{ 5x^2 + 2z^2x + 5y^6 - 15y^5 + 15y^4 - 5y^3 + 5z^2 = 0,\ 25y^6 - 75y^5 + 75y^4 - z^4 - 25y^3 + 25z^2 < 0 \right\},$$

$$\left\{ 5x + z^2 = 0,\ 25y^6 - 75y^5 + 75y^4 - 25y^3 - z^4 + 25z^2 = 0,\ 64z^4 - 1600z^2 + 25 > 0,\ z \neq 0,\ z - 5 \neq 0,\ z + 5 \neq 0 \right\},$$

$$\left\{ x = 0,\ y - 1 = 0,\ z = 0 \right\},\quad \left\{ x = 0,\ y = 0,\ z = 0 \right\},$$

$$\left\{ x + 5 = 0,\ y - 1 = 0,\ z - 5 = 0 \right\},\quad \left\{ x + 5 = 0,\ y = 0,\ z - 5 = 0 \right\},$$

$$\left\{ x + 5 = 0,\ y - 1 = 0,\ z + 5 = 0 \right\},\quad \left\{ x + 5 = 0,\ y = 0,\ z + 5 = 0 \right\},$$

$$\left\{ 5x + z^2 = 0,\ 2y - 1 = 0,\ 64z^4 - 1600z^2 + 25 = 0 \right\}$$

Figure 2.2: The real solutions of $V(F)$ computed by `RealTriangularize`

which means that $\{R_1\}$ *is a triangular decomposition of* F *in the sense of Kalkbrener. Using the notion of* localization *in algebra* [4]*, it is possible to interpret Equation (2.1) by stating that* R_1 *describes the solutions of* F *of the following form:*

$$\begin{cases} x = -\frac{y}{b} \\ y = \frac{b^2}{a} \end{cases},$$

for any complex value of a *and* b *where* $a\,b \neq 0$*. However, this latter solution set is missing the solutions of* F *given by the regular chains* $R_2 = \{x, y, b\}$ *and* $R_3 = \{y, a, b\}$*. In fact, one can easily verify, with* Maple *calculations, that the following holds:*

$$V(F) = (V(R_1) \setminus V(a\,b)) \cup V(R_2) \cup V(R_3),$$

which means that $\{R_1, R_2, R_3\}$ *is a triangular decomposition of* F *in the sense of Lazard and Wu.*

In Example 8, one can see that we have

$$\overline{W(R_1)} \setminus W(R_1) = V(R_2) \cup V(R_3),$$

that is, the limit points of $W(R_1)$ in Zariski topology consist of $W(R_1)$ and the two lines given by $V(R_2)$ and $V(R_3)$. At this point, we review the notion of a limit point.

[4] See the wikipedia page `https://en.wikipedia.org/wiki/Localization_(algebra)`, as of May 2017.

```
> R := PolynomialRing([x, y, z]) : F := [5·x^2 + 2·x·z^2 + 5·y^6 + 15·y^4 + 5·z^2 − 15·y^5 − 5·y^3] :
  dec := Triangularize(F, R) : Display(dec, R);
                              [5 x^2 + 2 z^2 x + 5 y^6 − 15 y^5 + 15 y^4 − 5 y^3 + 5 z^2 = 0]
```

Figure 2.3: The complex solutions of $V(F)$ computed by `Triangularize`

2.1.1 Limit points

Let (X, τ) be a topological space. A point $p \in X$ is a *limit* of a sequence $(x_n, n \in \mathbb{N})$ of points of X if, for every neighbourhood U of p, there exists an N such that, for every $n \geq N$, we have $x_n \in U$; when this holds we write $\lim_{n\to\infty} x_n = p$. If X is a Hausdorff space then limits of sequences are unique, when they exist. Let $S \subseteq X$ be a subset. A point $p \in X$ is a *limit point* of S if every neighbourhood of p contains at least one point of S different from p itself. Equivalently, p is a limit point of S if it is in the closure of $S \setminus \{p\}$. In addition, the closure of S is equal to the union of S and the set of its limit points. If the space X is sequential, and in particular if X is a metric space, the point p is a limit point of S if and only if there exists a sequence $(x_n, n \in \mathbb{N})$ of points of $S \setminus \{p\}$ with p as limit. In practice, the "interesting" limit points of S are those which do not belong to S. For this reason, we call such limit points *non-trivial* and we denote by $\lim(S)$ the set of non-trivial limit points of S.

Definition 9. *For the regular chain $R \subset \mathbf{k}[X_1, \ldots, X_n]$ a point $p \in \overline{\mathbf{k}}^n$ is called a* limit point *of R, if it is a limit point of $W(R)$ in the above topological sense. Further, the point p is said a* non-trivial *limit point of R if it is a non-trivial limit point of $W(R)$. The set of all non-trivial limit points of $W(R)$ is denoted by* $\lim(W(R))$. *From now on, when we talk about the limit points of a quasi-component of a regular chain, we actually refer to its non-trivial limit points.*

2.2 Power series and Puiseux expansions

In mathematics, Puiseux series are a generalization of power series. As we know, power series are used to approximate an analytic function with a polynomial about a point. As an example, $\frac{1}{1-x} = \sum_{i=0}^{\infty} x^i = 1 + x + x^2 + x^3 + \cdots$ is the power series expansion of the function $\frac{1}{1-x}$ about $x = 0$. Now one might ask the question of how to approximate a plane algebraic curve about one of its points, in particular singular points. In this case, such curve may not be seen as the graph of an analytic function, even locally. Puiseux series address this more general setting. A *Puiseux series*[5] in x typically looks like

$$\sum_{i=k}^{\infty} c_i x^{\frac{i}{d}},$$

where d is a positive integer and k is an integer, possibly negative.

(Univariate) Puiseux series in X_1, with complex number coefficients, form an algebraically closed field, denoted by $\mathbb{C}((X_1^*))$. Consequently, any bivariate polynomial $F \in \mathbb{C}[X_1, X_2]$ can be factored into linear factors in X_2 with coefficients in $\mathbb{C}((X_1^*))$. This result, known as Puiseux's

[5] See the wikipedia page `https://en.wikipedia.org/wiki/Puiseux_series`, as of May 2017.

theorem, has a geometrical interpretation. These linear factors correspond to the *branches* of the plane curve[6] $F(X_1, X_2) = 0$ around the origin. Being able to compute the branches of a plane curve around one of its points will be an essential tool in Chapter 4, upon which we will build our algorithm for computing the limit points of regular chains with one free variable.

Example 7. *By factoring the polynomial* $F := X_1X_2^3 + X_2^2 + X_2 + X_1$, *over* $\mathbb{C}((X_1^*))$, *we obtain the following three linear factors:*

- $X_2 + 1$,
- $X_2 + X_1 + X_1^2 + O(X_1^3)$,
- $X_2 + \frac{1}{X_1} - 1 - X_1 - X_1^2 + O(X_1^3)$,

and thus a description of the branches of $F(X_1, X_2) = 0$ *around the origin:*

- $X_2 = -1$,
- $X_2 = -X_1 - X_1^2 + O(X_1^3)$,
- $X_2 = -\frac{1}{X_1} + 1 + X_1 + X_1^2 + O(X_1^3)$,

Since Puiseux expansions may not have finitely many terms, the big-oh notation is used to indicate that higher-order terms are not displayed.

```
> PS := PowerSeries([X, Y]):
  with(PS):
  UPoPS := UnivariatePolynomialOverPowerSeries([X, Y], Z):
  with(UPoPS):
  u := UPoPS-FromListOfPolynomials([Y, 1, X+1]);
  UPoPS-PolynomialPart(u, 2);
  (p, alpha) := UPoPS-WeierstrassPreparation(u, 2);
  UPoPS-PolynomialPart(p, 2);
  UPoPS-PolynomialPart(alpha, 2);
```

$$u := \text{polynomial\_over\_power\_series}$$

$$Y + Z + (X+1)\,Z^2$$

$$p,\ \alpha := \text{polynomial\_over\_power\_series, polynomial\_over\_power\_ser}$$

$$Y^2 + Y + Z$$

$$-X\,Y - Y^2 - Y + 1 + (X+1)\,Z$$

```
> P := PowerSeries([y, z]):
  U := UnivariatePolynomialOverPowerSeries([y, z], x):
  poly := y·x^3 + (-2·y + z + 1)·x + y:
  U-ExtendedHenselConstruction(poly, [0, 0], 3);
```

$$\left[\left[x = \frac{-\mathrm{RootOf}(\_Z^2 + y) + \mathrm{RootOf}(\_Z^2 + y)\,y - \frac{1}{2}\,\mathrm{RootOf}(\_Z^2 + y)\,z + \frac{1}{2}\,y^2}{y}\right],\right.$$

$$\left.\left[x = \frac{\mathrm{RootOf}(\_Z^2 + y) - \mathrm{RootOf}(\_Z^2 + y)\,y + \frac{1}{2}\,\mathrm{RootOf}(\_Z^2 + y)\,z + \frac{1}{2}\,y^2}{y}\right],\ [x = -y]\right]$$

Figure 2.4: On the right: Weierstrass Preparation Factorization for a univariate polynomial with multivariate power series coefficients. On the Left: Extended Hensel construction applied to a trivariate polynomial for computing its absolute factorization.

```
> P := PowerSeries([y]):
  U := UnivariatePolynomialOverPowerSeries([y], x):
  poly := y·x^3 + (-2·y + 1)·x + y:
  OutputFlag :: name := 'parametric':
  parametricVar :: name := T:
  iter := 3:
  verificationFlag :: boolean := true:
  U-ExtendedHenselConstruction(poly, 0, iter, OutputFlag, parametricVar, verificationFlag);
```

$$\left[\left[y = T^2,\ x = \frac{\mathrm{RootOf}(\_Z^2 + 1)\,T - T^3\,\mathrm{RootOf}(\_Z^2 + 1) + \frac{1}{2}\,T^4}{T}\right],\right.$$

$$\left.\left[y = T^2,\ x = \frac{-\mathrm{RootOf}(\_Z^2 + 1)\,T + T^3\,\mathrm{RootOf}(\_Z^2 + 1) + \frac{1}{2}\,T^4}{T}\right],\ [y = T^2,\ x = -T^3]\right]$$

Figure 2.5: Extended Hensel construction applied to a bivariate polynomial for computing its Puiseux parametrizations around the origin.

For computing Puiseux expansions of multivariate polynomials, we rely on the `PowerSeries` library in MAPLE. The `PowerSeries` library consists of two modules, dedicated respectively

[6] See the wikipedia page `https://en.wikipedia.org/wiki/Algebraic_curve`, as of May 2017.

to multivariate power series over the algebraic closure of $\mathbb{Q}$, and univariate polynomials with multivariate power series coefficients.

Figure 2.4 illustrates *Weiertrass Preparation Factorization*[7]. The command `PolynomialPart` displays all the terms of a power series (or a univariate polynomial over power series) up to a specified degree. In fact, each power series is represented by its terms that have been computed so far together with a program for computing the next ones. A command like `WeiertrassPreparation` computes the terms of the factors p and α up to the specified degree; moreover, the encoding of p and α contains a program for computing their terms in higher degree.

Figures 2.4 and 2.5 illustrate the *Extended Hensel Construction* (EHC) which will be discussed in Chapter 3[8]. For the case of an input bivariate polynomial, see Figure 2.5, the EHC coincides with the Newton-Puiseux algorithm, thus computing the Puiseux parametrizations of a plane curve about a point; this functionality is at the core of the `LimitPoints` command of the `RegularChains` library for computing limit points of quasi-components of the regular chains of dimension one. Note that the latter command can be used in two different flavors `LimitPoints`$(R, coefficient = real)$ and `LimitPoints`$(R, coefficient = complex)$, where the argument `coefficient` is used to indicate the coefficient ring for computing limit points corresponding to input regular chain R. For the case of a univariate polynomial with multivariate polynomial coefficients, the EHC is a weak version of Jung-Abhyankar Theorem[9]. The command `ExtendedHenselConstruction` of the sub-package `UnivariatePolynomialOverPowerSeries` in the `PowerSeries` library provides this latter flavor of the EHC.

2.3 The problem and related work

In regular chain theory, one desirable and challenging objective is, given a regular chain R, to obtain the (non-trivial) limit points of its quasi-component $W(R)$, or equivalently, computing the variety of its saturated ideal $\mathrm{sat}(R)$. The set $\lim(W(R))$ of the non-trivial limit points of $W(R)$ satisfies $V(\mathrm{sat}(R)) = \overline{W(R)} = W(R) \cup \lim(W(R))$. Hence, $\lim(W(R))$ is the set-theoretic difference $V(\mathrm{sat}(R)) \setminus W(R)$. Deducing $\lim(W(R))$ or $V(\mathrm{sat}(R))$ from R is a central question which has theoretical applications (like the so-called *Ritt Problem*) and practical ones (like removing redundant components in triangular decomposition, or tangent cone computation).

In this thesis, the main problem we are facing is computing limit points corresponding to a regular chain. Up to our knowledge, the only way to compute limit points, is via a method based on Gröbner basis theory, using the well-known *Rabinowitsch trick*, see `https://en.wikipedia.org/wiki/Rabinowitsch_trick` (as of May 2017).

This latter method, when applied to the computation of $\mathrm{sat}(R)$ does not take the advantage of the triangular structure of the regular chain R. Therefore, it is desirable to consider a method well-adapted to regular chains. Moreover, one may wonder whether it is possible to compute

[7] See the wikipedia page `https://en.wikipedia.org/wiki/Weierstrass_preparation_theorem`, as of May 2017.

[8] For Hensel Lemma, see the wikipedia page `https://en.wikipedia.org/wiki/Hensel%27s_lemma`, as of May 2017.

[9] See this page from MathOverflow `https://mathoverflow.net/questions/92618/what-is-the-original-statement-of-jung-abhyankar-theorem`

$\lim(W(R))$ in polynomial time with respect to the degrees and coefficient heights of our input regular chain R.

Our method for computing $\lim(W(R))$, when $\text{sat}(R)$ has dimension 1, see Chapter 4, relies on a theorem of D. Munfored [70], which relates the closures of a constructible set in the Euclidean and Zariski topologies. One the problem is transported in the Euclidean topology, one can *Puiseux series expansions*, see [63, 2].

Initially, our implementation of the `LimitPoints` command was built upon Maple's `algcurve` package, which implements Newton-Puiseux's algorithm. But, as we wanted to factor polynomials over multivariate Puiseux series rings, we had to switch to the EHC. And since no implementation of the EHC is available in Maple, we had to realize our own, which was the original motivation for developing the `PowerSeries` library[10].

The EHC was originally introduced in [82] by Sasaki and Kako. Their work was further extended by their students in [47, 81, 48, 46, 83]. In [3], and thus in Chapter 3, we present techniques enhancing the EHC as well as complexity estimates for its main sub-routines. In [51], Kung and Traub also present a complexity analysis for Newton-Puiseux algorithm over the field $\mathbb{C}$ of complex numbers. Considering that EHC method computes all the branches while Newton-Puiseux algorithm computes only one branch among the conjugate branches, our complexity results matches theirs, up to some log factors.

Returning to the problem of computing limit points of quasi-components of regular chains, we have also taken another approach to address this problem, by using changes of coordinates, see [4] and Chapter 6. This work not only focuses on computing limit points but also tries studying regular chains under changes of coordinates specially N other normalization. For more details on Nöther normalization, see books [39, 34] and papers [84, 43].

Turning our attention now to applications of limit point computations, we discuss tangent cones of space curves. Tangent cone computations can be approached at least in two ways. First, one can consider the formulation based on homogeneous components of least degree, see Definition 21. The original algorithm of Mora [65] follows this point of view. Secondly, one can consider the more "intuitive" characterization based on limits of secants, see Lemma 31.

As it was mentioned in Chapter 1, parametric representations of algebraic curves and surfaces are used in Computer Aided Geometric Design. However, working with parametric representation instead of the implicit representation bring its own challenges, in particular the problem of determining missing points. One way of dealing with this problem is to find, when it exists, a parametrization that covers the whole curve or surface, or, in other words, a *normal parametrization*. In the case of a curve, this problem is solved in [85], while it remains open for algebraic surfaces. An alternative approach, taken in [12], [86], and [87] is to compute finitely many parametric representations to cover all the points on the surface.

Another interesting application of computing limit points of quasi-components of regular chains is computing the limit of fractions of multivariate polynomials. Computing limits of such functions is a basic task in multivariate calculus and different mathematical concepts are defined based on these limits. The case of univariate analytic functions, including transcendental ones, has been well studied [40, 41, 80] and the corresponding algorithms are available in popular computer algebra systems. In calculus, we learned how to use *L'hospital's rule* in order to compute the limit of univariate functions. In [54] by G.R. Lawlor, a generalization of

[10]This package can be downloaded from `http://regularchains.org/downloads.html`.

L'hospital's rule has been developed for computing the limit of multivariate functions in some special cases. Surprisingly, the limit computations of multivariate functions is still an active research area.

In [104] S.J. Xiao and G.X. Zeng proposed a first algorithm that, given a multivariate rational function $q \in \mathbb{Q}(x_1, \ldots, x_n)$, decides whether $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q$ is zero or not. The "not-case" includes the situation where $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q$ does not exist as well as the case where it exists but it is not zero.

In [21], C. Cadavid, S. Molina and J.D. Vélez proposed an algorithm, now available in MAPLE as the `limit/multi` command, for determining the existence and possible value of limits of the form $\lim_{(x,y)\to(0,0)} q$, where q is a bivariate rational function, and such that $(0, 0)$ is an isolated zero of the real algebraic set defined by the denominator of q. In a follow-up preprint [98], J.D. Vélez, P. Hernández and C. Cadavid extend the method of [21] to rational functions in three variables, still assuming that the origin is an isolated zero of the denominator. Both papers [21] and [98] rely on the key observation that, for determining the existence and possible value of limits of the form $\lim_{(x,y)\to(0,0)} q$ and $\lim_{(x,y,z)\to(0,0,0)} q$, it is sufficient to study *limits along a real algebraic set* $\chi(q)$, that is, limits of the form $\lim_{(x,y)\to(0,0),(x,y)\in\chi(q)} q$ and $\lim_{(x,y,z)\to(0,0,0),(x,y,z)\in\chi(q)} q$.

The method of S.J. Xiao and G.X. Zeng [104] has the advantage of not making any assumptions on the number of variables nor the zero set of the denominator. Meanwhile, the works of C. Cadavid, S. Molina, J.D. Vélez and P. Hernández avoid the use of infinitesimal elements and rely on a deeper geometrical insight, through a notion of *discriminant variety*; unfortunately, the recourse to singular loci and irreducible decomposition is a limitation in view of an implementation of the method proposed in [21].

In [8], we have proposed an algorithm for determining the existence and possible value of $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q$, for an arbitrary number n of variables. As in [21] and [98], we assume that the origin is an isolated zero of the denominator of the rational function q. However, we avoid the computation of singular loci and the decomposition into irreducible components of the real and complex algebraic sets involved in the method. Instead, we take advantage of the theory of regular chains and the RealTriangularize algorithm [23, 22] for decomposing semi-algebraic systems.

The experimental results reported in Section 6 of [8], suggest that our algorithm can solve more problems than the algorithm of S.J. Xiao and G.X. Zeng, in particular when the number of variables increases.

As it was mentioned, we have made an assumption on the problem of computing the limit of fractions of multivariate functions: the origin must be an isolated zero of the denominator of q. The relaxation of this assumption is discussed in Section 8.6.

Chapter 3

Extended Hensel Construction

3.1 Introduction

The *Extended Hensel Construction* (EHC) is an algorithm which is used for factorizing univariate polynomials with power series coefficients. It was proposed in [82] by T. Sasaki and F. Kako. Their goal was to provide a practically more efficient alternative to the classical Newton-Puiseux method for univariate power series coefficients. In the same paper, Sasaki and Kako proposed an extension of the EHC to power series coefficients in more than one variable. Figure 3.1 illustrates our implementation of the EHC in the `PowerSeries` library, available at `www.regularchains.org`.

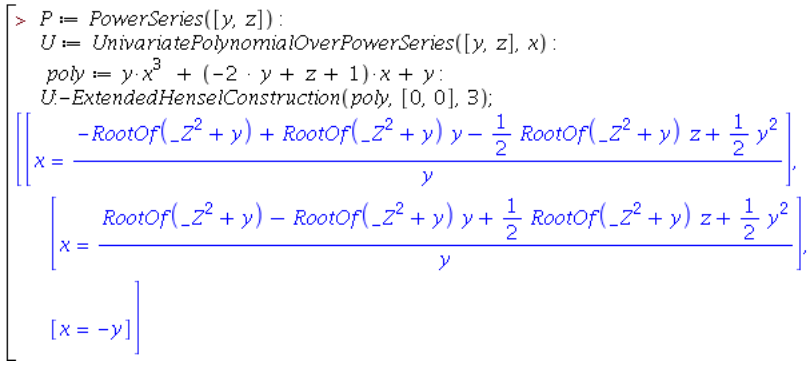


Figure 3.1: EHC applied to a trivariate polynomial.

The work of Sasaki and Kako was further extended by their students, see the papers [47, 81, 48, 46, 83]. See also the works of S. Abhyankar [1] and T.-C. Kuo [52]. The EHC relies on the so-called *Yun-Moses polynomials* originally introduced in [69], studied in [94], and called *Lagrange interpolation polynomials* in [82]. The definition of those polynomials suggests to compute them by applying the Extended Euclidean Algorithm (EEA) over a field of multivariate rational functions. In practice, this is a computational bottleneck. In [81], Sasaki and D. Inaba suggest to use Gröbner bases instead and report on favourable experimental results.

In this chapter, we propose a new method for computing the Yun-Moses polynomials using Wronskian matrices. For an input bivariate polynomial $F(X, Y)$ with coefficients in a field

k and total degree d, we show that the Yun-Moses polynomials (needed when applying the EHC to $F(X, Y)$) can be computed within $O(d^3\,\mathsf{M}(d))$ operations in **k**, where $n \longmapsto \mathsf{M}(n)$ is a (polynomial) multiplication time [99]. In addition, we exhibit a new strategy for performing the lifting steps so that the k-th lifting step of the EHC applied to $F(X, Y)$, can be computed within $O(k\,d\,\mathsf{M}(d)^2)$ operations in **k** (instead of $O(k^2\,d\,\mathsf{M}(d)^2)$ in a direct approach) or within $O(k\,d\,\mathsf{M}(d))$ operations in the algebraic closure of **k**. These enhancements of the EHC are described in Sections 3.3 to 3.4, and supported by the experimentation reported in Section 3.5.

In [51], H.T. Kung and J.F. Traub present a complexity analysis for the Newton-Puiseux method over the field $\mathbb{C}$ of complex numbers. They show that the first k iterations of Newton-Puiseux on an input bivariate polynomial of degree d requires $O(d\,k\,\mathsf{M}(k))$ operations in $\mathbb{C}$ using a *linear lifting scheme* (Theorem 5.2 in [51]) and $O(d\,\mathsf{M}(k))$ operations in $\mathbb{C}$ using a *quadratic lifting scheme* (Corollary 5.1 in [51]). This latter estimate is improved in [28] by D. V. Chudnovsky and G. V. Chudnovsky, yielding $O(d\,k)$ operations in $\mathbb{C}$. When the base field **k** is finite, state of the art algorithms are presented by A. Poteaux and M. Rybowicz in [73].

In both [51] and [28], the estimated cost is for computing a *single branch*. Thus, for computing all branches, the costs of the linear and quadratic lifting schemes of [51] become respectively $O(d^2\,k\,\mathsf{M}(k))$ and $O(d^2\,\mathsf{M}(k))$ operations in $\mathbb{C}$. The EHC currently uses a linear lifting scheme and, with the enhancements proposed in this chapter, it computes all the branches, for the first k operations, within $O(k^2\,d\,\mathsf{M}(d))$ operations in $\mathbb{C}$. The experimentation reported in Section 3.5 show that, for problems of practical interest, an EHC implementation can outperform counterparts based on the linear and quadratic lifting schemes of [51]. Since we implemented both Kung and Traub's algorithm and our enhanced EHC, let us go further in comparing their algebraic complexity. All the above mentioned algorithms need to factor a univariate polynomial over $\mathbb{C}$. This is the Newton polynomial of $F(X, Y)$ in the case of the EHC and the polynomial $F(X, 0)$ for the algorithm of Kung and Traub. If both polynomials split into linear factors over **k**, where **k** is $\mathbb{Q}$ or an algebraic extension of $\mathbb{Q}$, and putting aside the cost of factoring those polynomials (which can be regarded as similar), the total cost, counting operations in $\mathbb{C}$, of factoring $F(X, Y)$ into linear factors in X over $\mathbb{C}(\langle Y^* \rangle)$, computing k terms in each branch, is $O(d^3\mathsf{M}(d) + k^2\,d\,\mathsf{M}(d))$ for the EHC and $O(d^2\,k\,\mathsf{M}(k))$ (resp. $O(d^2\,\mathsf{M}(k))$) the algorithm of Kung and Traub using a linear (resp. quadratic) lifting scheme.

Let $F(X, Y)$ be the input polynomial for both EHC algorithm and Kung and Traub algorithm. In the fair case, assume all roots of $F(X, 0)$ and the Newton polynomial are simple root. Therefore the Kung and Traub algorithm will be in regular case and the EHC algorithm will be called once. The complexity of Kung and Traub algorithm, to compute all branches, is $O(d^2\,k\,\mathsf{M}(k))$ operations over $\mathbb{C}$, using a linear lifting scheme, and the complexity of our EHC algorithm, to compute all branches, is $O(d^3\mathsf{M}(d) + k^2 d\mathsf{M}(d))$ operations over $\mathbb{C}$. Hence, for the fixed input polynomial, our EHC algorithm is asymptotically faster than Kung and Traub algorithm. By Corollary 6 in [59], the splitting field of a degree d polynomial $f(X) \in \mathbb{Z}[X]$ can be computed in time polynomial to the degree of the splitting field over $\mathbb{Q}$ and $(\sum_{i=0}^{d} a_i^2)^{1/2}$ where, for $i = 0, \ldots, d$, a_i is the coefficient of X^i in $f(X)$. So the worst case of this algorithm costs $O(d!)$. But Renault and Yokoyama in the more recent result in [75, 74] presented an algorithm in which the cost of computing the splitting field of $f(X)$ is a polynomial to d, $c(G_f)$ and $M(log(B_R))$ where G_f is Galois group of f and $c(G_f)$ is $c-$size of it [74], also B_R is such that $\prod_{i=1}^{s} q_i = O(B_R)$ where multi-modular algorithm happens in q_i's which are powers of prime p and finally $M(q)$ is the unit cost of integer arithmetics of size q.

In practice, the EHC has the advantage that its computation flow has a simpler structure and offers opportunities for efficient implementation. This observation is based on our experience with both approaches through a series of papers [6, 9, 8] .

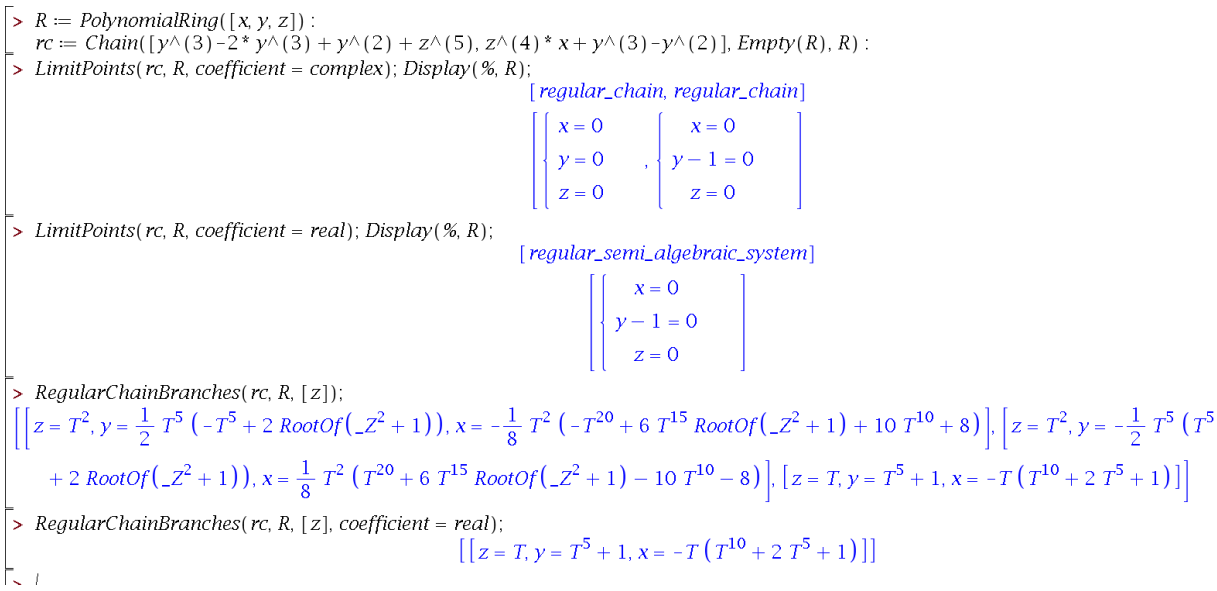


Figure 3.2: Computational of limit points: complex and real cases.

In addition to polynomial factorization, the EHC can be applied to the computation of limits of multivariate rational functions [8] and tangent cones [9]. In [6], an algorithm is proposed for computing the non-trivial limit points of the quasi-component $W(T)$ of a regular chain $T \subset \mathbb{Q}[X_1, \ldots, X_n]$. Those points form the set $\overline{W(T)} \setminus W(T)$, where $\overline{W(T)}$ is the Zariski closure of $W(T)$.

In Chapter 5, we use the EHC for computing the non-trivial limit points of the *real* quasi-component of T. To be precise, letting $W_{\mathbb{R}}(T) := Z_{\mathbb{R}}(T) \setminus Z_{\mathbb{R}}(h_T)$, we are interested in the set $\overline{W_{\mathbb{R}}(T)} \setminus W_{\mathbb{R}}(T)$, where $\overline{W_{\mathbb{R}}(T)}$ is the closure of $W_{\mathbb{R}}(T)$ in $\mathbb{R}^n$ endowed with the Euclidean topology. Unfortunately, it is not true that the non-trivial limit points of $W_{\mathbb{R}}(T)$ are the non-trivial limit points of $W(T)$ with real coordinates. Figure 3.2 yields a counter-example, which illustrates how the factorization produced by the EHC helps computing the limit points of both $W(T)$ (complex case) and $W_{\mathbb{R}}(T)$ (real case). Section 5.2 is devoted to this question.

3.2 Extended Hensel construction

The purpose of this chapter requires a somehow detailed review of the EHC. Most of the proofs are omitted, though, and we refer to [82]. We also recall the notion of Puiseux series and refer to the book of G. Fischer [35] for this topic.

Notation 1. *Let $F(X, Y) \in \mathbb{C}[X, Y]$ be a bivariate polynomial with complex number coefficients. We assume that F is monic and square-free as a univariate polynomial in X; we denote by d its partial degree w.r.t. X. We assume that F has at least two terms and that $F(X, 0) = X^d$*

holds. We explain in Remark 2 how to reduce to this latter hypothesis. For $f_1, \ldots, f_m$ in some polynomial ring, we denote by $\langle f_1, \ldots, f_m \rangle$ the ideal that $f_1, \ldots, f_m$ generate in that ring.

Newton line. We plot each non-zero term $c\, X^{e_x} Y^{e_y}$ of $F(X, Y)$ to the point of coordinates (e_x, e_y) in the Euclidean plane equipped with Cartesian coordinates. We call *Newton Line* the straight line L passing through the point $(d, 0)$ and another point, such that no other points lie below L. The equation of L is $e_x/d + e_y/\delta = 1$ for some $\delta \in \mathbb{Q}$. We define $\hat{\delta}, \hat{d} \in \mathbb{Z}^{>0}$ such that $\hat{\delta}/\hat{d} = \delta/d$ and $\gcd(\hat{\delta}, \hat{d}) = 1$ both hold.

Newton polynomial. The sum of all the terms of $F(X, Y)$, which are plotted on the Newton line of F, is called the *Newton polynomial* of F. We denote it by $F^{(0)}$. Observe that Newton's polynomial is a homogeneous polynomial in $(X, Y^{\delta/d})$. Let $\zeta_1, \ldots, \zeta_r \in \mathbb{C}$ be the distinct roots of $F^{(0)}(X, 1)$, for some $r \geq 2$. Hence we have $\zeta_i \neq \zeta_j$ for all $1 \leq i < j \leq r$ and there exist positive integers $m_1 \leq m_2 \leq \cdots \leq m_r$ such that, using the homogeneity of $F^{(0)}(X, Y)$, we have

$$F^{(0)}(X, Y) = (X - \zeta_1 Y^{\delta/d})^{m_1} \cdots (X - \zeta_r Y^{\delta/d})^{m_r}.$$

The *initial factors* of $F^{(0)}(X, Y)$ are

$$G_i^{(0)}(X, Y) := (X - \zeta_i Y^{\delta/d})^{m_i},$$

for $1 \leq i \leq r$. For simplicity, we put $\hat{Y} = Y^{\hat{\delta}/\hat{d}}$.

Puiseux series. Let $\mathbf{k}$ be an algebraic number field and $\overline{\mathbf{k}}$ its algebraic closure. We denote by $\mathbf{k}[[Y]]$ and $\mathbf{k}\langle Y \rangle$ the respective rings of formal power series and convergent power series in Y with coefficients in $\mathbf{k}$. We denote by $\mathbf{k}[[Y^*]] = \bigcup_{\ell=1}^{\infty} \mathbf{k}[[Y^{\frac{1}{\ell}}]]$ the ring of *formal Puiseux series*. Hence, given $\varphi \in \mathbf{k}[[Y^*]]$, there exists $\ell \in \mathbb{N}_{>0}$ such that $\varphi \in \mathbf{k}[[Y^{\frac{1}{\ell}}]]$ holds and we can write $\varphi = \sum_{m=0}^{\infty} a_m Y^{\frac{m}{\ell}}$, for some $a_0, \ldots, a_m, \ldots \in \mathbf{k}$. We denote by $\mathbf{k}((Y^*))$ the quotient field of $\mathbf{k}[[Y^*]]$. Let $\varphi \in \mathbf{k}[[Y^*]]$ and $\ell \in \mathbb{N}$ such that $\varphi = f(Y^{\frac{1}{\ell}})$ holds for some $f \in \mathbf{k}[[Y]]$. We say that the Puiseux series φ is *convergent* if we have $f \in \mathbf{k}\langle Y \rangle$. The ring of convergent Puiseux series is denoted by $\mathbf{k}\langle Y^* \rangle$ and its quotient field by $\mathbf{k}(\langle Y^* \rangle)$. We recall Puiseux's theorem: if $\mathbf{k}$ is an algebraically closed field of characteristic zero, the field $\mathbf{k}((Y^*))$ of formal Puiseux series over $\mathbf{k}$ is the algebraic closure of the field of formal Laurent series over $\mathbf{k}$; moreover, if $\mathbf{k} = \mathbb{C}$, then the field $\mathbb{C}(\langle Y^* \rangle)$ of convergent Puiseux series over $\mathbb{C}$ is algebraically closed as well.

The purpose of the EHC, as stated in Algorithm 1, is to factorize $F(X, Y)$ as $F(X, Y) = G_1(X, Y) \cdots G_r(X, Y)$, with $G_i(X, Y) \in \mathbb{C}(\langle Y^* \rangle)[X]$ and $\deg_X(G_i) = m_i$, for $1 \leq i \leq r$. Thus, the EHC factorizes $F(X, Y)$ over $\mathbb{C}(\langle Y^* \rangle)$. However, $\deg_X(G_i) = 1$ may not hold for some i. Nevertheless, as shown hereafter factorizing $F(X, Y)$ into linear factors is achieved by repeated applications of the EHC. Lemma 2 and Theorem 1 are the fundamental results of the EHC.

Lemma 2 (Yun-Moses polynomials)**.** *Let $\hat{G}_i(X, \hat{Y}) \in \mathbb{C}\langle \hat{Y} \rangle[X]$, for $i = 1, \ldots, r$ with $r \geq 2$, be homogeneous polynomials in $(X, \hat{Y})$ such that $\gcd(\hat{G}_i, \hat{G}_j) = 1$ for any $i \neq j$. Let $d = \deg_X\left(\hat{G}_1 \cdots \hat{G}_r\right)$ and $\deg_X(\hat{G}_i) = m_i$, for $i = 1, \ldots, r$. Then, for each $\ell \in \{0, \ldots, d-1\}$, there exists a unique set of polynomials $\{W_i^{(\ell)}(X, \hat{Y}) \in \mathbb{C}\langle \hat{Y} \rangle[X] \mid i = 1, \ldots, r\}$ satisfying*

$$W_1^{(\ell)}\left(\left(\hat{G}_1 \cdots \hat{G}_r\right)/\hat{G}_1\right) + \cdots + W_r^{(\ell)}\left(\left(\hat{G}_1 \cdots \hat{G}_r\right)/\hat{G}_r\right) = X^{\ell} \hat{Y}^{d-\ell},$$

where $\deg_X(W_i^{(\ell)}(X, \hat{Y})) < \deg_X(\hat{G}_i(X, \hat{Y}))$, $i = 1, \ldots, r$. The polynomials $W_i^{(0)}, \ldots, W_i^{(d-1)}$ for $1 \leq i \leq r$ are homogeneous in $(X, \hat{Y})$ of degree m_i. We call Yun-Moses polynomials *the elements of $\{W_i^{(\ell)} \mid (\ell, i) \in \{0, \ldots, d-1\} \times \{1, \ldots, r\}\}$.*

PROOF. We shall first prove that there exists only one set of polynomials $\{W_i^{(\ell)}(x,1) \mid i = 1,\ldots,r\}$ satisfying the condition in the above lemma, when $\hat{Y} = 1$. Using the extended Euclidean algorithm, one can compute $A_1,\ldots,A_r \in \mathbb{C}[X]$ such that $A_1\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_1} + \cdots + A_r\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_r} = 1$. If we multiply both sides of the above equality by X^ℓ, then we have

$$A_1X^\ell\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_1} + \cdots + A_rX^\ell\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_r} = X^\ell. \tag{3.1}$$

For each $i = 1,\ldots,r-1$, let $Q_i, R_i \in \mathbb{C}[X]$ such that $A_iX^\ell = Q_i\hat{G}_i + R_i$ and $\deg_X(R_i) < \deg_X(\hat{G}_i)$. Thus the last equality can be re-written as:

$$R_1\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_1} + \cdots + R_{r-1}\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_{r-1}} + (A_rX^\ell + \sum_{i=1}^{r-1} Q_i\hat{G}_r)\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_r} = X^\ell.$$

Observe that we have $\deg_X(R_i\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_i}) < d$ for $i = 1,\ldots,r-1$, $\deg_X(\frac{\hat{G}_1\cdots\hat{G}_r}{\hat{G}_r}) = d - m_r$, and also $\ell < d$. Combined with relation 3.1, we obtain

$$\deg_X(A_rX^\ell + \sum_{i=1}^{r-1} Q_i\hat{G}_r) < m_r = \deg_X(\hat{G}_r).$$

Hence, we set $W_i^{(\ell)}(X,1) = R_i$, for $i = 1,\ldots,r-1$ and $W_r^{(\ell)}(X,1) = A_rX^\ell + \sum_{i=1}^{r-1} Q_i\hat{G}_r$. Since

$$\deg_X\left(W_i^{(\ell)}(X,1)\left(\hat{G}_1\cdots\hat{G}_r\right)/\hat{G}_i\right) < d,$$

we can homogenize in degree d both $W_i^{(\ell)}(X,1)$ and $\hat{G}_i(X,1)$, for $i = 1,\ldots,r$, using $\hat{Y}$ as homogenization variable. This homogenization process defines each $W_i^{(\ell)}(X,\hat{Y})$ uniquely. Moreover, we have $\deg_X(W_i^{(\ell)}(X,\hat{Y})) < \deg_X(\hat{G}_i)$. □

Theorem 1 (Extended Hensel Construction)**.** *Let F be as in Notation 1 and let $F^{(0)}(X,Y)$ be the Newton polynomial of $F(X,Y)$. We denote by $G_1^{(0)}(X,Y),\ldots,G_r^{(0)}(X,Y)$ the initial factors of $F^{(0)}(X,Y)$. Hence we have*

$$F^{(0)}(X,Y) = G_1^{(0)}(X,Y)\cdots G_r^{(0)}(X,Y),$$

where $G_i^{(0)}(X,Y) = (X - \zeta_iY^{\hat{\delta}/\hat{d}})^{m_i}$ for $i = 1,\ldots,r$ and $\zeta_i \in \mathbb{C}$. We define the ideal

$$S_k = \langle X^dY^{(k+0)/\hat{d}}, X^{d-1}Y^{(k+\hat{\delta})/\hat{d}},\ldots,X^0Y^{(k+d\hat{\delta})/\hat{d}}\rangle, \tag{3.2}$$

for $k = 1,2,\ldots$. Then, for all integer $k > 0$, we can construct $G_i^{(k)}(X,Y) \in \mathbb{C}\langle Y^{1/\hat{d}}\rangle[X]$, for $i = 1,\ldots,r$, satisfying

$$F(X,Y) = G_1^{(k)}(X,Y)\cdots G_r^{(k)}(X,Y) \mod S_{k+1}, \tag{3.3}$$

and $G_i^{(k)}(X,Y) \equiv G_i^{(0)}(X,Y) \mod S_1$, for all $i = 1,\ldots,r$.

Proof. See Theorem 1 in [82]. The proof is constructive and by induction on k. **Base case:** Since $F(X,Y) \equiv F^{(0)}(X,Y) \mod S_1$, the theorem is valid for $k = 0$. **Inductive step:** Let the theorem be valid up to the $(k-1)$-th construction. We write:

$$G_i^{(k-1)} = G_i^{(0)}(X,Y) + \Delta G_i^{(1)}(X,Y) + \cdots + \Delta G_i^{(k-1)}(X,Y),$$

such that $G_i^{(k')}(X,Y) \in S_{k'}$, and $\deg_X(\Delta G_i^{(k')}(X,Y)) < \deg_X(G_i^{(0)}(X,Y)) = m_i$ for $k' = 1,\ldots,k-1$. These latter properties are part of the induction hypothesis. Now define

$$\Delta F^{(k)}(X,Y) := F(X,Y) - G_1^{(k-1)} \cdots G_r^{(k-1)} \mod S_{k+1}.$$

It follows from the induction hypotheses that $\Delta F^{(k)}(X,Y) \in S_k$ holds. Thus, we can write

$$\Delta F^{(k)}(X,Y) = f_{d-1}^{(k)} X^{d-1} Y^{\hat{\delta}/\hat{d}} + \cdots + f_0^{(k)} X^0 Y^{d\hat{\delta}/\hat{d}} \tag{3.4}$$

where $f_\ell^{(k)} = c_\ell^{(k)} Y^{k/\hat{d}}$ and $c_\ell^{(k)} \in \mathbb{C}$ for $\ell = 0,\ldots,d-1$. We construct $G_i^{(k)}(X,Y)$, and thus $\Delta G_1^{(k)},\ldots,\Delta G_r^{(k)}$, such that we have:

$$G_i^{(k)}(X,Y) = G_i^{(k-1)}(X,Y) + \Delta G_i^{(k)}(X,Y),$$

and $\Delta G_i^{(k)}(X,Y) \equiv 0 \mod S_k$. Then we have:

$$\begin{aligned} F(X,Y) &\equiv \textstyle\prod_{i=1}^r \left(G_i^{(k-1)} + \Delta G_i^{(k)}\right) \bmod S_{k+1} \\ &\equiv G_1^{(k-1)} \cdots G_r^{(k-1)} + \Delta G_1^{(k)} \tfrac{F^{(0)}}{G_1^{(0)}} + \cdots + \Delta G_r^{(k)} \tfrac{F^{(0)}}{G_r^{(0)}} + \underbrace{\text{other terms}}_{\text{containing } \Delta G_i^{(k)}(X,Y)\Delta G_j^{(k)}(X,Y)} \mod S_{k+1} \\ &\equiv G_1^{(k-1)} \cdots G_r^{(k-1)} + \Delta G_1^{(k)} \tfrac{F^{(0)}}{G_r^{(0)}} + \cdots + \Delta G_r^{(k)} \tfrac{F^{(0)}}{G_r^{(0)}} \mod S_{k+1}. \end{aligned}$$

Indeed, we have $\Delta G_i^{(k)}(X,Y)\Delta G_j^{(k')}(X,Y) \equiv 0 \mod S_{k+1}$ for $k,k' \geq 0$, from the induction hypotheses and the relation $S_k S_{k'} = S_{k+k'}$. Therefore, we have

$$\Delta F^{(k)} \equiv \Delta G_1^{(k)} \frac{F^{(0)}}{G_1^{(0)}} + \cdots + \Delta G_r^{(k)} \frac{F^{(0)}}{G_r^{(0)}} \bmod S_{k+1}. \tag{3.5}$$

If in Lemma 2, we let $\hat{G}_i(X,\hat{Y}) = G_i^{(0)}(X,\hat{Y})$, combining Equations (3.4) and (3.5), one can solve for $\Delta G_1^{(k)},\ldots,\Delta G_r^{(k)}$

$$\begin{aligned} \textstyle\sum_{i=1}^r \Delta G_i^{(k)} \frac{F^{(0)}}{G_i^{(0)}} &= \textstyle\sum_{\ell=0}^{d-1} f_\ell^{(k)} X^\ell \hat{Y}^{d-\ell} \\ &= \textstyle\sum_{\ell=0}^{d-1} f_\ell^{(k)} \left(\sum_{i=1}^r W_i^{(\ell)} \frac{F^{(0)}}{G_i^{(0)}}\right) \\ &= \textstyle\sum_{i=1}^r \left(\sum_{\ell=0}^{d-1} f_\ell^{(k)} W_i^{(\ell)}\right) \frac{F^{(0)}}{G_i^{(0)}}. \end{aligned}$$

Since $\deg_X(f_\ell^{(k)} W_i^{(\ell)}) < \deg_X(G_i^{(0)})$ and $\deg_X(\Delta G_i^{(k)}(X,Y)) < \deg_X(G_i^{(0)})$ both hold for $i = 1,\ldots,r$, we deduce $\Delta G_i^{(k)}(X,Y) = \sum_{\ell=0}^{d-1} W_i^{(\ell)}(X,Y) f_\ell^{(k)}(Y)$, for $i = 1,\ldots,r$. □

Remark 1. *Theorem 1 still holds if $G_1^{(0)}(X,Y),\ldots,G_r^{(0)}(X,Y)$ just satisfy the same properties as $\hat{G}_1(X,\hat{Y}),\ldots,\hat{G}_r(X,\hat{Y})$ of Lemma 2.*

Remark 2. *Write $F(X,0) = X^d + a_1X^{e_1} + \cdots + a_mX^{e_m} + a_{m+1}$. If the polynomial F doesn't satisfy the assumption $F(X,0) = X^d$, we apply to $F(X,Y)$ the change of variables $(X,Y) := (W/Y^{1/d}, Y)$ and factor out $1/Y$. We obtain a polynomial $\overline{F}(W,Y)$ satisfying $\overline{F}(W,0) = W^d$. After applying the EHC to $\overline{F}$, we multiply each computed factor by $1/Y^{1/d}$ and revert the change of variables.*

Remark 3. *Assume the Newton polynomial factorizes to $F^{(0)} = (X - aY)^d$ for some $a \in \mathbf{k}$. Since $d \geq 2$, we split $F^{(0)}$ into at least two factors, as follows. Let $Y = 1$ and apply the change of variables $X := W - a/d$, called the* Shreedharacharya-Tschirnhaus *trick in Lemma 1.8 of [61]. After homogenizing back, we obtain a polynomial $\overline{F}(W,Y)$ whose Newton polynomial splits into at least two co-prime factors. Applying the EHC to $\overline{F}(W,Y)$ produces at least two factors.*

Algorithm 1: EHC_Lift

Input: a given F as in Notation 1 and a positive integer k
Output: Extended Hensel Construction on F.

begin
  Compute the Newton polynomial $F^{(0)}$ and $\hat{\delta}, \hat{d}$;
  Compute $F^{(0)} = G_1^{(0)} \cdots G_r^{(0)}$, see Remark 1;
  if $r = 1$ **then**
    Apply the change of variable in Remark 3;
  Compute the Yun-Moses polynomials $W_i^{(\ell)}$ for $i = 1, \cdots, r$ and $\ell = 0, \cdots, d-1$; (see Section 3.3);
  for $j = 1, \ldots, k$ **do**
    Compute $\Delta F^{(j)}(X,Y) := F(X,Y) - \prod_{i=1}^{r} G_i^{(j-1)} \mod \bar{S}_{j+1}$ (see Section 3.4 as well as Page 13 of [82]);
    Compute $\Delta G_i^{(j)} = \sum_{\ell=0}^{m-1} W_i^{(\ell)} f_\ell^{(j)}$, for $i = 1, \cdots, r$;
    Let $G_i^{(j)} = G_i^{(j-1)} + \Delta G_i^{(j)}$ for $i = 1, \cdots, r$;
  Reverse the change of variable, if any;
  return $G_1^{(k)}, \ldots, G_r^{(k)}$;
end

3.2.1 Extended Hensel construction of multivariate polynomials

Let $F(X, Y_1, \ldots, Y_n) \in \mathbf{k}[X, Y_1, \ldots, Y_n]$ be monic, square-free in X, and $F(X, 0, \ldots, 0) = X^d$. Let U be a new variable called *total degree variable*. We perform the following substitutions in F:

$$Y_1 \mapsto U\,Y_1, \ldots, Y_n \mapsto U\,Y_n.$$

The intention of this substitution is to view F as a bivariate polynomial in X, U where $Y_1, \ldots, Y_n$ are now regarded as parameters. To be more precise, we regard F as a polynomial in $\mathbf{k}(Y_1, \ldots, Y_n)[X, U]$.

Newton Line for multivariate polynomials. We plot each non-zero term $c\,X^{e_x}Y_1^{e_{y_1}} \cdots Y_n^{e_{y_n}}$ of $F(X, Y_1, \ldots, Y_n)$ to the point of coordinates $(e_x, e_{y_1} + \cdots + e_{y_n})$ in the (e_x, e_U) two-dimensional

Euclidean plane equipped with Cartesian coordinates, where $e_U := e_{y_1} + \cdots + e_{y_n}$. We call *Newton Line* the straight line L passing through the point $(d, 0)$ and another plotted point such that no other plotted points lie below L. This line L has a Cartesian equation of the form $e_x/d + e_U/\delta = 1$ for some $\delta \in \mathbb{Q}$. We define $\hat{\delta}, \hat{d} \in \mathbb{Z}^{>0}$ such that $\hat{\delta}/\hat{d} = \delta/d$ and $\gcd(\hat{\delta}, \hat{d}) = 1$ both hold.

Newton polynomial for multivariate polynomials. The sum of all the terms of $F(X, Y_1, \ldots, Y_n)$, which are plotted on the Newton line of F, is called the *Newton polynomial* of F. We denote it by $F^{(0)}$. Observe that Newton polynomial is a homogeneous polynomial in $(X, U^{\delta/d})$.

Lemma 3. *Any non-constant monomial of F is plotted at one of the points of the set*

$$\left\{(d, (k+0)/\hat{d}), (d-1, (k+\hat{\delta}/\hat{d})), \ldots, (0, (k+d\hat{\delta})/\hat{d})\right\}$$

Note again that $F^{(0)}$ is homogeneous in $(X, \hat{U})$ where $\hat{U} := U^{\hat{\delta}/\hat{d}}$. Define

$$S_k := \left\{X^d\, U^{(k+0)/\hat{d}}, X^{d-1}\, U^{(k+\hat{\delta})/\hat{d}}, \ldots, U^{(k+d\hat{\delta})/\hat{d}}\right\}.$$

Let also

$$S'_k := \left\{X^d, X^{d-1}, \ldots, 1\right\}.$$

In general, $F^{(0)}(X, Y_1, \ldots, Y_n)$ can be factorized in $\mathbb{C}[X, Y_1, \ldots, Y_n]$ as following:

$$\begin{aligned} &F^{(0)}(X, Y_1, \ldots, Y_n) = H_1^{(0)}(X, Y_1, \ldots, Y_n)^{m_1} \cdots H_s^{(0)}(X, Y_1, \ldots, Y_n)^{m_s} \\ &\gcd(H_i^{(0)}, H_j^{(0)}) = 1, \quad \text{for any } i \neq j, \text{ and } \quad \deg_X(H_i^{(0)}) = r_i \end{aligned} \tag{3.6}$$

where each $H_i^{(0)}$ is an irreducible factor for $F^{(0)}$. Suppose $s \geq 2$. We let

$$F_i^{(0)} = H_i^{(0)}(X, Y_1, \ldots, Y_n)^{m_i} \quad \text{for } i = 1 \ldots s.$$

Lemma 4 (Yun-Moses polynomials)**.** *For each $\ell = 1, \ldots, d-1$, there exist Yun-moses polynomials $W_1^{(\ell)}, \ldots W_s^{(\ell)} \in \mathbb{C}(Y_1, \ldots, Y_n)[X]$ satisfying,*

- $W_1^{(\ell)}\left(F_1^{(0)} \cdots F_s^{(0)}/F_1^{(0)}\right) + \cdots + W_1^{(\ell)}\left(F_1^{(0)} \cdots F_s^{(0)}/F_s^{(0)}\right) = X^\ell$
- $\deg_X(W_i^{(\ell)}) < \deg_X(F_i^{(0)}), \quad i = 1, \ldots, s$

For each $i = 1, \ldots, s$, $W_i^{(\ell)} \hat{U}^{d-\ell}$ are homogenous in X and $\hat{U}$ where $\hat{U} = U^{\hat{\delta}/\hat{d}}$ of total degree $\deg_X(F_i^{(0)})$ w.r.t X and $\hat{U}$.

Remark 4. *Since $W_i^{(\ell)}$ are rational functions in $Y_1, \ldots, Y_n$ with different denominators, it is not possible any time to clear denominators by multiplying a simple expression to the equality.*

Theorem 2 (Extended Hensel Construction for multivarate case)**.** *Let $F(X, Y_1, \ldots, Y_n) \in \mathbb{C}[X, Y_1, \ldots, Y_n]$ be monic in X and square-free. Let also $F^{(0)}$ be its Newton polynomial which is factorized as in 3.6. Then for any integer k, we can construct $F_i^{(k)}$ such that*

- $F(X, Y_1, \ldots, Y_n) \equiv F_1^{(k)}(X, Y_1, \ldots, Y_n) \cdots F_s^{(k)}(X, Y_1, \ldots, Y_n) \mod S'_{k+1}$
- $F_1^{(k)}(X, Y_1, \ldots, Y_n) \equiv H_i(X, Y_1, \ldots, Y_n)^{m_i} \mod S'_1, \quad i = 1, \ldots, s$

Each coefficient of the term $X^{e_X} U^{e_U}$ in $F_i^{(k)}(X, Y_1, \ldots, Y_n)$ is of the form $\frac{N}{D}$ where N and D are homogenous polynomials in $Y_1, \ldots, Y_n$ and $\deg(N) - \deg(D) = e_U$.

3.2.2 Complete factorization in $\mathbb{C}(\langle Y^* \rangle)[X]$

To separate all the branches of the curve $F(X, Y) = 0$ around the origin, one should use a sufficient accuracy (that is, degree in Y) for the lifted factors. Theorem 4.5 in [44] suggests a minimum accuracy of $B := 2\, deg_X(F)\, deg_Y(F)$.

After applying EHC_Lift(F, k) with $k = \hat{d}B - \hat{\delta}$, which is the number of iteration needed for accuracy B, one needs to re-apply the EHC on each lifted factor of multiplicity greater than 1. For each additional call, with a lifted factor $G := G_i^{(k)}(X, Y)$, the value of k is set to $\hat{d}B' - \hat{\delta}$, where $B' := 2\, deg_X(G)\, deg_Y(G)$. Moreover, for each lifted factor $G_i^{(k)}(X, Y)$, with the notations of Theorem 1, we apply the change of coordinates $X = X - \zeta_i Y$. See [82] for details. This process generates a tree of calls to the EHC. Obviously, one needs to do at most d calls in total.

One may wonder what is the maximum total number of lifting steps along a branch of that tree. One can easily verify that after completing the factorization of F in $\mathbb{C}(\langle Y^* \rangle)[X]$ into linear factors, that this maximum is given by $\hat{d}B - \hat{\delta}$.

3.3 On the Yun-Moses polynomials

We use the notations of Section 3.2, including the proof of Theorem 1. Define $\tilde{Y} = Y^{1/\hat{d}}$. In this section, we take advantage of the fact each Yun-Moses polynomial is a rational function in X, Y, whose denominator is just a power of Y.

Lemma 5. *We have $\Delta F^{(k)} \in \mathbb{K}[X, \tilde{Y}]$, for all $k = 1, 2, \ldots$.*

PROOF. From the Extended Hensel Construction, it is known that $\Delta F^{(k)} \equiv F - G_1^{(k-1)} \cdots G_r^{(k-1)} \bmod S_{k+1}$, where $G_i^{(k-1)} = G_i^{(0)} + \Delta G_i^{(1)} + \cdots + \Delta G_i^{(k-1)}$. And we have

$$\Delta F^{(k)}(X, \tilde{Y}) = f_{d-1}^{(k)} X^{d-1} \tilde{Y}^{\hat{\delta}} + \cdots + f_0^{(k)} X^0 \tilde{Y}^{d\hat{\delta}}$$

where $f_\ell^{(k)} = c_\ell^{(k)} \tilde{Y}^k$ with $c_\ell^{(k)} \in \mathbb{C}$ for $\ell = 0, \ldots, d-1$. The goal is to prove $c_\ell^{(k)} \in \mathbb{K}$ and we prove it by induction. For $k = 1$, $\Delta F^{(1)} \equiv F - F^{(0)} \bmod S_2$. Since $F, F^{(0)} \in \mathbb{K}[X, Y]$, we have $\Delta F^{(1)} \in \mathbb{K}[X, \tilde{Y}]$. Now assume $\Delta F^{(k-1)} \in \mathbb{K}[X, \tilde{Y}]$, thus $G_1^{(k-2)} \cdots G_r^{(k-2)} = F - \Delta F^{(k-1)} \in \mathbb{K}[X, \tilde{Y}]$. We want to prove $\Delta F^{(k)} \in \mathbb{K}[X, \tilde{Y}]$. In modulo S_{k+1}, we have

$$\begin{aligned} \Delta F^{(k)} &\equiv F - G_1^{(k-1)} \cdots G_r^{(k-1)} \\ &\equiv F - (G_1^{(k-2)} + \Delta G_1^{(k-1)}) \cdots (G_r^{(k-2)} + \Delta G_r^{(k-1)}) \\ &\equiv F - (G_1^{(k-2)} \cdots G_r^{(k-2)} + \textstyle\sum_{i=1}^r \Delta G_i^{(k-1)} \frac{F^{(0)}}{G_i^{(0)}}). \end{aligned}$$

Last equivalence is valid, due to $\Delta G_i^{(k-1)} \Delta G_j^{(k-1)} \equiv 0 \bmod S_{k+1}$ and $(G_1^{(k-1)} \cdots G_r^{(k-1)})/G_i^{(k-1)} \equiv (G_1^{(0)} \cdots G_r^{(0)})/G_i^{(0)} \bmod S_{k+1}$. On the other hand, $\Delta G_i^{(k-1)} = \sum_{\ell=0}^{d-1} W_i^{(\ell)} f_\ell^{(k-1)}$. So, we have

$$\begin{aligned} \textstyle\sum_{i=1}^r \Delta G_i^{(k-1)} \frac{F^{(0)}}{G_i^{(0)}} &= \textstyle\sum_{i=1}^r \sum_{\ell=0}^{d-1} W_i^{(\ell)} f_\ell^{(k-1)} \frac{F^{(0)}}{G_i^{(0)}} \\ &= \textstyle\sum_{\ell=0}^{d-1} f_\ell^{(k-1)} \sum_{i=1}^r W_i^{(\ell)} \frac{F^{(0)}}{G_i^{(0)}} \\ &= \textstyle\sum_{\ell=0}^{d-1} f_\ell^{(k-1)} X^d \tilde{Y}^{\hat{\delta}(d-\ell)}, \end{aligned}$$

therefore, modulo S_{k+1}, we have

$$\begin{aligned}\Delta F^{(k)} &\equiv F - (G_1^{(k-1)} \cdots G_r^{(k-1)} + \textstyle\sum_{\ell=0}^{d-1} f_\ell^{(k-1)} X^d \tilde{Y}^{\hat{\delta}(d-\ell)}) \\ &\equiv F - (G_1^{(k-1)} \cdots G_r^{(k-1)} + \textstyle\sum_{\ell=0}^{d-1} c_\ell^{(k-1)} X^d \tilde{Y}^{(k-1)\hat{\delta}(d-\ell)}).\end{aligned}$$

By induction assumption for $k-1$, we have $c_\ell^{(k-1)} \in \mathbb{K}$ and $G_1^{(k-1)} \cdots G_r^{(k-1)} \in \mathbb{K}[X, \tilde{Y}]$, therefore, $\Delta F^{(k)} \in \mathbb{K}[X, \tilde{Y}]$. □

From Lemma 2, the Yun-Moses polynomials associated with the initial factors $G_1^{(0)}, \ldots, G_r^{(0)}$ of $F^{(0)}$ satisfy

$$\sum_{i=1}^{r} W_i^{(\ell)} \frac{F^{(0)}}{G_i^{(0)}} = X^\ell \hat{Y}^{d-\ell} \text{ for } \ell = 0, \cdots d-1, \tag{3.7}$$

where $\hat{Y} = Y^{\hat{\delta}/\hat{d}}$ with $G_i^{(0)} = (X - \zeta_i \hat{Y})^{m_i}$ where ζ_i is a root of $F^{(0)}(X, 1)$ and m_i is its multiplicity. Also, we have $\deg_X(W_i^{(\ell)}) < m_i$, thus, we write $W_i^{(\ell)} = \sum_{j=0}^{m_i-1} w_{i,j}^{(\ell)}(\hat{Y}) X^j$ for any ℓ. Let us fix λ in $\{1, \ldots, r\}$. Define the column vector $\mathcal{X}_\lambda^\ell = [w_{\lambda,j}^{(\ell)}]$. The goal is to find $\mathcal{X}_\lambda^\ell$, what we shall do by solving a system of linear equations. Now for $\mu = 0, 1, \ldots, m_\lambda - 1$, we take the μ-th derivative of each side in Equation (3.7) and let $X = \zeta_\lambda \hat{Y}$ in those derivatives. In other words, we have

$$\frac{\partial^\mu}{\partial X^\mu} \left(\sum_{i=1}^{r} W_i^{(\ell)} \frac{F^{(0)}}{G_i^{(0)}} \right) \Big|_{X=\zeta_\lambda \hat{Y}} = \frac{\partial^\mu}{\partial X^\mu} (X^\ell \hat{Y}^{d-\ell})|_{X=\zeta_\lambda \hat{Y}}.$$

On the left-hand side of the above equality, after evaluating at $X = \zeta_\lambda \hat{Y}$, all terms of the sum become zero, except the λ-th term. Therefore, we have

$$\frac{\partial^\mu}{\partial X^\mu} \left(W_\lambda^{(\ell)} \frac{F^{(0)}}{G_\lambda^{(0)}} \right) \Big|_{X=\zeta_\lambda \hat{Y}} = \frac{\partial^\mu}{\partial X^\mu} (X^\ell \hat{Y}^{d-\ell})|_{X=\zeta_\lambda \hat{Y}}.$$

Also we have $W_\lambda^{(\ell)} = \sum_{j=0}^{m_\lambda-1} w_{\lambda,j}(\hat{Y}) X^j$, thus, we have

$$\sum_{j=0}^{m_\lambda-1} \frac{\partial^\mu}{\partial X^\mu} \left(X^j \frac{F^{(0)}}{G_\lambda^{(0)}} \right) \Big|_{X=\zeta_\lambda \hat{Y}} w_{\lambda,j}^{(\ell)} = \frac{\partial^\mu}{\partial X^\mu} (X^\ell \hat{Y}^{d-\ell}) \Big|_{X=\zeta_\lambda \hat{Y}}. \tag{3.8}$$

On the other hand, we know that

$$\frac{\partial^\mu}{\partial X^\mu} \left(\frac{F^{(0)}}{G_\lambda^{(0)}} \right) \Big|_{X=\zeta_\lambda \hat{Y}} = \frac{1}{m_\lambda!} \frac{\partial^{\mu+m_\lambda}}{\partial X^{\mu+m_\lambda}} (F^{(0)}) \Big|_{X=\zeta_\lambda \hat{Y}}.$$

Since $F^{(0)} \in \mathbf{k}[X, \hat{Y}]$, we have $\frac{\partial^\mu}{\partial X^\mu} \left(\frac{F^{(0)}}{G_\lambda^{(0)}} \right) \Big|_{X=\zeta_\lambda \hat{Y}} \in \mathbf{k}(\zeta_\lambda)[\hat{Y}]$. So, Equation (3.8) is a system of linear equations $\mathcal{W}_\lambda \mathcal{X}_\lambda^{(\ell)} = \mathcal{B}_\lambda^{(\ell)}$ in $\mathbf{k}(\zeta_\lambda)[\hat{Y}]$ (also see [82]) with coefficient matrix

$$\mathcal{W}_\lambda = [\alpha_{j,\mu}] \text{ with } \alpha_{j,\mu} = \frac{\partial^\mu}{\partial X^\mu} \left(X^j \frac{F^{(0)}}{G_\lambda^{(0)}} \right) \Big|_{X=\zeta_\lambda \hat{Y}}, \tag{3.9}$$

unknown vector $\mathcal{X}_\lambda^\ell = [w_{\lambda,j}^{(\ell)}]$ and constant vector

$$\mathcal{B}_\lambda^{(\ell)} = [\beta_\mu] \text{ with } \beta_\mu = \frac{\partial^\mu}{\partial X^\mu}(X^\ell \hat{Y}^{d-\ell})|_{X=\zeta_\lambda \hat{Y}} \tag{3.10}$$

for $j, \mu = 0, 1, \ldots, m_\lambda - 1$. The matrix $\mathcal{W}_\lambda$ is a Wronskian matrix. It is known that a Wronskian matrix is invertible whenever the functions in the first row are analytic and linearly independent, see [16]. In our case, the functions $\left(X^j \frac{F^{(0)}}{G_\lambda^{(0)}}\right)|_{X=\zeta_\lambda \hat{Y}}$, for $j = 0, 1, \ldots, m_\lambda - 1$, are, indeed, linearly independent polynomials in $\mathbf{k}(\zeta_\lambda)[\hat{Y}]$, therefore, the Wronskian matrix $\mathcal{W}_\lambda$ is invertible.

Now let us find the inverse of $\mathcal{W}_\lambda$. For simplicity of notations, let $f := \left(\frac{F^{(0)}}{G_\lambda^{(0)}}\right)|_{X=\zeta_\lambda \hat{Y}}$ and $f^{(\mu)} := \left(\frac{\partial^\mu}{\partial X^\mu}\frac{F^{(0)}}{G_\lambda^{(0)}}\right)\Big|_{X=\zeta_\lambda \hat{Y}}$ for $\mu = 1, \ldots, m_\lambda - 1$.

Proposition 1. *The inverse of $\mathcal{W}_\lambda$ is $\mathcal{W}_\lambda^{-1} = M_2 M_1$ where M_1 and M_2 are square matrices of order m_λ, defined as follows. The matrix M_1 writes $M_1 = M_{1(m_\lambda-1)} \cdots M_{11} M_{10}$ such that, for $j = 0, \cdots, m_\lambda - 1$, we have*

$$M_{1j} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \frac{1}{j!f} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \binom{j+1}{j}\frac{-f'}{f} & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \binom{m_\lambda-1}{j}\frac{-f^{(m_\lambda-1-j)}}{f} & 0 & \cdots & 1 \end{bmatrix}.$$

Hence, the matrix M_{1j} differs from the identity matrix only in its $(j+1)$-th column. The matrix M_2 is an upper triangular matrix $M_2 = [\gamma_{j,k}]$ with $\gamma_{j,k} = (-1)^{j+k}\binom{k}{k-j}\zeta_\lambda^{k-j}\hat{Y}^{k-j}$ if $j \le k$ and $\gamma_{j,k} = 0$ if $j > k$, for $j, k \in \{0, 1, \ldots, m_\lambda - 1\}$.

PROOF. To prove $\mathcal{W}_\lambda^{-1} = M_2 M_1$, it is enough to show that $M_2^{-1} = M_1 \mathcal{W}_\lambda$ holds, where M_2^{-1} is given by the next claim.

Claim: M_2^{-1} is upper triangular with $\binom{k}{k-j}\zeta_\lambda^{k-j}\hat{Y}^{k-j}$ as $(j,k)-$entry.

Proof of the claim: Let A be the upper triangular matrix with $\binom{k}{k-j}T^{k-j}$ as $(j,k)-$entry where T is a new variable. We show that $A|_{T=\zeta_\lambda \hat{Y}} \cdot M_2 = I$ where I is the identity matrix of order m_λ. Let us look at the dot product of the $(j+1)-$th row of A and the $(k+1)-$th column of M_2 where $k \ge j$. This dot product is:

$$\sum_{l=0}^{k-j}(-1)^{k+j+l}\binom{k}{k-j-l}T^l\binom{j+l}{l}\zeta_\lambda^{k-j-l}\hat{Y}^{k-j-l}.$$

The above quantity is also equal to each side of Equation (3.11):

$$\binom{k}{j}\sum_{l=0}^{k-j}(-1)^{k+j+l}\binom{k-j}{l}T^l\zeta_\lambda^{k-j-l}\hat{Y}^{k-j-l} = \binom{k}{j}(T-\zeta_\lambda\hat{Y})^{k-j}. \tag{3.11}$$

So for $k = j$, the right hand side of Equation (3.11) equals 1, and when $k \neq j$ (i.e. $k > j$), by evaluating $T = \zeta_\lambda \hat{Y}$, it is 0. Hence, we have $A|_{T=\zeta_\lambda \hat{Y}} \cdot M_2 = I$ and $M_2^{-1} = A|_{T=\zeta_\lambda \hat{Y}}$, proving the claim.

Now, it is enough to show that $M_2^{-1} = M_1 \cdot \mathcal{W}_\lambda$ holds. Observe that M_{1j} is the product of some elementary matrices (which are obtained by applying one elementary row operation on the identity matrix, like above matrices). Let $N_{j-1} := M_{1(j-1)} \cdots M_{10} \mathcal{W}_\lambda$. By multiplying M_{1j} by N_{j-1}, we are factoring out f from the $(j+1)-$th row and adding $-\binom{k}{j} f^{(k)}$ multiple of the $(j+1)-$th row to the $(j+k)-$th row for $k = 2, \ldots, m_\lambda - j - 1$. Therefore, the factor f will be removed from the $(j+1)-$th row. Furthermore, the term with highest derivative will also be removed from all rows after the $(j+1)-$th one. Hence, $M_{1(m_\lambda - 1)} \cdots M_{10} \mathcal{W}_\lambda$ is an upper triangular matrix such that every entry in the upper triangle is given by multiplying the term with lowest derivative of f by $1/(j!f)$. Since the $(j+1, k+1)-$entry of $\mathcal{W}_\lambda$ is $\frac{\partial^j}{\partial X^j}\left(X^k \frac{F^{(0)}}{G_\lambda^{(0)}}\right)$ at $X = \zeta_\lambda \hat{Y}$, the $(j+1, k+1)-$entry of $M_{1(m_\lambda-1)} \cdots M_{10} \mathcal{W}_\lambda$ is

$$\frac{1}{j!f} \frac{k!}{(k-j)!} \zeta_\lambda^{k-j} \hat{Y}^{k-j} f = \binom{k}{k-j} \zeta_\lambda^{k-j} \hat{Y}^{k-j},$$

which is exactly M_2^{-1}. This completes the proof. □

Lemma 2 yields the following for Yun-Moses polynomials.

Corollary 1. *If $F(X, Y) \in \mathbb{K}[X, Y]$, then $W_\lambda^{(\ell)} \in \mathbb{K}(\zeta_\lambda)\langle \hat{Y} \rangle[X]$, where ζ_λ is the root of the initial factor of $F^{(0)}$ corresponding to $W_\lambda^{(\ell)}$,*

Proof. From Lemma 2, we have $W_\lambda^\ell \in \mathbb{C}\langle \hat{Y} \rangle[X]$. Thus, it is enough to show that the coefficients of W_λ^ℓ are from $\mathbb{K}(\zeta_\lambda)$. First, observe that $F^{(0)}$ and $G_\lambda^{(0)}$ are two homogeneous polynomials of degrees $\sum_j m_j$ and m_λ in $\mathbb{K}[X, \hat{Y}]$ and $\mathbb{K}(\zeta_\lambda)[X, \hat{Y}]$, respectively. For any $\mu = 0, 1, \ldots, m_\lambda - 1$, we have

$$\frac{\partial^\mu}{\partial X^\mu}\left(\frac{F^{(0)}}{G_\lambda^{(0)}}\right)\Big|_{X=\zeta_\lambda \hat{Y}} \in \mathbf{k}(\zeta_\lambda)[\hat{Y}].$$

Hence, the coefficients of all entries of $\mathcal{W}_\lambda^{-1}$, defined in Proposition 1, live in $\mathbb{K}(\zeta_\lambda)$. Also, observe that the coefficients of all entries in matrix $\mathcal{B}_\lambda^{(\ell)}$ defined in (3.10) live in the same field $\mathbb{K}(\zeta_\lambda)$, therefore, $w_{\lambda,j} \in \mathbb{K}(\zeta_\lambda)\langle Y \rangle$, for all $j = 0, 1, \ldots, m_\lambda - 1$. Hence $W_\lambda^{(\ell)} \in \mathbb{K}(\zeta_\lambda)\langle Y \rangle[X]$. □

3.3.1 Computing the W_λ

In this section, we discuss how we compute the Yun-Moses polynomials W_λ. We regard each W_λ as a univariate polynomial in X, so we need to compute the coefficients of X^j for $j = 0, 1, \ldots, m_\lambda - 1$, which are univariate polynomials in $\hat{Y} = Y^{\hat{\delta}/\hat{d}}$. Therefore, we need to compute the inverse of the Wronskian matrix $\mathcal{W}_\lambda$ by computing $M_{10}, M_{11}, \ldots, M_{1(m_\lambda-1)}$ and M_2 at $X = \zeta_\lambda \hat{Y}$. Since $\frac{F^{(0)}}{G_\lambda^{(0)}}(X, \hat{Y})$ is a homogeneous polynomial, then f, as defined before Proposition 1, is just a term in $\hat{Y}$. Therefore, all entries of $\mathcal{W}_\lambda^{-1}$ are just terms in $\hat{Y}$; so to compute $\mathcal{W}_\lambda^{-1} = M_2 M_{10} \cdots M_{1(m_\lambda-1)}$, we just need to do arithmetic on the coefficients of $\hat{Y}$ and keep track of the degree of $\hat{Y}$ in each entry. We can observe that the degree of $\hat{Y}$ in the $(j, k)-$entry of $\mathcal{W}_\lambda^{-1}$ is $m_\lambda - d - j + k$ (see Section 3.3.2). On the other hand, the degree of $\hat{Y}$ in the $j-$th entry of $\mathcal{B}_\lambda^{(\ell)}$

is $d - j$ (see Section 3.3.2). Hence, computing the product $\mathcal{W}_\lambda^{-1} = M_2 M_{10} \cdots M_{1(m_\lambda - 1)}$ can be done as if those matrices had coefficients in $\mathbb{K}(\zeta_\lambda)$ rather than $\mathbb{K}(\zeta_\lambda)[\hat{Y}]$.

3.3.2 Complexity analysis

Let $f := F^{(0)}(X, \hat{Y})/(X - \zeta_\lambda \hat{Y})^{m_\lambda}$, where ζ_λ is a root of $F^{(0)}(X, 1)$, and let d_f be the degree of f w.r.t. X. So $d_f = d - m_\lambda$. We use the notations of Proposition 1. After evaluation at $X = \zeta_\lambda \hat{Y}$, in all entries of M_1 below the main diagonal, the degree of the denominator of the $(j, k)-$entry is $(j - k + 1)d_f$, the degree of the numerator is $(j - k)(d_f - 1)$ for $j, k = 1, \ldots, m_\lambda$, with $j \geq k$. Thus, in the $(j, k)-$entry, the degree of $\hat{Y}$ is $-(d - m_\lambda + j - k)$. In M_2, the degree of $\hat{Y}$ on the $(j, k)-$entry is $k - j$ for $j, k = 1, \ldots, m_\lambda$ with $k \geq j$. Hence, the $\hat{Y}$-degree in the $(j, k)-$entry of $\mathcal{W}_\lambda^{-1}$ is $2m_\lambda - d + k - j$.

In particular, $\mathsf{M}(n)$ is an upper bound for the number of operations in $\mathbb{K}(\zeta_\lambda)$ required for multiplying two univariate polynomials in $\mathbb{K}(\zeta_\lambda)$ with degree less than n. Let $\mathsf{A}(n)$ be an upper bound for the number of operations in $\mathbb{K}$ required by one addition or multiplication in a simple algebraic extension of $\mathbb{K}$ of degree n. We have: $\mathsf{A}(n) \in O(\mathsf{M}(n))$. Observe that the cost of evaluating f and its derivatives up to $f^{(m_\lambda - 1)}$ is negligible. Let C_1 be the cost of constructing the matrices $M_{10}, M_{11}, \ldots, M_{1(m_\lambda - 1)}$ and M_2. Assuming that $1/\zeta_\lambda$ and all involved binomial coefficients are precomputed, we have:

$$C_1 = \left(\frac{(m_\lambda - 1)m_\lambda}{2} + \sum_{j=0}^{m_\lambda - 1} m_\lambda - j \right) A(d).$$

The cost C_2 of multiplying $M_{10}, M_{11}, \ldots, M_{1(m_\lambda - 1)}$ and M_2 is:

$$C_2 = \left(\sum_{j=1}^{m_\lambda - 1} (m_\lambda - j)(2j - 1) + m_\lambda \sum_{j=1}^{m_\lambda} 2(j - 1) - 2 \sum_{k=1}^{m_\lambda} \sum_{j=1}^{k} j \right) A(d).$$

To understand where the factor $A(d)$ comes from, one should note that, if $F^{(0)}(X, 1)$ does not split into linear factors over $\mathbf{k}$, it is sufficient to work with its irreducible factors over $\mathbf{k}$, see Remark 1. Therefore, the cost C_{YM} of computing the Yun-Moses polynomials $W_\lambda^{(\ell)}$, for $\ell \in \{0, \ldots, d - 1\}$, is given by $C_{\mathrm{YM}} = C_1 + C_2 = O(m_\lambda^3 \mathsf{M}(d))$. This leads us to:

Theorem 3. *One can compute all the Yun-Moses polynomials $W_i^{(\ell)}$ ($0 \leq \ell \leq d - 1$, $1 \leq i \leq r$), within $O(d^3\, \mathsf{M}(d))$ operations in* $\mathbf{k}$.

PROOF. For constructing the matrices $M_{10}, M_{11}, \ldots, M_{1(m_\lambda - 1)}$, we need, respectively, $m_\lambda, m_\lambda - 1, \ldots, 1$ arithmetic calculations and therefore, we have $\left(\frac{(m_\lambda - 1)m_\lambda}{2}\right) A(d)$ as the total cost. Also for M_2, since it is an upper triangular matrix, it needs $\sum_{j=0}^{m_\lambda - 1} m_\lambda - j$ arithmetic computations. Thus the total cost for constructing the matrices $M_{10}, M_{11}, \ldots, M_{1(m_\lambda - 1)}$, and M_2 is

$$C_1 = \left(\frac{(m_\lambda - 1)m_\lambda}{2} + \sum_{j=0}^{m_\lambda - 1} m_\lambda - j \right) A(d).$$

For multiplying matrices $M_{10}, M_{11}, \ldots, M_{1(m_\lambda - 1)}$, due to sparsity of these matrices, there are $\sum_{j=1}^{m_\lambda - 1} (m_\lambda - j)(2j - 1)$ multiplications and additions required to compute $M_1 = M_{10} M_{11} \ldots M_{1(m_\lambda - 1)}$.

Row j of M_1 \ Column k of M_2	1	2	3	$\cdots$	k	$\cdots$	m_λ
1	$2\,m_\lambda$	$2\,(m_\lambda-1)$	$2\,(m_\lambda-2)$	$\cdots$	$2\,(m_\lambda-k-1)$	$\cdots$	2
2	$2\,(m_\lambda-1)$	$2\,(m_\lambda-1)$	$2\,(m_\lambda-2)$	$\cdots$	$2\,(m_\lambda-k-2)$	$\cdots$	2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
j	$2\,(m_\lambda-j+1)$	$2\,(m_\lambda-j+1)$	$2\,(m_\lambda-j+1)$	$\cdots$	$2\,(m_\lambda-k-j)$	$\cdots$	2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m_λ	2	2	2	$\cdots$	2	$\cdots$	2

Table 3.1: Calculation of the total number of multiplications and additions for multiplying each row of M_1 by each column of M_2.

Next, we multiply M_2 and M_1. As we know, the matrices M_2 and M_1 are, respectively, upper and lower triangular matrices. Here we suppose that the multiplication of a number and zero is negligible. The number of multiplications and additions for multiplying each row of M_2 by each column of M_1 is listed in Table 3.1. Thus the total number of the arithmetic computations for computing M_2M_1 is

$$\left(m_\lambda \sum_{j=1}^{m_\lambda} 2(j-1) - 2\sum_{k=1}^{m_\lambda}\sum_{j=1}^{k} j\right)A(d).$$

Thus the total cost of arithmetic computations for building $M_1 = M_{10}M_{11}\cdots M_{1(m_\lambda-1)}$ and multiplying M_2 and M_1 is

$$C_2 = \left(\sum_{j=1}^{m_\lambda-1}(m_\lambda-j)(2j-1) + m_\lambda\sum_{j=1}^{m_\lambda} 2(j-1) - 2\sum_{k=1}^{m_\lambda}\sum_{j=1}^{k} j\right)A(d).$$

Thus the cost of computing Yun-Moses polynomials $W_\lambda^{(\ell)}$, for $\ell = 0,\ldots,d-1$, is $O(m_\lambda^3 M(d))$. Thus the total cost for computing all the Yun-Moses polynomials $W_i^{(}\ell)$, for $\ell = 0,\ldots,d-1$, and for $i = 1,\ldots,r$, is $O(d^3\,\mathsf{M}(d))$. □

3.4 Lifting the factors

We turn our attention to the lifting of the factors during the EHC, Lines 8-12 in Algorithm 1. A naive implementation of that step would make the running-time of the i-iteration growing quadratically with i. Adapting and enhancing an idea of L. Bernardin in [15], we make this running-time in $\Theta(i)$ instead of $\Theta(i^2)$.

Let $\tilde{Y} = Y^{1/\hat{d}}$. Let Δ_i^k be such that $\Delta G_i^{(k)} = \Delta_i^k \tilde{Y}^k$ and define $\Delta_i^0 = G_i^{(0)}$. Therefore, Δ_i^k, for $k > 0$, is homogeneous with respect to $(X, \hat{Y})$ of degree m_i and we can write

$$G_i^{(k)} = \Delta_i^0 + \Delta_i^1\tilde{Y} + \Delta_i^2\tilde{Y}^2 + \cdots + \Delta_i^k\tilde{Y}^k.$$

While Bernardin in [15] discusses his "recycling" strategy for univariate polynomials with constant coefficients, we enhance his idea for the bivariate polynomials $G_i^{(k)}$.

For $j = 2, \ldots, r$ and $k \geq 1$, we let P_j^k be a degree k univariate polynomial in $\tilde{Y}$ satisfying $P_j^k \equiv G_1^{(k-1)} \cdots G_j^{(k-1)} \bmod S_{k+1}$. So, initially, we have $P_j^1 \equiv G_1^{(0)} \cdots G_j^{(0)} \bmod S_2$, for $j = 2, \ldots, r$. For $j = 2$ and $k > 1$ we have

$$\begin{array}{rcll} P_2^k & \equiv & G_1^{(k-1)} G_2^{(k-1)} & \bmod S_{k+1}, \text{ so} \\ P_2^k & = & \Delta_1^0 \Delta_2^0 & + \left(\Delta_1^0 \Delta_2^1 + \Delta_2^0 \Delta_1^1\right) \tilde{Y} + \cdots \\ & & & + \left(\Delta_1^0 \Delta_2^{k-1} + \cdots + \Delta_2^0 \Delta_1^{k-1}\right) \tilde{Y}^{k-1} \\ & & & + \left(\Delta_1^1 \Delta_2^{k-1} + \cdots + \Delta_2^1 \Delta_1^{k-1}\right) \tilde{Y}^k. \end{array}$$

For the next iteration, that is from k to $k + 1$, we have:

$$\begin{array}{rcll} P_2^{k+1} & \equiv & G_1^{(k)} G_2^{(k)} & \bmod S_{k+2}, \text{ so} \\ P_2^{k+1} & = & \Delta_1^0 \Delta_2^0 & + \left(\Delta_1^0 \Delta_2^1 + \Delta_2^0 \Delta_1^1\right) \tilde{Y} + \cdots \\ & & & + \left(\Delta_1^0 \Delta_2^{k-1} + \cdots + \Delta_2^0 \Delta_1^{k-1}\right) \tilde{Y}^{k-1} \\ & & & + \left(\Delta_1^0 \Delta_2^k + \cdots + \Delta_2^0 \Delta_1^k\right) \tilde{Y}^k \\ & & & + \left(\Delta_1^1 \Delta_2^k + \cdots + \Delta_2^1 \Delta_1^k\right) \tilde{Y}^{k+1}. \end{array}$$

If we assume that P_2^k has been computed and stored at the previous iteration, then it is enough to compute $\Delta_1^0 \Delta_2^k$, $\Delta_2^0 \Delta_1^k$ and $\Delta_1^1 \Delta_2^k + \cdots + \Delta_2^1 \Delta_1^k$ in the current iteration in order to deduce P_2^{k+1}, with the following recursive formula:

$$P_2^{k+1} = P_2^k + (\Delta_1^0 \Delta_2^k + \Delta_1^k \Delta_2^0) \tilde{Y}^k + (\Delta_1^1 \Delta_2^k + \cdots + \Delta_1^k \Delta_2^1) \tilde{Y}^{k+1}.$$

Now for $j = 3, \ldots, r$, define

$$\begin{array}{rcll} P_j^k & \equiv & P_{j-1}^k G_j^{(k-1)} & \bmod S_{k+1}, \text{ so} \\ P_j^k & = & p_{j-1}^{k,0} \Delta_j^0 & + \left(p_{j-1}^{k,1} \Delta_j^0 + p_{j-1}^{k,0} \Delta_j^1\right) \tilde{Y} + \cdots \\ & & & + \left(p_{j-1}^{k,0} \Delta_j^{k-1} + \cdots + p_{j-1}^{k,k-1} \Delta_j^0\right) \tilde{Y}^{k-1} \\ & & & + \left(p_{j-1}^{k,1} \Delta_j^{k-1} + \cdots + p_{j-1}^{k,k} \Delta_j^0\right) \tilde{Y}^k, \end{array}$$

where $P_{j-1}^k = p_{j-1}^{k,0} + p_{j-1}^{k,1} \tilde{Y} + \cdots + p_{j-1}^{k,k} \tilde{Y}^k$. Hence, we deduce:

$$\begin{array}{rcll} P_j^{k+1} & = & P_{j-1}^{k+1} G_j^{(k)} & \bmod S_{k+2}, \text{ so} \\ P_j^{k+1} & = & p_{j-1}^{k+1,0} \Delta_j^0 & + \left(p_{j-1}^{k+1,1} \Delta_j^0 + p_{j-1}^{k+1,0} \Delta_j^1\right) \tilde{Y} + \cdots \\ & & & + \left(p_{j-1}^{k+1,0} \Delta_j^{k-1} + \cdots + p_{j-1}^{k+1,k-1} \Delta_j^0\right) \tilde{Y}^{k-1} \\ & & & + \left(p_{j-1}^{k+1,0} \Delta_j^k + \cdots + p_{j-1}^{k+1,k} \Delta_j^0\right) \tilde{Y}^k \\ & & & + \left(p_{j-1}^{k+1,1} \Delta_j^k + \cdots + p_{j-1}^{k+1,k+1} \Delta_j^0\right) \tilde{Y}^{k+1}. \end{array}$$

If we assume that P_j^k and P_{j-1}^k have been computed and stored at the previous iteration, then we can recycle some of the terms of P_j^k and P_{j-1}^k in support of the calculation of P_j^{k+1}. However, there are definitely new terms in P_j^{k+1} that we need to compute in the current iteration, namely $p_{j-1}^{k+1,0} \Delta_j^k$ and $p_{j-1}^{k+1,1} \Delta_j^k + \cdots + p_{j-1}^{k+1,k+1} \Delta_j^0$.

Observe that $p_{j-1}^{k+1,i} = p_{j-1}^{k,i}$ holds for $i = 0, 1, \ldots, k-1$, while $p_{j-1}^{k+1,k} = p_{j-1}^{k,k} + q_j^{k+1}$ holds, where q_j^{k+1} is recursively given by

$$q_j^{k+1} = p_{j-1}^{k+1,0}\Delta_j^k + q_{j-1}^{k+1}\Delta_j^0 \text{ with } q_2^{k+1} = \Delta_2^k\Delta_1^0 + \Delta_2^0\Delta_1^k. \tag{3.12}$$

Now observe that we have

$$\begin{aligned} p_j^{k+1,k} &= p_{j-1}^{k+1,0}\Delta_j^k + \cdots + p_{j-1}^{k+1,k}\Delta_j^0 \\ &= p_{j-1}^{k+1,0}\Delta_j^k + p_{j-1}^{k,1}\Delta_j^{k-1} + \cdots + p_{j-1}^{k,k-1}\Delta_j^1 + (p_{j-1}^{k,k} + q_j^{k+1})\Delta_j^0 \\ &= p_{j-1}^{k+1,0}\Delta_j^k + p_j^{k,k} + q_{j-1}^{k+1}\Delta_j^0 = p_j^{k,k} + q_j^{k+1}. \end{aligned}$$

Therefore, we can write

$$P_j^{k+1} = P_j^k + q_j^{k+1}\tilde{Y}^k + \left(p_{j-1}^{k+1,1}\Delta_j^k + \cdots + p_{j-1}^{k+1,k+1}\Delta_j^0\right)\tilde{Y}^{k+1}. \tag{3.13}$$

Note: the term q_j^{k+1} is missing in the formula at the top of the left column on p. 3 of [15].

3.4.1 Complexity analysis

It follows from Equation (3.13) that each P_j^ℓ, for $0 \le \ell \le k+1$, is derived from $P_j^{\ell-1}$ and P_{j-1}^ℓ *in a Pascal Triangle fashion*. More precisely, letting $\ell = k+1$, if P_j^k and P_{j-1}^{k+1} are known, computing P_j^{k+1} requires 2 multiplications for computing q_j^{k+1} (see Equations (3.12) and (3.13)) and k multiplications for the new terms (see Equation(3.13)). All multiplications are product of a polynomial of degree m_j to a polynomial of degree $m_1 + \cdots + m_{j-1}$. Also, all P_j^ℓ for $j = 1, \ldots, r$ and $\ell = 1, \ldots, k$ need to be computed before computing P_r^{k+1}. Let C_{lift} be the cost of computing P_r^{k+1}. We have: $C_{\text{lift}} = \sum_{l=2}^{r}(k+2)\mathsf{M}(\max(m_1 + \cdots + m_{l-1}, m_l))\mathsf{A}(d)$. This leads us to the following result.

Theorem 4. *The k-th iteration of Step 9 in the Algorithm 1 runs in $O(k\,d\mathsf{M}(d)^2)$ operations in* **k**.

3.5 Experimentation

Table 3.2 gathers running times for comparing the EHC and Kung-Traub's method for $k = 10$ and $k = 20$, where k is as in Section 3.1. The columns `KT Lin` and `KT Quad` correspond to linear and quadratic lifting methods of Kung and Traub, respectively. Thus, for the EHC, which is based on a linear lifting, as well as for `KT Lin`, $k = 10$ and $k = 20$ means 10 and 20 iterations of the "main loop". For `KT Quad`, $k = 10$ and $k = 20$ means 4 and 5 iterations of the "main loop".

Each test-example has a number and can be found from `wwww.regularchains.org/papers/Benchmark-ISSAC-2017.zip`. The column MD gives the degree of the main variable in the input polynomial. The columns KT10 and KT20 correspond to $k = 10$ and $k = 20$. The sub-columns EHC10 and EHC20 under EHCWM, give the timings for our enhanced EHC, described in this chapter, that is, based on Sections 3.3 and 3.4. The sub-column EHC10, under EHCEEA, gives the timings for an implementation of the original EHC method as described

in [82]. The sub-columns YM1 and YM2 show the timings for computing the Yun-Moses polynomials corresponding to EHC10, respectively for EHCWM and EHCEEA.

In Table 3.2, the three most significant digits of the timings are recorded and ∞ means the computations exceeded either the time limit of 3600sec, or the memory limit of 48Gb. These experimental results were obtained on an Ubuntu desktop (1.6GHz Intel(R) Xeon(R) CPU).

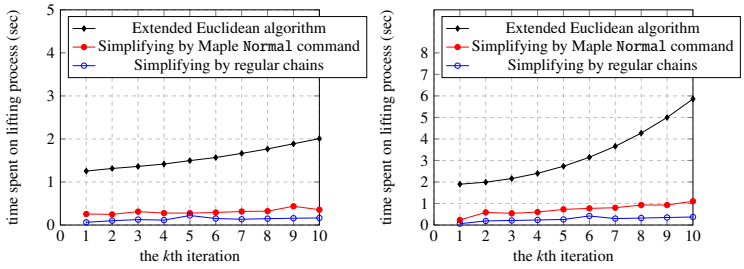


Figure 3.3: For each k on the x-axis, these plots show the time spent for lifting the factors of EHC, from step $k-1$ to step k see Lines 8-12 in Algorithm 1: (1) the black curve corresponds to the original EHC [82]; (2) The red curve corresponds to the implementation of EHC with the optimization tricks presented in the current chapter, when the simplifications of algebraic numbers are done with the `Normal` command of MAPLE, and (3) the blue curve is the timing of EHC with the optimization tricks when the simplifications of algebraic numbers are done with the `RegularChains` library.

Figure 3.3 focuses on the performance of the optimization tricks applied on the lifting process of EHC as explained in this chapter, for two different bivariate polynomials. Note that square-root scaling has been used for the y-axis. The EHC algorithm, as well as Kung-Traub's method, are implemented in MAPLE and they are integrated into `PowerSeries` library. The `PowerSeries` library is available at `www.regularchains.org`.

Ex	MD	KT Lin		KT Quad		EHCWM			EHCEEA	
		KT10	KT20	KT10	KT20	EHC10	YM1	EHC20	EHC10	YM2
1	5	2.22	18.6	4.93	4.91	0.48	0.22	0.73	0.90	0.21
7	4	5.60	65.8	0.56	0.58	0.22	0.14	0.23	0.34	0.13
8	4	14.9	230	1.25	1.25	0.23	0.13	0.28	0.36	0.12
9	3	5.53	114	1.51	1.56	0.30	0.11	0.39	0.88	0.10
10	3	2.71	42.0	0.28	0.63	0.16	0.08	0.20	0.32	0.12
11	3	0.46	2.34	0.21	0.21	0.16	0.08	0.17	0.26	0.12
12	3	0.50	6.86	0.28	0.32	0.16	0.08	0.18	0.30	0.12
13	4	0.86	10.9	0.50	0.48	0.26	0.15	0.28	0.46	0.24
14	4	3.21	34.8	0.69	0.71	0.26	0.15	0.34	0.52	0.24
15	6	27.6	535	4.85	4.85	0.64	0.42	0.82	2.05	1.08
16	7	45.6	836	8.45	9.91	0.64	0.43	0.92	2.33	1.74
17	7	145	∞	23.4	23.2	0.78	0.43	3.37	4.12	1.77
19	4	0.14	0.16	0.16	0.14	0.39	0.26	0.45	0.51	0.15
20	4	2.79	7.98	0.77	0.82	0.26	0.15	0.29	0.50	0.24
21	4	8.58	143	1.96	1.93	0.23	0.12	0.31	0.47	0.16
24	5	2.90	24.8	1.11	1.11	0.26	0.15	0.35	0.49	0.17
25	7	1.83	9.45	0.90	1.00	0.46	0.31	0.50	0.73	0.42
26	8	2.35	12.3	3.09	3.29	0.66	0.53	0.74	2.18	1.80
27	8	60.8	2876	23.1	27.1	0.77	0.53	1.20	2.31	1.28
28	9	215	∞	73.8	123	1.88	1.03	2.11	7.03	4.92
30	17	∞	∞	∞	∞	39.8	6.70	41.3	53.8	16.5
31	32	∞	∞	∞	∞	599	24.9	∞	∞	∞
32	33	∞	∞	∞	∞	224	25.0	∞	∞	∞

Table 3.2: Comparing EHC versus Kung-Traub's method (timings are in seconds)

Chapter 4

Computing Limit Points via Puiseux Series Expansions

4.1 Introduction

The theory of regular chains, since its introduction by J.F. Ritt [77], has been applied successfully in many areas including differential systems [27, 17, 45], difference systems [37], unmixed decompositions [49] and primary decomposition [91] of polynomial ideals, intersection multiplicity calculations [62], cylindrical algebraic decomposition [26], parametric [105] and non-parametric [22] semi-algebraic systems. Today, regular chains are at the core of algorithms computing triangular decomposition of polynomial systems and which are available in several software packages [58, 102, 103]. Moreover, those algorithms provide back-engines for computer algebra system front-end solvers, such as MAPLE's `solve` command.

One of the algorithmic strengths of the theory of regular chains is its *regularity test* procedure. Given a polynomial p and a regular chain R, both in a multivariate polynomial ring $\mathbf{k}[X_1, \ldots, X_n]$ over a field $\mathbf{k}$, this procedure computes regular chains $R_1, \ldots, R_e$ such that $R_1, \ldots, R_e$ is a decomposition of R in some technical sense [1] and for each $1 \leq i \leq e$ the polynomial p is either null or regular modulo the saturated ideal of R_i. Thanks to the D5 Principle [32], this regularity test avoids factorization into irreducible polynomials and involves only polynomial GCD and resultant computations.

One of the technical difficulties of this theory, however, is the fact that regular chains do not fit well in the "usual algebraic-geometric dictionary" (Chapter 4, [29]). Indeed, the "good" zero set encoded by a regular chain R is a constructible set $W(R)$, called the *quasi-component* of R, which does not correspond exactly to the "good" ideal encoded by R, namely $\mathrm{sat}(R)$, the *saturated ideal* of R. In fact, the affine variety defined by $\mathrm{sat}(R)$ equals $\overline{W(R)}$, that is, the Zariski closure of $W(R)$.

For this reason, a decomposition algorithm, such as the one of M. Kalkbrener [49] (which, for an input polynomial ideal $\mathcal{I}$ computes regular chains $R_1, \ldots, R_e$ such that $\sqrt{\mathcal{I}}$ equals the intersection of the radicals of the saturated ideals of $R_1, \ldots, R_e$) can not be seen as a decomposition algorithm for the variety $V(\mathcal{I})$. Indeed, the output of Kalkbrener's algorithm yields

[1] The radical of the saturated ideal of R is equal to the intersection of the radicals of the saturated ideals of $R_1, \ldots, R_e$.

$V(\mathcal{I}) = \overline{W(R_1)} \cup \cdots \cup \overline{W(R_e)}$ while a decomposition of the form $V(\mathcal{I}) = W(R_1) \cup \cdots \cup W(R_f)$ would be more explicit.

Kalkbrener's decompositions, and in fact all decompositions of differential ideals [27, 17, 45] raise another notorious issue: the *Ritt problem*, stated as follows. Given two regular chains (algebraic or differential) R and S, check whether the inclusion of saturated ideals $\mathrm{sat}(R) \subseteq \mathrm{sat}(S)$ holds or not. In the algebraic case, this inclusion can be tested by computing a set of generators of $\mathrm{sat}(R)$, using Gröbner bases. In practice, this solution is too expensive for the purpose of removing redundant components in Kalkbrener's decompositions and only some criteria are applied [57]. In the differential case , there has not even an algorithmic solution.

In the algebraic case, both issues would be resolved if one would have a practically efficient procedure with the following specification: for the regular chain R compute regular chains $R_1, \ldots, R_e$ such that we have $\overline{W(R)} = W(R_1) \cup \cdots \cup W(R_e)$. If in addition, such procedure does not require a system of generators of $\mathrm{sat}(R)$, this might suggest a solution in the differential version of the Ritt problem.

We propose a solution to this algorithmic quest, in the algebraic case. To be precise, our procedure computes the *non-trivial limit points* of the quasi-component $W(R)$, that is, the set $\lim(W(R)) := \overline{W(R)} \setminus W(R)$ as a finite union of quasi-components of some other regular chains, see Theorem 11 in Section 4.7. This turns out to be $\overline{W(R)} \cap V(h_R)$, where $V(h_R)$ is the hypersurface defined by the product of the initials of polynomials in R. We focus on the case where $\mathrm{sat}(R)$ has dimension one. The case $\mathrm{sat}(R)$ has dimension higher than one is discussed in Chapter 6.

When the regular chain R consists of a single polynomial r, primitive w.r.t. its main variable, one can easily check that $\lim(W(R)) = V(r, h_r)$ holds, where h_r is the initial of r. Unfortunately, there is no generalization of this result when R consists of several polynomials, unless R enjoys remarkable properties, such as being a *primitive regular chain* [57]. To overcome this difficulty, it becomes necessary to view R as a "parametric representation" of the quasi-component $W(R)$. In this setting, the points of $\lim(W(R))$ can be computed as limits (in the usual sense of the Euclidean topology [2]) of sequences of points along "branches" (in the sense of the theory of algebraic curves) of $W(R)$. It turns out that these limits can be obtained as constant terms of convergent Puiseux series defining the "branches" of $W(R)$ in the neighborhood of the points of interest.

Here comes the main technical difficulty of this approach. When computing a particular point of $\lim(W(R))$, one needs to follow one branch per defining equation of R. Following a branch means computing a truncated Puiseux expansion about a point. Since the equation of R defining a given variable, say X_j, depends on the equations of R defining the variables $X_{j-1}, X_{j-2}, \ldots$, the truncated Puiseux expansion for X_j is defined by an equation whose coefficients involve the truncated Puiseux expansions for $X_{j-1}, X_{j-2}, \ldots$.

From Sections 4.3 to 4.7, we show that this principle indeed computes the desired limit points. In particular, we introduce the notion of a *system of Puiseux parametrizations of a regular chain*, see Section 4.3. This allows us to state in Theorem 7 a concise formula for $\lim(W(R))$ in terms of this latter notion. Then, we estimate to which accuracy one needs to effectively compute such a system of Puiseux parametrizations in order to deduce $\lim(W(R))$, see Theorem 10 in Section 4.6.

[2] The closures of $W(R)$ in Zariski and the Euclidean topologies are equal when $\mathbf{k} = \mathbb{C}$.

In Section 4.8, we report on a preliminary implementation of the algorithms presented in this chapter. We evaluate our code by applying it to the question of removing redundant components in Kalkbrener's decompositions and observe the benefits of this strategy.

In order to facilitate the presentation of those technical materials, we dedicate Section 4.2.1 to the case of regular chains in 3 variables. Section 4.2 briefly reviews notions from the theories of regular chains and algebraic curves. We conclude this introduction with an example.

Consider the regular chain $R = \{r_1, r_2\} \subset \mathbf{k}[X_1, X_2, X_3]$ with $r_1 = X_1X_2^2 + X_2 + 1, r_2 = (X_1 + 2)X_1X_3^2 + (X_2 + 1)(X_3 + 1)$. We have $W(R) = V(R) \setminus V(h_R)$ with $h_R = X_1^2(X_1 + 2)$. To determine $\lim(W(R))$, we compute Puiseux series expansions of r_1 at $X_1 = 0$ and $X_1 = -2$. For such calculation, we use MAPLE's command algcurves[puiseux] [95]. We start with $X_1 = 0$. The Puiseux expansions of r_1 at $X_1 = 0$ are:

$$[X_1 = T, X_2 = \tfrac{-T^2-T}{T} + O(T^2)],$$
$$[X_1 = T, X_2 = \tfrac{-1+T^2+T}{T} + O(T^2)].$$

Clearly, the second expansion does not yield a limit point. After substituting the first expansion into r_2, we have:

$$\begin{aligned} r_2' &= r_2(X_1 = T, X_2 = \tfrac{-T^2-T}{T} + O(T^2), X_3) \\ &= (T + 2)TX_3^2 + (-T + O(T^2))(X_3 + 1) \end{aligned}$$

Now, we compute Puiseux series expansions of r_2' which are

$$[T = T, X_3 = 1 - 1/3T + O(T^2)],$$
$$[T = T, X_3 = -1/2 + 1/12\,T + O(T^2)].$$

So the regular chains $\{X_1, X_2 + 1, X_3 - 1\}$ and $\{X_1, X_2 + 1, X_3 + 1/2\}$ give the limit points of $W(R)$ at $X_1 = 0$.

Next, we consider $X_1 = -2$. We compute Puiseux series expansions of r_1 at the point $X_1 = -2$. We have:

$$[X_1 = T - 2, X_2 = 1 + 1/3\,T + O(T^2)],$$
$$[X_1 = T - 2, X_2 = -1/2 - 1/12\,T + O(T^2)].$$

After substitution into r_2, we obtain:

$$\begin{aligned} r_{12}' &= r_2(X_1 = T - 2, X_2 = 1 + 1/3\,T + O(T^2), X_3) \\ &= (T - 2)\,T{X_3}^2 + \left(2 + 1/3\,T + O(T^2)\right)(X_3 + 1) \\ r_{22}' &= r_2([X_1 = T - 2, X_2 = -1/2 - 1/12\,T + O(T^2)) \\ &= (T - 2)\,T{X_3}^2 + \left(1/2 - 1/12\,T + O(T^2)\right)(X_3 + 1)\,. \end{aligned}$$

So those Puiseux expansions of r_{12}' and r_{22}' at $T = 0$ which result in a limit point are as follows:

i) for r_{12}': $[T = T, X_3 = \frac{T^2-T}{T} + O(T^2)]$

ii) for r_{22}': $[T = T, X_3 = \frac{4\,T^2-T}{T} + O(T^2)]$

Thus, the limit points of R at the point $X_1 = -2$ can be represented by the regular chains $\{X_1+2, X_2-1, X_3+1\}$ and $\{X_1+2, X_2+1/2, X_3+1\}$. One can check that a triangular decomposition of the system $R \cup \{X_1\}$ is $\{X_2 + 1, X_1\}$ and, thus, does not yield $\lim(W(R)) \cap V(X_1)$, but in fact a superset of it.

4.2 Preliminaries

This section is a review of various notions from the theories of regular chains, algebraic curves and topology. For these latter subjects, our references are the textbooks of R.J. Walker [101], G. Fischer [35] and J. R. Munkres [71]. Notations and hypotheses introduced in this section are used throughout this chapter.

Multivariate polynomials. Let $\mathbf{k}$ be a field which is algebraically closed. Let $X_1 < \cdots < X_s$ be $s \geq 1$ ordered variables. We denote by $\mathbf{k}[X_1, \ldots, X_s]$ the ring of polynomials in the variables $X_1, \ldots, X_s$ and with coefficients in $\mathbf{k}$. For a non-constant polynomial $p \in \mathbf{k}[X_1, \ldots, X_s]$, the greatest variable in p is called *main variable* of p, denoted by $\mathrm{mvar}(p)$, and the leading coefficient of p w.r.t. $\mathrm{mvar}(p)$ is called *initial* of p, denoted by $\mathrm{init}(p)$.

Zariski topology. We denote by $\mathbb{A}^s$ the *affine s-space* over $\mathbf{k}$. An *affine variety* of $\mathbb{A}^s$ is the set of common zeroes of a collection $F \subseteq \mathbf{k}[X_1, \ldots, X_s]$ of polynomials. The *Zariski topology* on $\mathbb{A}^s$ is the topology whose closed sets are the affine varieties of $\mathbb{A}^s$. The *Zariski closure* of a subset $W \subseteq \mathbb{A}^s$ is the intersection of all affine varieties containing W. This is also the set of common zeroes of the polynomials in $\mathbf{k}[X_1, \ldots, X_s]$ vanishing at any point of W.

Relation between Zariski topology and the Euclidean topology. When $\mathbf{k} = \mathbb{C}$, the affine space $\mathbb{A}^s$ is endowed with both Zariski topology and the Euclidean topology. The basic open sets of the Euclidean topology are the balls while the basic open sets of Zariski topology are the complements of hypersurfaces. A Zariski closed (resp. open) set is closed (resp. open) in the Euclidean topology on $\mathbb{A}^s$. The following properties emphasize the fact that Zariski topology is coarser than the Euclidean topology: every nonempty Euclidean open set is Zariski dense and every nonempty Zariski open set is dense in the Euclidean topology on $\mathbb{A}^s$. However, the closures of a constructible set in Zariski topology and the Euclidean topology are equal. More formally, we have the following (Corollary 1 in I.10 of [70]) key result. Let $V \subseteq \mathbb{A}^s$ be an irreducible affine variety and $U \subseteq V$ be open in the Zariski topology induced on V. Then, the closure of U in Zariski topology and the closure of U in the Euclidean topology are both equal to V.

Limit points. Let (X, τ) be a topological space. A point $p \in X$ is a *limit* of a sequence $(x_n, n \in \mathbb{N})$ of points of X if, for every neighborhood U of p, there exists an N such that, for every $n \geq N$, we have $x_n \in U$; when this holds we write $\lim_{n\to\infty} x_n = p$. If X is a Hausdorff space then limits of sequences are unique, when they exist. Let $S \subseteq X$ be a subset. A point $p \in X$ is a *limit point* of S if every neighborhood of p contains at least one point of S different from p itself. Equivalently, p is a limit point of S if it is in the closure of $S \setminus \{p\}$. In addition, the closure of S is equal to the union of S and the set of its limit points. If the space X is sequential, and in particular if X is a metric space, the point p is a limit point of S if and only if there exists a sequence $(x_n, n \in \mathbb{N})$ of points of $S \setminus \{p\}$ with p as limit. In practice, the "interesting" limit points of S are those which do not belong to S. For this reason, we call such limit points *non-trivial* and we denote by $\lim(S)$ the set of non-trivial limit points of S.

Regular chain. A set R of non-constant polynomials in $\mathbf{k}[X_1, \ldots, X_s]$ is called a *triangular set*, if for all $p, q \in R$ with $p \neq q$ we have $\mathrm{mvar}(p) \neq \mathrm{mvar}(q)$. A variable X_i is said *free* w.r.t. R if there exists no $p \in R$ such that $\mathrm{mvar}(p) = X_i$. For a nonempty triangular set R, we define the *saturated ideal* $\mathrm{sat}(R)$ of R to be the ideal $\langle R \rangle : h_R^\infty$, where h_R is the product of the initials of the polynomials in R. The saturated ideal of the empty triangular set is defined as

the trivial ideal $\langle 0 \rangle$. The ideal $\mathrm{sat}(R)$ has several properties, in particular it is unmixed [19]. We denote its height, that is the number of polynomials in R, by e, thus $\mathrm{sat}(R)$ has dimension $s - e$. Let $X_{i_1} < \cdots < X_{i_e}$ be the main variables of the polynomials in R. We denote by r_j the polynomial of R whose main variable is X_{i_j} and by h_j the initial of r_j. Thus h_R is the product $h_1 \cdots h_e$. We say that R is a *regular chain* whenever R is empty or $\{r_1, \ldots, r_{e-1}\}$ is a regular chain and h_e is regular modulo the saturated ideal $\mathrm{sat}(\{r_1, \ldots, r_{e-1}\})$. The regular chain R is said *strongly normalized* whenever each of the main variables of the polynomials of R (that is, $X_{i_1} < \cdots < X_{i_e}$) does not appear in h_R.

Limit points of the quasi-component of a regular chain. We denote by $W(R) := V(R) \setminus V(h_R)$ the *quasi-component* of R, that is, the common zeros of R that do not cancel h_R. The above discussion implies that the closure of $W(R)$ in Zariski topology and the closure of $W(R)$ in the Euclidean topology are both equal to $V(\mathrm{sat}(R))$, that is, the affine variety of $\mathrm{sat}(R)$. We denote by $\overline{W(R)}$ this common closure and $\lim(W(R))$ this common set of limit points.

Rings of formal power series. Recall that $\mathbf{k}$ is an algebraically closed field. From now on, we further assume that $\mathbf{k}$ is topologically complete. Hence $\mathbf{k}$ may be the field $\mathbb{C}$ of complex numbers but not the algebraic closure of the field $\mathbb{Q}$ of rational numbers. We denote by $\mathbf{k}[[X_1, \ldots, X_s]]$ and $\mathbf{k}\langle X_1, \ldots, X_s \rangle$ the rings of formal and convergent power series in $X_1, \ldots, X_s$ with coefficients in $\mathbf{k}$. Note that the ring $\mathbf{k}\langle X_1, \ldots, X_s \rangle$ is a subring of $\mathbf{k}[[X_1, \ldots, X_s]]$. When $s = 1$, we write T instead of X_1. Thus $\mathbf{k}[[T]]$ and $\mathbf{k}\langle T \rangle$ are the rings of formal and convergent univariate power series in T and coefficients in $\mathbf{k}$. For $f \in \mathbf{k}[[X_1, \ldots, X_s]]$, its *order* is defined by

$$\mathrm{ord}(f) = \begin{cases} \min\{d \mid f_{(d)} \neq 0\} & \text{if } f \neq 0, \\ \infty & \text{if } f = 0. \end{cases}$$

where $f_{(d)}$ is the *homogeneous part* of f in degree d. Recall that $\mathbf{k}[[X_1, \ldots, X_s]]$ is topologically complete for Krull Topology and that $\mathbf{k}\langle X_1, \ldots, X_s \rangle$ is a Banach Algebra for the norm defined by $\| f \|_\rho = \Sigma_e |a_e| \rho^e$ where $f = \Sigma_e a_e X^e \in \mathbf{k}[[X_1, \ldots, X_s]]$ and $\rho = (\rho_1, \ldots, \rho_s) \in \mathbb{R}^s_{>0}$. We denote by $\mathcal{M}_s$ the only maximal ideal of $\mathbf{k}[[X_1, \ldots, X_s]]$, that is,

$$\mathcal{M}_s = \{f \in \mathbf{k}[[X_1, \ldots, X_s]] \mid \mathrm{ord}(f) \geq 1\}.$$

Let $f \in \mathbf{k}[[X_1, \ldots, X_s]]$ with $f \neq 0$. Let $k \in \mathbb{N}$. We say that f is (1) *general* in X_s if $f \neq 0 \mod \mathcal{M}_{s-1}$, (2) *general* in X_s of order k if we have $\mathrm{ord}(f \mod \mathcal{M}_{s-1}) = k$.

Formal Puiseux series. We denote by $\mathbf{k}[[T^*]] = \bigcup_{n=1}^{\infty} \mathbf{k}[[T^{\frac{1}{n}}]]$ the ring of *formal Puiseux series*. For a fixed $\varphi \in \mathbf{k}[[T^*]]$, there is an $n \in \mathbb{N}_{>0}$ such that $\varphi \in \mathbf{k}[[T^{\frac{1}{n}}]]$. Hence $\varphi = \sum_{m=0}^{\infty} a_m T^{\frac{m}{n}}$, where $a_m \in \mathbf{k}$. We call *order of* φ the rational number defined by $\mathrm{ord}(\varphi) = \min\{\frac{m}{n} \mid a_m \neq 0\} \geq 0$. We denote by $\mathbf{k}((T^*))$ the quotient field of $\mathbf{k}[[T^*]]$.

Convergent Puiseux series. Let $\varphi \in \mathbb{C}[[T^*]]$ and $n \in \mathbb{N}$ such that $\varphi = f(T^{\frac{1}{n}})$ holds for some $f \in \mathbb{C}[[T]]$. We say that the Puiseux series φ is *convergent* if we have $f \in \mathbb{C}\langle T \rangle$. Convergent Puiseux series form an integral domain denoted by $\mathbb{C}\langle T^* \rangle$; its quotient field is denoted by $\mathbb{C}(\langle T^* \rangle)$. For every $\varphi \in \mathbb{C}((T^*))$, there exist $n \in \mathbb{Z}$, $r \in \mathbb{N}_{>0}$ and a sequence of complex numbers $a_n, a_{n+1}, a_{n+2}, \ldots$ such that we have

$$\varphi = \sum_{m=n}^{\infty} a_m T^{\frac{m}{r}} \text{ and } a_n \neq 0.$$

Then, we define $\text{ord}(\varphi) = \frac{n}{r}$.

Puiseux Theorem. If $\mathbf{k}$ has characteristic zero, the field $\mathbf{k}((T^*))$ is the algebraic closure of the field of formal Laurent series over $\mathbf{k}$. Moreover, if $\mathbf{k} = \mathbb{C}$, the field $\mathbb{C}(\langle T^* \rangle)$ is algebraically closed as well. From now on, we assume $\mathbf{k} = \mathbb{C}$.

Puiseux expansion. Let $\mathbb{B} = \mathbb{C}((X^*))$ or $\mathbb{C}(\langle X^* \rangle)$. Let $f \in \mathbb{B}[Y]$, where $d := \deg(f, Y) > 0$. Let $h := \text{lc}(f, Y)$. According to Puiseux Theorem, there exists $\varphi_i \in \mathbb{B}$, $i = 1, \ldots, d$, such that $\frac{f}{h} = (Y - \varphi_1) \cdots (Y - \varphi_d)$. We call $\varphi_1, \ldots, \varphi_d$ the *Puiseux expansions* of f at the origin.

Puiseux parametrization. Let $f \in \mathbb{C}\langle X \rangle[Y]$. A *Puiseux parametrization* of f is a pair $(\psi(T), \varphi(T))$ of elements of $\mathbb{C}\langle T \rangle$ for some new variable T, such that (1) $\psi(T) = T^\varsigma$, for some $\varsigma \in \mathbb{N}_{>0}$; (2) $f(X = \psi(T), Y = \varphi(T)) = 0$ holds in $\mathbb{C}\langle T \rangle$, and (3) there is no integer $k > 1$ such that both $\psi(T)$ and $\varphi(T)$ are in $\mathbb{C}\langle T^k \rangle$. The index ς is called the *ramification index* of the parametrization $(T^\varsigma, \varphi(T))$. It is intrinsic to f and $\varsigma \leq \deg(f, Y)$. Let $z_1, \ldots, z_\varsigma$ denote the distinct roots of unity of order ς in $\mathbb{C}$. Then $\varphi(z_i X^{1/\varsigma})$, for $i = 1, \ldots, \varsigma$, are ς Puiseux expansions of f. For a Puiseux expansion φ of f, let c minimum such that both $\varphi = g(T^{1/c})$ and $g \in \mathbb{C}\langle T \rangle$ holds. Then $(T^c, g(T))$ is a Puiseux parametrization of f.

We conclude this section by a few lemmas which are immediate consequences of the above review.

Lemma 6. *We have:* $\lim(W(R)) = \overline{W(R)} \cap V(h_R)$. *In particular,* $\lim(W(R))$ *is either empty or an affine variety of dimension* $s - e - 1$.

Lemma 7. *If R is a primitive regular chain, that is, if R is a system of generators of its saturated ideal, then we have* $\lim(W(R)) = V(R) \cap V(h_R)$.

Lemma 8. *If N is a strongly normalized regular chain such that* $\text{sat}(R) = \text{sat}(N)$ *and* $V(h_N) = V(\widehat{h_R})$ *both hold, then we have* $\lim(W(R)) \subseteq \lim(W(N))$.

Lemma 9. *Let $x \in \mathbb{A}^s$ such that $x \notin W(R)$. Then $x \in \lim(W(R))$ holds if and only if there exists a sequence $(\alpha_n, n \in \mathbb{N})$ of points in $\mathbb{A}^s$ such that $\alpha_n \in W(R)$ for all $n \in \mathbb{N}$ and $\lim_{n \to \infty} \alpha_n = x$.*

Lemma 10. *Recall that R writes $\{r_1, \ldots, r_e\}$. If $e > 1$ holds, writing $R' = \{r_1, \ldots, r_{e-1}\}$ and $r = r_e$, we have*

$$\lim(W(R' \cup r)) \subseteq \lim(W(R')) \cap \lim(W(r)).$$

Lemma 11. *Let $\varphi \in \mathbb{C}(\langle T^* \rangle)$ and let $p/q \in \mathbb{Q}$ be the order of φ. Let $(\alpha_n, n \in \mathbb{N})$ be a sequence of complex numbers converging to zero and let N be a positive integer such that $(\varphi(\alpha_n), n \geq N)$ is well defined. Then, if $p/q < 0$ holds, the sequence $(\varphi(\alpha_n), n \geq N)$ escapes to infinity while if $p/q \geq 0$, the sequence $(\varphi(\alpha_n), n \geq N)$ converges to the complex number $\varphi(0)$.*

4.2.1 Basic techniques

This section is an overview of the basic techniques of this chapter . This presentation is meant to help the non-expert reader understand our objectives and solutions. In particular, the results of this section are stated for regular chains in three variables, while the statements of Sections 4.3 to 4.7 do not have this restriction.

Recall that $R \subseteq \mathbb{C}[X_1, \ldots, X_s]$ is a regular chain whose saturated ideal has height $1 \leq e \leq s$. As mentioned in the introduction, we mainly focus on the case $e = s - 1$, that is, $\mathrm{sat}(R)$ has dimension one.

Lemma 6 and the assumption $e = s - 1$ imply that $\lim(W(R))$ consists of finitely many points.

We further assume that R is strongly normalized, thus we have h_R lies in $\mathbb{C}[X_1]$.

Lemma 7 and the assumption $h_R \in \mathbb{C}[X_1]$ imply that computing $\lim(W(R))$ reduces to check, for each root $\alpha \in \mathbb{C}$ of h_R whether or not there is a point $x \in \lim(W(R))$ whose X_1-coordinate is α. Without loss of generality, it is enough to develop our results for the case $\alpha = 0$. Indeed, a change of coordinates can be used to reduce to this latter assumption.

We start by considering the case $n = 2$. Thus, our regular chain R consists of a single polynomial $r_1 \in \mathbb{C}[X_1, X_2]$ whose initial h_1 satisfies $h_1(0) = 0$. Lemma 12 provides a necessary and sufficient condition for a point of $(\alpha, \beta) \in \mathbb{A}^2$, with $\alpha = 0$, to satisfy $(\alpha, \beta) \in \lim(W(\{r_1\}))$.

Let d be the degree of r_1 in X_2. Applying Puiseux Theorem, we consider $\varphi_1, \ldots, \varphi_d \in \mathbb{C}(\langle X_1^* \rangle)$ such that the following holds

$$\frac{r_1}{h_1} = (X_2 - \varphi_1) \cdots (X_2 - \varphi_d) \tag{4.1}$$

in $\mathbb{C}(\langle X_1^* \rangle)[X_2]$. We assume that the series $\varphi_1, \ldots, \varphi_d$ are numbered in such a way that each of $\varphi_1, \ldots, \varphi_c$ has a non-negative order while each of $\varphi_{c+1}, \ldots, \varphi_d$ has a negative order, for some c such that $0 \leq c \leq d$.

Lemma 12. *With $h_1(0) = 0$, for all $\beta \in \mathbb{C}$, the following two conditions are equivalent*

(*i*) $(0, \beta) \in \lim(W(r_1))$ *holds,*

(*ii*) *there exists $1 \leq j \leq c$ and a sequence $(\alpha_n, n \in \mathbb{N})$ of complex numbers such that the sequence $(\varphi_j(\alpha_n), n \in \mathbb{N})$ is well defined, we have $h_1(\alpha_n) \neq 0$ for all $n \in \mathbb{N}$ and we we have*

$$\lim_{n \to \infty} \alpha_n = 0 \text{ and } \lim_{n \to \infty} \varphi_j(\alpha_n) = \beta.$$

Proof. We first prove the implication (*ii*) $\Rightarrow$ (*i*). Equation (4.1) together with (*ii*) implies $(\alpha_n, \varphi_j(\alpha_n)) \in V(r_1)$ for all $n \in \mathbb{N}$. Since we also have $(\alpha_n, \varphi_j(\alpha_n)) \notin V(h_1)$ for all $n \in \mathbb{N}$ and $\lim_{n \to \infty} (\alpha_n, \varphi_j(\alpha_n)) = (0, \beta)$, we deduce (*i*), thanks to Lemma 9.

We now prove the implication (*i*) $\Rightarrow$ (*ii*). By Lemma 9, there exists a sequence $((\alpha_n, \beta_n), n \in \mathbb{N})$ in $\mathbb{A}^2$ such that for all $n \in \mathbb{N}$ we have: (1) $h_1(\alpha_n) \neq 0$, (2) $r_1(\alpha_n, \beta_n) = 0$, and (3) $\lim_{n \to \infty} (\alpha_n, \beta_n) = (0, \beta)$. Since $\lim_{n \to \infty} \alpha_n = 0$, each series $\varphi_1(\alpha_n), \ldots, \varphi_d(\alpha_n)$ is well defined for n larger than some positive integer N. Hypotheses (1) and (2), together with Equation (4.1), imply that for all $n \geq N$ the product

$$(\beta_n - \varphi_1(\alpha_n)) \cdots (\beta_n - \varphi_c(\alpha_n))(\beta_n - \varphi_{c+1}(\alpha_n)) \cdots (\beta_n - \varphi_d(\alpha_n))$$

is 0. Since $\lim_{n \to \infty} \beta_n = \beta$, and by definition of the integer c, each of the sequences $(\beta_n - \varphi_1(\alpha_n)), \ldots, (\beta_n - \varphi_c(\alpha_n))$ converges while each of the sequences $(\beta_n - \varphi_{c+1}(\alpha_n)), \ldots, (\beta_n - \varphi_d(\alpha_n))$ escapes to infinity. Thus, for n large enough the product $(\beta_n - \varphi_1(\alpha_n)) \cdots (\beta_n - \varphi_c(\alpha_n))$ is zero. Therefore, one of sequences $(\beta_n - \varphi_1(\alpha_n)), \ldots, (\beta_n - \varphi_c(\alpha_n))$ converges to 0 and the conclusion follows. □

Lemmas 11 and 12 immediately imply the following.

Proposition 2. *With $h_1(0) = 0$, for all $\beta \in \mathbb{C}$, we have*

$$(0, \beta) \in \lim(W(r_1)) \iff \beta \in \{\varphi_1(0), \ldots, \varphi_c(0)\}.$$

Next, we consider the case $n = 3$. Hence, our regular chain R consists of two polynomials $r_1 \in \mathbb{C}[X_1, X_2]$ and $r_2 \in \mathbb{C}[X_1, X_2, X_3]$ with respective initials h_1 and h_2. We assume that 0 is a root of the product $h_1 h_2$ and we are looking for all $\beta \in \mathbb{C}$ and all $\gamma \in \mathbb{C}$ such that $(0, \beta, \gamma) \in \lim(W(r_1, r_2))$.

Lemma 10 tells us that $(0, \beta, \gamma) \in \lim(W(r_1, r_2))$ implies $(0, \beta) \in \lim(W(r_1))$. This observation together with Proposition 2 yields immediately the following.

Proposition 3. *With $h_1(0) = 0$ and $h_2(0) \neq 0$, assuming that r_1 is primitive over $\mathbb{C}[X_1]$, for all $\beta \in \mathbb{C}$ and all $\gamma \in \mathbb{C}$, we have*

$$(0, \beta, \gamma) \in \lim(W(r_1, r_2)) \iff (0, \beta, \gamma) \in V(r_1, r_2).$$

We turn now our attention to the case $h_1(0) = h_2(0) = 0$. Since $(0, \beta) \in \lim(W(r_1))$ is a necessary condition for $(0, \beta, \gamma) \in \lim(W(r_1, r_2))$ to hold we apply Proposition 2 and assume $\beta \in \{\varphi_1(0), \ldots, \varphi_c(0)\}$. Without loss of generality, we further assume $\beta = 0$. For each $1 \leq j \leq c$, such that $\varphi_j(0) = 0$ holds, we define the univariate polynomial $f_2^j \in \mathbb{C}(\langle X_1^* \rangle)[X_3]$ by

$$f_2^j(X_1, X_3) = r_2(X_1, \varphi_j(X_1), X_3). \tag{4.2}$$

Let b be the degree of f_2^j. Applying again Puiseux theorem, we consider $\psi_1, \ldots, \psi_b \in \mathbb{C}(\langle X_1^* \rangle)$ such that the following holds

$$\frac{f_2^j}{h_2} = (X_3 - \psi_1) \cdots (X_3 - \psi_b) \tag{4.3}$$

in $\mathbb{C}(\langle X_1^* \rangle)[X_3]$. We assume that the series $\psi_1, \ldots, \psi_b$ are numbered in such a way that each of $\psi_1, \ldots, \psi_a$ has a non-negative order while each of $\psi_{a+1}, \ldots, \psi_b$ has a negative order, for some a such that $0 \leq a \leq b$.

Lemma 13. *For all $\gamma \in \mathbb{C}$, the following two conditions are equivalent.*

(*i*) $(0, 0, \gamma) \in \lim(W(r_1, r_2))$ *holds,*

(*ii*) *there exist integers j, k with $1 \leq j \leq c$ and $1 \leq k \leq a$, and two sequences $(\alpha_n, n \in \mathbb{N})$, $(\beta_n, n \in \mathbb{N})$ of complex numbers such that:*

  (*a*) *the sequences $(\varphi_j(\alpha_n), n \in \mathbb{N})$ and $(\psi_k(\beta_n), n \in \mathbb{N})$ are well defined,*

  (*b*) *$h_1(\alpha_n) \neq 0$ and $h_2(\alpha_n) \neq 0$, for all $n \in \mathbb{N}$,*

  (*c*) *$\beta_n = \varphi_j(\alpha_n)$, for all $n \in \mathbb{N}$,*

  (*d*) *$\lim_{n \to \infty} (\alpha_n, \beta_n, \psi_k(\beta_n)) = (0, 0, \gamma)$.*

Proof. Proving the implication $(ii) \Rightarrow (i)$ is easy. We now prove the implication $(i) \Rightarrow (ii)$. By Lemma 9, there exists a sequence $((\alpha_n, \beta_n, \gamma_n), n \in \mathbb{N})$ in $\mathbb{A}^3$ s.t. for all $n \in \mathbb{N}$ we have: (1) $h_1(\alpha_n) \neq 0$, (2) $h_2(\alpha_n) \neq 0$, (3) $r_1(\alpha_n, \beta_n) = 0$, (4) $r_2(\alpha_n, \beta_n, \gamma_n) = 0$, (5) $\lim_{n\to\infty}(\alpha_n, \beta_n, \gamma_n) = (0, 0, \gamma)$. Following the proof of Lemma 12, we know that for n large enough the product $(\beta_n - \varphi_1(\alpha_n)) \cdots (\beta_n - \varphi_c(\alpha_n))$ is zero. Therefore, from one of the sequences $(\beta_n - \varphi_1(\alpha_n)), \ldots, (\beta_n - \varphi_c(\alpha_n))$, say the j-th, one can extract an (infinite) sub-sequence whose terms are all zero. Thus, without loss of generality, we assume that $\beta_n = \varphi_j(\alpha_n)$ holds, for all $n \in \mathbb{N}$. Hence, for all $n \in \mathbb{N}$, we have $f_2^j(\alpha_n, \gamma_n) = r_2(\alpha_n, \beta_n, \gamma_n) = 0$. Together with Equation (4.3) and following the proof of Lemma 12, we deduce the desired result. □

Lemmas 11 and 13 immediately imply the following.

Proposition 4. *For all $\gamma \in \mathbb{C}$, the following two conditions are equivalent.*

(*i*) $(0, 0, \gamma) \in \lim(W(r_1, r_2))$ *holds,*

(*ii*) *there exist integers j, k with $1 \leq j \leq c$ and $1 \leq k \leq a$, such that $\varphi_j(0) = 0$ and $\psi_k(0) = \gamma$.*

Therefore, applying Puiseux theorem to r_1 and f_2^j, then checking the constant terms of the series $\psi_1, \ldots, \psi_b$ provides a way to compute all $\gamma \in \mathbb{C}$ such that $(0, 0, \gamma)$ is a limit point of $W(r_1, r_2)$. Theorem 7 in Sections 4.3 states this principle formally for an arbitrary regular chain R in dimension one.

Finally, one should also consider the case $h_1(0) \neq 0, h_2(0) = 0$. In fact, it is easy to see that this latter case can be handled in a similar manner as the case $h_1(0) = 0, h_2(0) = 0$.

4.3 Puiseux expansions of a regular chain

In this section, we introduce the notion of Puiseux expansions of a regular chain, motivated by the work of [63, 2] on Puiseux expansions of space curves. Throughout this section, let $R = \{r_1, \ldots, r_{s-1}\} \subset \mathbb{C}[X_1 < \cdots < X_s]$ be a strongly normalized regular chain whose saturated ideal has dimension one and assume that X_1 is free w.r.t. R.

Lemma 14. *Let $h_R(X_1)$ be the product of the initials of the polynomials in R. Let $\rho > 0$ be small enough such that the set $U_\rho := \{x = (x_1, \ldots, x_s) \in \mathbb{C}^s \mid 0 < |x_1| < \rho\}$ does not contain any zeros of h_R. Define $V_\rho(R) := V(R) \cap U_\rho$. Then, we have $W(R) \cap U_\rho = V_\rho(R)$. Let $R' := \{\text{primpart}(r_1), \ldots, \text{primpart}(r_{s-1})\}$. Then $V_\rho(R) = V_\rho(R')$.*

Proof. Let $x \in W(R) \cap U_\rho$, then $x \in V(R)$ and $x \in U_\rho$ hold, which implies that $W(R) \cap U_\rho \subseteq V(R) \cap U_\rho$. Let $x \in V(R) \cap U_\rho$. Since $U_\rho \cap V(h_R) = \emptyset$, we have $x \in W(R)$. Thus $V(R) \cap U_\rho \subseteq W(R) \cap U_\rho$. So $W(R) \cap U_\rho = V_\rho(R)$. Similarly we have $V_\rho(R) = V_\rho(R')$. □

Notation 2. *Let $W \subseteq \mathbb{C}^s$. Denote* $\lim_0(W) := \{x = (x_1, \ldots, x_s) \in \mathbb{C}^s \mid x \in \lim(W) \text{ and } x_1 = 0\}$.

Lemma 15. *We have* $\lim_0(W(R)) = \lim_0(V_\rho(R))$.

Proof. By Lemma 14, we have $W(R) \cap U_\rho = V_\rho(R)$. Meanwhile, $\lim_0(W(R)) = \lim_0(W(R) \cap U_\rho)$ holds. Thus $\lim_0(W(R)) = \lim_0(V_\rho(R))$ holds. □

Lemma 16. *For $1 \leq i \leq s-1$, let $d_i := \deg(r_i, X_{i+1})$. Then R generates a zero-dimensional ideal in $\mathbb{C}(\langle X_1^* \rangle)[X_2, \ldots, X_s]$. Let $V^*(R)$ be the zero set of R in $\mathbb{C}(\langle X_1^* \rangle)^{s-1}$. Then $V^*(R)$ has exactly $\prod_{i=1}^{s-1} d_i$ points, counting multiplicities.*

Proof. It follows directly from the definition of regular chain, and the fact that $\mathbb{C}(\langle X_1^* \rangle)$ is an algebraically closed field. □

Definition 10. *We use the notations of Lemma 16. Each point in $V^*(R)$ is called a* Puiseux expansion *of R.*

Notation 3. *Let $m = |V^*(R)|$. Write $V^*(R) = \{\Phi_1, \ldots, \Phi_m\}$. For $i = 1, \ldots, m$, write $\Phi_i = (\Phi_i^1(X_1), \ldots, \Phi_i^{s-1}(X_1))$. Let $\rho > 0$ be small enough such that for $1 \leq i \leq m$, $1 \leq j \leq s-1$, each $\Phi_i^j(X_1)$ converges in $0 < |X_1| < \rho$. We define $V_\rho^*(R) := \cup_{i=1}^m \{x \in \mathbb{C}^s \mid 0 < |x_1| < \rho, x_{j+1} = \Phi_i^j(x_1), j = 1, \ldots, s-1\}$.*

Theorem 5. *We have $V_\rho^*(R) = V_\rho(R)$.*

Proof. We prove this by induction on s. For $i = 1, \ldots, s-1$, recall that h_i is the initial of r_i. If $s = 2$, we have

$$r_1(X_1, X_2) = h_1(X_1) \prod_{i=1}^{d_1} (X_2 - \Phi_i^1(X_1)).$$

So $V_\rho^*(R) = V_\rho(R)$ clearly holds.

Now we consider $s > 2$. Write $R' = \{r_1, \ldots, r_{s-2}\}$, $R = R' \cup \{r_{s-1}\}$, $X' = X_2, \ldots, X_{s-1}$, $X = (X_1, X', X_s)$, $x' = x_2, \ldots, x_{s-1}$, $x = (x_1, x', x_s)$, and $m' = |V^*(R')|$. For $i = 1, \ldots, m$, let $\Phi_i = (\Phi_i', \Phi_i^{s-1})$, where Φ_i' stands for $\Phi_i^1, \ldots, \Phi_i^{s-2}$. Assume the theorem holds for R', that is $V_\rho^*(R') = V_\rho(R')$. For any $i = 1, \ldots, m'$, there exist $i_1, \ldots, i_{d_{s-1}} \in \{1, \ldots, m\}$ such that we have

$$r_{s-1}(X_1, X' = \Phi_i', X_s) = h_{s-1}(X_1) \prod_{k=1}^{d_{s-1}} (X_s - \Phi_{i_k}^{s-1}(X_1)). \tag{4.4}$$

Note that $V^*(R) = \cup_{i=1}^{m'} \cup_{k=1}^{d_{s-1}} \{(X' = \Phi_i', X_s = \Phi_{i_k}^{s-1})\}$. Therefore, by induction hypothesis and Equation (4.4), we have

$$\begin{aligned}
V_\rho^*(R) &= \cup_{i=1}^{m'} \cup_{k=1}^{d_{s-1}} \{x \mid x \in U_\rho, x' = \Phi_i'(x_1), x_s = \Phi_{i_k}^{s-1}(x_1)\} \\
&= \cup_{k=1}^{d_{s-1}} \{x \mid (x_1, x') \in V_\rho^*(R'), x_s = \Phi_{i_k}^{s-1}(x_1)\} \\
&= \{x \mid (x_1, x') \in V_\rho^*(R'), r_{s-1}(x_1, x', x_s) = 0\} \\
&= \{x \mid (x_1, x') \in V_\rho(R'), r_{s-1}(x_1, x', x_s) = 0\} \\
&= V_\rho(R).
\end{aligned}$$

□

Theorem 6. *Let $V_{\geq 0}^*(R) := \{\Phi = (\Phi^1, \ldots, \Phi^{s-1}) \in V^*(R) \mid \mathrm{ord}(\Phi^j) \geq 0, j = 1, \ldots, s-1\}$. Then we have*

$$\lim_0(W(R)) = \cup_{\Phi \in V_{\geq 0}^*(R)} \{(X_1 = 0, \Phi(X_1 = 0))\}.$$

Proof. By definition of $V^*_{\geq 0}(R)$, we immediately have

$$\lim_0(V^*_\rho(R)) = \cup_{\Phi \in V^*_{\geq 0}(R)}\{(X_1 = 0, \Phi(X_1 = 0))\}.$$

Next, by Theorem 5, we have $V^*_\rho(R) = V_\rho(R)$. Thus, we have $\lim_0(V^*_\rho(R)) = \lim_0(V_\rho(R))$. Besides, with Lemma 15, we have $\lim_0(W(R)) = \lim_0(V_\rho(R))$. Thus the theorem holds. □

Definition 11. *Let $V^*_{\geq 0}(R)$ be as defined in Theorem 6. Let $M = |V^*_{\geq 0}(R)|$. For each $\Phi_i = (\Phi_i^1, \ldots, \Phi_i^{s-1}) \in V^*_{\geq 0}(R)$, $1 \leq i \leq M$, we know that $\Phi_i^j \in \mathbb{C}(\langle X_1^* \rangle)$. Moreover, by Equation (4.4), we know that for $j = 1, \ldots, s-1$, Φ_i^j is a Puiseux expansion of $r_j(X_1, X_2 = \Phi_i^1, \ldots, X_j = \Phi_i^{j-1}, X_{j+1})$. Let $\varsigma_{i,j}$ be the ramification index of Φ_i^j and $(T^{\varsigma_{i,j}}, X_{j+1} = \varphi_i^j(T))$, where $\varphi_i^j \in \mathbb{C}\langle T \rangle$, be the corresponding Puiseux parametrization of Φ_i^j. Let ς_i be the least common multiple of $\{\varsigma_{i,1}, \ldots, \varsigma_{i,s-1}\}$. Let $g_i^j = \varphi_i^j(T = T^{\varsigma_i/\varsigma_{i,j}})$. We call the set $\mathfrak{G}_R := \{(X_1 = T^{\varsigma_i}, X_2 = g_i^1(T), \ldots, X_s = g_i^{s-1}(T)), i = 1, \ldots, M\}$ a* system of Puiseux parametrizations *of R.*

Theorem 7. *We have*

$$\lim_0(W(R)) = \mathfrak{G}_R(T = 0).$$

Proof. It follows directly from Theorem 6 and Definition 11. □

Remark 5. *The limit points of $W(R)$ at $X_1 = \alpha \neq 0$ can be reduced to the computation of $\lim_0(W(R))$ by a coordinate transformation $X_1 = X_1 + \alpha$. Given an arbitrary one-dimensional regular chain R, the set $\lim(W(R))$ can be computed in the following manner. Compute a regular chain N which is strongly normalized and such that $\mathrm{sat}(R) = \mathrm{sat}(N)$ and $V(h_N) = V(\widehat{h_R})$ both hold, where $\widehat{h_R}$ is the iterated resultant of h_R w.r.t R. See [25]. Let $X_i := \mathrm{mvar}(h_R)$. Note that N is still a regular chain w.r.t. the new order $X_i < \{X_1, \ldots, X_n\} \setminus \{X_i\}$. By Lemma 8, we have $\lim(W(R)) = \lim(W(N)) \setminus W(R)$.*

4.4 Puiseux parametrization in finite accuracy

In this section, we define the Puiseux parametrizations of a polynomial $f \in \mathbb{C}\langle X \rangle[Y]$ in finite accuracy, see Definition 13. For $f \in \mathbb{C}\langle X \rangle[Y]$, we also define the approximation of f for a given accuracy, see Definition 12. This approximation of f is a polynomial in $\mathbb{C}[X, Y]$. In Sections 4.5 and 4.6, we prove that to compute a Puiseux parametrization of f of a given accuracy, it suffices to compute a Puiseux parametrization of an approximation of f of some finite accuracy.

In this section, we review and adapt the classical Newton-Puiseux algorithm to compute Puiseux parametrizations of a polynomial $f \in \mathbb{C}[X, Y]$ of a given accuracy. Since we do not need to compute the singular part of Puiseux parametrizations, the usual requirement $\mathrm{discrim}(f, Y) \neq 0$ is dropped.

Definition 12. *Let $f = \sum_{i=0}^{\infty} a_i X^i \in \mathbb{C}[[X]]$. For any $\tau \in \mathbb{N}$, we call $f^{(\tau)} := \sum_{i=0}^{\tau} a_i X^i$ the* polynomial part *of f* of accuracy $\tau + 1$. *Let $f = \sum_{i=0}^{d} a_i(X) Y^i \in \mathbb{C}\langle X \rangle[Y]$. For any $\tau \in \mathbb{N}$, we call $\widetilde{f^{(\tau)}} := \sum_{i=0}^{d} a_i^{(\tau)} Y^i$ the* approximation of f of accuracy $\tau + 1$.

Definition 13. *Let $f \in \mathbb{C}\langle X\rangle[Y]$, with $\deg(f, Y) > 0$. Let $\sigma, \tau \in \mathbb{N}_{>0}$ and $g(T) = \sum_{k=0}^{\tau-1} b_k T^k$. Let $\{T^{k_1}, \ldots, T^{k_m}\}$ be the support of $g(T)$. The pair $(T^\sigma, g(T))$ is called* a Puiseux parametrization of f of accuracy τ *if there exists a Puiseux parametrization $(T^\varsigma, \varphi(T))$ of f such that:*

(*i*) *σ divides ς.*
(*ii*) $\gcd(\sigma, k_1, \ldots, k_m) = 1$.
(*iii*) *$g(T^{\varsigma/\sigma})$ is the polynomial part of $\varphi(T)$ of accuracy $(\varsigma/\sigma)(\tau - 1) + 1$.*

Note that if $\sigma = \varsigma$, then $g(T)$ is the polynomial part of $\varphi(T)$ of accuracy τ.

We borrow the following notion from [33] in order to state an algorithm for computing Puiseux parametrizations.

Definition 14 ([33])**.** *A $\mathbb{C}$-term[3] is defined as a triple $t = (q, p, \beta)$, where q and p are coprime integers, $q > 0$ and $\beta \in \mathbb{C}$ is non-zero. A $\mathbb{C}$-expansion is a sequence $\pi = (t_1, t_2, \ldots)$ of $\mathbb{C}$-terms, where $t_i = (q_i, p_i, \beta_i)$. We say that π is* finite *if there are only finitely many elements in π.*

Definition 15. *Let $\pi = (t_1, \ldots, t_N)$ be a finite $\mathbb{C}$-expansion. We define a pair $(T^\sigma, g(T))$ of polynomials in $\mathbb{C}[T]$ in the following manner:*

(*i*) *if $N = 1$, set $\sigma = 1$, $g(T) = 0$ and $\delta_N = 0$,*
(*ii*) *otherwise, let $a := \prod_{i=1}^{N} q_i$, $c_i := \sum_{j=1}^{i} \left(p_j \prod_{k=j+1}^{N} q_k\right)$ $(1 \le i \le N)$, and $\delta_i := c_i/\gcd(a, c_1, \ldots, c_N)$ $(1 \le i \le N)$. Set $\sigma := a/\gcd(a, c_1, \ldots, c_N)$ and $g(T) := \sum_{i=1}^{N} \beta_i T^{\delta_i}$.*

We call the pair $(T^\sigma, g(T))$ the Puiseux parametrization of π of accuracy $\delta_N + 1$. *Denote by* ConstructParametrization *an algorithm to compute $(T^\sigma, g(T))$ from π.*

Definition 16. *Let $f \in \mathbb{C}\langle X\rangle[Y]$ and write f as $f(X, Y) := \sum_{i=0}^{d} \left(\sum_{j=0}^{\infty} a_{i,j} X^j\right) Y^i$. The* Newton Polygon *of f is defined as the lower part of the convex hull of the set of points (i, j) in the plane such that $a_{i,j} \neq 0$.*

Let $f \in \mathbb{C}\langle X\rangle[Y]$. Next we present an algorithm, called NewtonPolygon to compute the segments in the Newton Polygon of f. This algorithm is from R.J. Walker's book [101].

NewtonPolygon(f, I)
Input: A polynomial $f \in \mathbb{C}\langle X\rangle[Y]$; a controlling flag I, whose value is 1 or 2.
Output: The Newton Polygon of f. If $I = 1$, only segments with non-positive slopes are computed. If $I = 2$, only segments with negative slopes are computed.
Description:
- Write f as $f = \sum_{i=0}^{d} b_i(X) Y^i$, where $b_i(X) = \sum_{j=0}^{\infty} a_{i,j} X^j$.
- For $0 \le i \le d$, define $\delta_i := \mathrm{ord}(b_i)$.
- For $0 \le i \le d$, we plot the points P_i with coordinates (i, δ_i); we omit P_i if $\delta_i = \infty$.
- We join P_0 to P_d with a convex polygonal arc each of whose vertices is a P_i and such that no P_i lies below the arc.
- If $I = 1$, output all segments with non-positive slopes in the polygon; if $I = 2$, output all segments with negative slopes in the polygon.

Next we introduce some notations which are necessary to present Algorithm 3 for computing Puiseux parametrization of some finite accuracy as defined in Definition 13.

Let $f \in \mathbb{C}[X, Y]$, $t = (q, p, \beta)$ be a $\mathbb{C}$-term and $\ell \in \mathbb{N}$ s.t. $\mathsf{NewPoly}(f, t, \ell) := X^{-\ell} f(X^q, X^p(\beta + Y)) \in \mathbb{C}[X, Y]$. Let $f = \sum_{i=0}^{d} \sum_{j=0}^{m} a_{i,j} X^j Y^i \in \mathbb{C}[X, Y]$ and let Δ be a segment of the Newton

[3] It is a simplified version of Duval's definition.

Polygon of f. Denote $\mathsf{SegmentPoly}(f, \Delta) := (q, p, \ell, \phi)$ such that the following holds: (1) $q, p, \ell \in \mathbb{N}$; $\phi \in \mathbb{C}[Z]$; q and p are coprime, $q > 0$; (2) for any $(i, j) \in \Delta$, we have $qj + pi = \ell$; and (3) letting $i_0 := \min(\{i \mid (i, j) \in \Delta\})$, we have $\phi = \sum_{(i,j)\in\Delta} a_{i,j} Z^{(i-i_0)/q}$.

Theorem 8. *Algorithm 3 terminates and is correct.*

Proof. It directly follows from the proof of the Newton-Puiseux algorithm in Walker's book [101], the relation between $\mathbb{C}$-expansion and Puiseux parametrization discussed in Duval's paper [33], and Definitions 15 and 13. □

Algorithm 2: NonzeroTerm

Input: $f \in \mathbb{C}[X, Y]$; $I = 1$ or 2.
Output: A finite set of pairs (t, ℓ), where t is a $\mathbb{C}$-term, and $\ell \in \mathbb{N}$.

```
1  begin
2      S := ∅;
3      for each Δ ∈ NewtonPolygon(f, I) do
4          (q, p, ℓ, φ) := SegmentPoly(f, Δ);
5          for each root ξ of φ in ℂ do
6              for each root β of U^q − ξ in ℂ do
7                  t := (q, p, β);
8                  S := S ∪ {(t, ℓ)}
9      return S
10 end
```

4.5 Computing in finite accuracy

Let $f \in \mathbb{C}\langle X\rangle[Y]$. In this section, we consider the following problems:

(a) Is it possible to use an approximation of f of some finite accuracy m in order to compute a Puiseux parametrization of f of some finite accuracy τ?
(b) If yes, how to deduce m from f and τ?
(c) Provide a bound on m.

Theorem 9 provides the answers to (a) and (b) while Lemma 20 answers (c).

Lemma 17 ([35])**.** *Let $\underline{X} = X_1, \ldots, X_s$ and $\underline{Y} = Y_1, \ldots, Y_m$. For $g_1, \ldots, g_s \in \mathbb{C}[[\underline{Y}]]$, with* $\mathrm{ord}(g_i) \geq 1$*, there is a $\mathbb{C}$-algebra homomorphism (called the* substitution homomorphism*)*

$$\Phi_g : \begin{array}{ccc} \mathbb{C}[[\underline{X}]] & \longrightarrow & \mathbb{C}[[\underline{Y}]] \\ f & \longmapsto & f(g_1(\underline{Y}), \ldots, g_s(\underline{Y})). \end{array}$$

Moreover, if $g_1, \ldots, g_s$ are convergent power series, then we have $\Phi_g(\mathbb{C}\langle\underline{X}\rangle) \subseteq \mathbb{C}\langle\underline{Y}\rangle$ holds.

Definition 17 ([35])**.** *Let $f = \sum a_{\mu\nu} X^\mu Y^\nu \in \mathbb{C}[[X, Y]]$. The* carrier *of f is defined as*

$$\mathrm{carr}(f) = \{(\mu, \nu) \in \mathbb{N}^2 \mid a_{\mu\nu} \neq 0\}.$$

Algorithm 3: NewtonPuiseux

Input: $f \in \mathbb{C}[X, Y]$; a given accuracy $\tau > 0 \in \mathbb{N}$.
Output: All the Puiseux parametrizations of f of accuracy τ.

```
1  begin
2      π := ( ); S := {(π, f)}; P := ∅;
3      while S ≠ ∅ do
4          choose (π*, f*) ∈ S; S := S \ {(π*, f*)};
5          if π* = ( ) then I := 1 else I := 2;
6          (T^σ, g(T)) := ConstructParametrization(π*);
7          if deg(g(T), T) + 1 < τ then
8              C := NonzeroTerm(f*, I);
9              if C = ∅ then
10                 P := P ∪ {(T^σ, g(T))}// a finite Puiseux parametrization is
                       found
11             else
12                 for each (t = (p, q, β), ℓ) ∈ C do
13                     π** := π* ∪ (t);
14                     f** := NewPoly(f*, t, ℓ);
15                     S := S ∪ {(π**, f**)}
16         else
17             P := P ∪ {(T^σ, g(T))}
18     return P
19 end
```

Lemma 18. *Let $f \in \mathbb{C}\langle X\rangle[Y]$. Let $d := \deg(f, Y) > 0$. Let $q \in \mathbb{N}_{>0}$, $p, \ell \in \mathbb{N}$ and assume that q and p are coprime. Let $\beta \neq 0 \in \mathbb{C}$. Assume that q, p, ℓ define a line $L : qj + pi = \ell$ in the (i, j)-plane such that:*

(a) There are at least two points $(j_1, i_1) \in \mathrm{carr}(f)$ and $(j_2, i_2) \in \mathrm{carr}(f)$ on L with $i_1 \neq i_2$.

(b) For any $(j, i) \in \mathrm{carr}(f)$, we have $qj + pi \geq \ell$.

Let $f_1 := X_1^{-\ell} f(X_1^q, X_1^p(\beta + Y_1))$.

Then, we have the following results:

(i) We have $f_1 \in \mathbb{C}\langle X_1\rangle[Y_1]$.

(ii) For any given $m_1 \in \mathbb{N}$, there exists a number $m \in \mathbb{N}$ such that the approximation of f_1 of accuracy m_1 can be computed from the approximation of f of accuracy m.

(iii) Moreover, it suffices to take $m = \lfloor \frac{m_1 + \ell}{q} \rfloor$.

Proof. Since $q > 0$ holds, we know that $\mathrm{ord}(X_1^q) = q > 0$ holds. We also have

$$f(X_1^q, X_1^p(\beta + Y_1)) \in \mathbb{C}\langle X_1\rangle[Y_1].$$

Let

$$f(X, Y) := \sum_{i=0}^{d} \left(\sum_{j=0}^{\infty} a_{i,j} X^j \right) Y^i.$$

Then we have $f_1(X_1, Y_1) = \sum_{i=0}^{d} \left(\sum_{j=0}^{\infty} a_{i,j} X_1^{(qj+pi-\ell)} \right) (\beta + Y_1)^i$. Since for any $(j, i) \in \mathrm{carr}(f)$, we have $qj + pi \geq \ell$, the power of X_1 cannot be negative. By Lemma 17, we have $f_1 \in \mathbb{C}\langle X_1 \rangle [Y_1]$. That is (i) holds.

We prove (ii). We have

$$\begin{aligned} & f_1(X_1, Y_1) \bmod \langle X_1^{m_1} \rangle \\ = \; & \sum_{i=0}^{d} \left(\sum_{qj+pi-\ell<m_1} a_{i,j} X_1^{(qj+pi-\ell)} \right) (\beta + Y_1)^i. \end{aligned}$$

Since $q \in \mathbb{N}_{>0}$ and m_1, ℓ and i are all finite, we know that j has to be finite. In other words, there exists a finite m such that the approximation of f_1 of accuracy m_1 can be computed from the approximation of f of accuracy m. That is, (ii) holds.

Since the first m_1 terms of f_1 depends on the j-th terms of f, which satisfies the constraint $qj + pi - \ell < m_1$, we have $j < \frac{(m_1+\ell)-pi}{q} \leq \frac{(m_1+\ell)}{q}$. Let m' be the the maximum of these j's . Now we have $m' - 1 < \frac{(m_1+\ell)}{q}$. Since m' is an integer, we have $m' \leq \lfloor \frac{(m_1+\ell)}{q} \rfloor$ holds. Let $m = \lfloor \frac{(m_1+\ell)}{q} \rfloor$. Next we show shat $m_1 \geq 1$ implies that $m \geq 1$ holds. If there is at least one point $(i, j) \in L$ such that $j \geq 1$, then we have $\ell \geq q$, which implies $m \geq 1$. If the j-coordinates of all points on L is 0, then $q = 1$ and $\ell = 0$, which implies also $m \geq 1$. Thus (iii) is proved. □

Remark 6. *We use the same notations as in the previous lemma. In particular, let* $f(X, Y) := \sum_{i=0}^{d} \left(\sum_{j=0}^{\infty} a_{i,j} X^j \right) Y^i$ *and* $f_1 := X_1^{-\ell} f(X_1^q, X_1^p(\beta + Y_1))$. *For a fixed term* $a_{i,j} X^j Y^i$ *of* f, *it appears in* f_1 *as*

$$a_{i,j} X_1^{qj+pi-\ell} (\beta + Y_1)^i = \sum_{k=0}^{i} \left(\binom{i}{k} \beta^{i-k} a_{i,j} X_1^{qj+pi-\ell} \right) Y_1^k.$$

For two fixed terms $a_{i,j_1} X^{j_1} Y^i$ *and* $a_{i,j_2} X^{j_2} Y^i$ *of* f *with* $j_1 < j_2$, *since* $qj_1 + pi - \ell < qj_2 + pi - \ell$, *we know that for any fixed* k, $a_{i,j_2} X^{j_2} Y^i$ *always contributes strictly higher order of powers of* X_1 *than* $a_{i,j_1} X^{j_1} Y^i$ *in* f_1.

Remark 7. *Let* $f(X, Y) := \sum_{i=0}^{d} \left(\sum_{j=0}^{\infty} a_{i,j} X^j \right) Y^i$. *For* $0 \leq i \leq d$, *let* a_{i,j^*} *be the first nonzero coefficient among* $\{a_{i,j} | 0 \leq j < \infty\}$. *We observe that the Newton polygon of* f *is completely determined by* a_{i,j^*}, $0 \leq i \leq d$.

Theorem 9. *Let* $f \in \mathbb{C}\langle X \rangle [Y]$. *Let* $\tau \in \mathbb{N}_{>0}$. *Let* $\sigma \in \mathbb{N}_{>0}$ *and* $g(T) = \sum_{k=0}^{\tau-1} b_k T^k$. *Assume that* $(T^\sigma, g(T))$ *is a Puiseux parametrization of* f *of accuracy* τ. *Then one can compute a number* $m \in \mathbb{N}$ *such that* $(T^\sigma, g(T))$ *is a Puiseux parametrization of accuracy* τ *of* $\widetilde{f^{m-1}}$, *where* $\widetilde{f^{m-1}}$ *is the approximation of* f *of accuracy* m. *We denote by* AccuracyEstimate *an algorithm to compute* m *from* f *and* τ.

Proof. Let $f_0 := f$, $X_0 := X$ and $Y_0 := Y$. For $i = 1, 2, \ldots$, the Newton-Puiseux algorithm computes numbers $q_i, p_i, \ell_i, \beta_i$ and the transformation

$$f_i := X_i^{-\ell_i} f_{i-1}(X_i^{q_i}, X_i^{p_i}(\beta_i + Y_i))$$

such that the assumption of Lemma 18 is satisfied.

By Lemma 18, we know that for any i, a given number of terms of the coefficients of f_i in Y_i can be computed from a finite number of terms of the coefficients of f_{i-1} in Y_{i-1}. Thus

for any i, a given number of terms of the coefficients of f_i in Y_i can be computed from a finite number of terms of the coefficients of f in Y.

On the other hand, the construction of the Newton-Puiseux algorithm and Remark 7 tell us that there exists a finite M, such that σ and all the terms of $g(T)$ can be computed from a finite number of terms of the coefficients of f_i in Y_i, $i = 1, \ldots, M$.

Thus we conclude that there exists a number $m \in \mathbb{N}$ such that $(T^\sigma, g(T))$ is a Puiseux parametrization of accuracy τ of the approximation of f of accuracy m.

Next we show that there is an algorithm to compute m. We initially set $m' := \tau$. Let $f_0 := \sum_{i=0}^{d} \left(\sum_{j=0}^{m'} a_{i,j} X^j \right) Y^i$. That is, f_0 is the approximation of f of accuracy $m' + 1$. We run the Newton-Puiseux algorithm to check whether the terms $a_{k,m'} X^{m'} Y^k$, $0 \le k \le d$, make any contributions in constructing the Newton Polygons of all f_i. If at least one of them make contributions, we increase the value of m' and restart the Newton-Puiseux algorithm until none of the terms $a_{k,m'} X^{m'} Y^k$, $0 \le k \le d$, makes any contributions in constructing the Newton Polygons of all f_i. By Remark 6, we can set $m := m'$. □

Lemma 19. *Let $d, \tau \in \mathbb{N}_{>0}$. Let $a_{i,j}$, $0 \le i \le d$, $0 \le j < \tau$, and b_k, $0 \le k < \tau$ be symbols. Write $\mathbf{a} = (a_{0,0}, \ldots, a_{0,\tau-1}, \ldots, a_{d,0}, \ldots, a_{d,\tau-1})$ and $\mathbf{b} = (b_0, \ldots, b_{\tau-1})$. Let $f(\mathbf{a}, X, Y) = \sum_{i=0}^{d} \left(\sum_{j=0}^{\tau-1} a_{i,j} X^j \right) Y^i \in \mathbb{C}[\mathbf{a}][X, Y]$ and let $g(\mathbf{b}, X) = \sum_{k=0}^{\tau-1} b_k X^k \in \mathbb{C}[\mathbf{b}][X]$. Let $p := f(\mathbf{a}, X, Y = g(\mathbf{b}, X))$. Let $F_k := \mathrm{coeff}(p, X^k)$, $0 \le k < \tau - 1$, and $F := \{F_0, \ldots, F_{\tau-1}\}$. Then under the order $\mathbf{a} < \mathbf{b}$ and $b_0 < b_1 < \cdots < b_{\tau-1}$, F forms a zero-dimensional regular chain in $\mathbb{C}(\mathbf{a})[\mathbf{b}]$ with main variables $(b_0, b_1, \ldots, b_{\tau-1})$ and main degrees $(d, 1, \ldots, 1)$. In addition, we have*

(i) $F_0 = \sum_{i=0}^{d} a_{i,0} b_0^i$ *and*

(ii) $\mathrm{init}(F_1) = \cdots = \mathrm{init}(F_{\tau-1}) = \mathrm{der}(F_0, b_0) = \sum_{i=1}^{d} i \cdot a_{i,0} b_0^{i-1}$.

Proof. Write $p = \sum_{i=0}^{d} \left(\sum_{j=0}^{\tau-1} a_{i,j} X^j \right) \left(\sum_{k=0}^{\tau-1} b_k X^k \right)^i$ as a univariate polynomial in X. Observe that $F_0 = \sum_{i=0}^{d} a_{i,0} b_0^i$. Therefore F_0 is irreducible in $\mathbb{C}(\mathbf{a})[\mathbf{b}]$. Moreover, we have $\mathrm{mvar}(F_0) = b_0$ and $\mathrm{mdeg}(F_0) = d$.

Since $d > 0$, we know that $a_{1,0} \left(\sum_{k=0}^{\tau-1} b_k X^k \right)$ appears in p. Thus, for $0 \le k < \tau$, b_k appears in F_k. Moreover, for any $k \ge 1$ and $i < k$, b_k can not appear in F_i since b_k and X^k are always raised to the same power. For the same reason, for any $i > 1$, b_k^i cannot appear in F_k, for $1 \le k < \tau$. Thus $\{F_0, \ldots, F_{\tau-1}\}$ is a triangular set with main variables $(b_0, b_1, \ldots, b_{\tau-1})$ and main degrees $(d, 1, \ldots, 1)$.

Moreover, we have $\mathrm{init}(F_1) = \cdots = \mathrm{init}(F_{\tau-1}) = \sum_{i=1}^{d} i \cdot a_{i,0} b_0^{i-1}$, which is coprime with F_0. Thus $F = \{F_0, \ldots, F_{\tau-1}\}$ is a regular chain. □

Lemma 20. *Let $f = \sum_{i=0}^{d} \left(\sum_{j=0}^{\infty} a_{i,j} X^j \right) Y^i \in \mathbb{C}[[X]][Y]$. Assume that $d = \deg(f, Y) > 0$ and f is general in Y. Let $\varphi(X) = \sum_{k=0}^{\infty} b_k X^k \in \mathbb{C}[[X]]$ such that $f(X, \varphi(X)) = 0$ holds. Let $\tau > 0 \in \mathbb{N}$. Then all coefficients b_i, for $0 \le i < \tau$, can be completely determined by $\{a_{i,j} \mid 0 \le i \le d, 0 \le j < \tau\}$ if and only if b_0 is a simple zero of $f(0, Y)$. Therefore, "generically", all coefficients b_i, for $0 \le i < \tau$, can be completely determined by the approximation of f of accuracy τ.*

Proof. By $f(X, Y) = 0$, we know that $f(X, Y) = 0 \bmod \langle X^\tau \rangle$. Therefore, we have

$$\sum_{i=0}^{d} \left(\sum_{j<\tau} a_{i,j} X^j \right) \left(\sum_{k<\tau} b_k X^k \right)^i = 0 \bmod \langle X^\tau \rangle.$$

Let $p = \sum_{i=0}^{d} \left(\sum_{j<\tau} a_{i,j} X^j\right)\left(\sum_{k<\tau} b_k X^k\right)^i$. Let $F_i := \{\text{coeff}(p, X^i),\ 0 \le i < \tau\}$, and $F := \{F_0, \ldots, F_{\tau-1}\}$. Since f is general in Y and $f(X, \varphi(X)) = 0$, there exists $i^* > 0$ such that $a_{i^*,0} \neq 0$. By Lemma 19, we have $F_0 = \sum_{i=0}^{d} a_{i,0} b_0^i$. Thus b_0 can be completely determined by $a_{i,0}$, $0 \le i \le d$. In order to completely determine $b_1, \ldots, b_{\tau-1}$, it is enough to guarantee $\text{res}(F_0, F_i, b_0) \neq 0$ holds. Therefore the values of b_k, $0 \le k < \tau$ can be completely determined from almost all the values of $a_{i,j}$, $0 \le i \le d$, $0 \le j < \tau$. □

4.6 Accuracy estimates

Let $R := \{r_1(X_1, X_2), \ldots, r_{s-1}(X_1, \ldots, X_s)\} \subset \mathbb{C}[X_1 < \cdots < X_s]$ be a strongly normalized regular chain. In this section, we show that to compute the limit points of $W(R)$, it suffices to compute the Puiseux parametrizations of R of some accuracy. Moreover, we provide accuracy estimates in Theorem 10.

Lemma 21. *Let $f = a_d(X)Y^d + \cdots + a_0(X) \in \mathbb{C}\langle X\rangle[Y]$, where $d = \deg(f, Y) > 0$. For $0 \le i \le d$, let $\delta_i := \text{ord}(a_i)$. Let $k := \min(\delta_0, \ldots, \delta_d)$. Let $\widetilde{f} := X^{-k} f$. Then we have $\widetilde{f} \in \mathbb{C}\langle X\rangle[Y]$ and $\widetilde{f}$ is general in Y. This operation of producing $\widetilde{f}$ from f is called* "making f general" *and we denote it by* MakeGeneral.

Proof. Since $k = \min(\delta_0, \ldots, \delta_d)$, there exists i, $1 \le i \le d$, such that $k = \delta_i$. Moreover, for all $1 \le j \le d$, we have $\delta_j \ge k$. Thus for every such i, we have $\text{ord}(a_i(X)/X^k) = 0$ and $a_j(X)/X^k \in \mathbb{C}\langle X\rangle$, $0 \le j \le d$. This shows that $\widetilde{f} \in \mathbb{C}\langle X\rangle[Y]$ and $\widetilde{f}$ is general in Y.

□

The following lemma shows that computing limit points reduces to making a polynomial f general.

Lemma 22. *Let $f \in \mathbb{C}\langle X\rangle[Y]$, where $\deg(f, Y) > 0$, be general in Y. Let $\rho > 0$ be small enough such that f converges in $|X| < \rho$. Let $V_\rho(f) := \{(x, y) \in \mathbb{C}^2 \mid 0 < |x| < \rho, f(x, y) = 0\}$. Then $\lim_0(V_\rho(f)) = \{(0, y) \in \mathbb{C}^2 \mid f(0, y) = 0\}$ holds.*

Proof. With $1 \le i \le c$, for some c such that $1 \le c \le \deg(f, Y)$, let $(X = T^{\varsigma_i}, Y = \varphi_i(T))$ be the distinct Puiseux parametrizations of f. By Lemma 14 and Theorem 7, we have $\lim_0(V_\rho(f)) = \cup_{i=1}^{c}\{(0, y) \in \mathbb{C}^2 \mid y = \varphi_i(0)\}$. Let $(X = T^{\sigma_i}, g_i(T))$, $i = 1, \ldots, c$, be the corresponding Puiseux parametrizations of f of accuracy 1. By Theorem 9, there exists an approximation $\widetilde{f}$ of f of some finite accuracy such that $(X = T^{\sigma_i}, g_i(T))$, $i = 1, \ldots, c$, are also Puiseux parametrizations of $\widetilde{f}$ of accuracy 1. Thus, we have $\varphi_i(0) = g_i(0)$, $i = 1, \ldots, c$. Since $\widetilde{f}$ is general in Y, by Theorem 2.3 in [101], we have $\cup_{i=1}^{c}\{(0, y) \in \mathbb{C}^2 \mid y = g_i(0)\} = \{(0, y) \in \mathbb{C}^2 \mid \widetilde{f}(0, y) = 0\}$. Since $\widetilde{f}(0, y) = f(0, y)$, the lemma holds. □

Lemma 23. *Let $a(X_1, \ldots, X_s) \in \mathbb{C}[X_1, \ldots, X_s]$. Let $g_i = \sum_{j=0}^{\infty} c_{ij} T^j \in \mathbb{C}\langle T\rangle$, for $i = 1 \cdots s$. We write $a(g_1, \ldots, g_s)$ as $\sum_{k=0}^{\infty} b_k T^k$. To compute a given coefficient b_k, one only needs to know the coefficients of the polynomial a and the coefficients $c_{i,j}$ for $1 \le i \le s, 0 \le j \le k$.*

Proof. We observe that any $c_{i,j}$, where $j > k$, does not make any contribution to b_k. □

Lemma 24. *Let $f = a_d(X)Y^d + \cdots + a_0(X) \in \mathbb{C}\langle X\rangle[Y]$, where $d = \deg(f, Y) > 0$. Let $\delta := \mathrm{ord}(a_d(X))$. Then "generically", a Puiseux parametrization of f of accuracy τ can be computed from an approximation of f of accuracy $\tau + \delta$.*

Proof. Let $\widetilde{f} :=$ MakeGeneral(f). Observe that f and $\widetilde{f}$ have the same system of Puiseux parametrizations. Then the conclusion follows from Lemma 21 and 20. □

Let $R := \{r_1(X_1, X_2), \ldots, r_{s-1}(X_1, \ldots, X_s)\} \subset \mathbb{C}[X_1 < \cdots < X_s]$ be a strongly normalized regular chain. For $1 \le i \le s-1$, let $h_i := \mathrm{init}(r_i)$, $d_i := \deg(r_i, X_{i+1})$ and $\delta_i := \mathrm{ord}(h_i)$. We define $f_i, \varsigma_i, T_i, \varphi_i(T_i)$, $1 \le i \le s-1$, as follows. Let $f_1 := r_1$. Let $(X_1 = T_1^{\varsigma_1}, X_2 = \varphi_1(T_1))$ be a Puiseux parametrization of f_1. For $i = 2, \ldots, s-1$ do

(*i*) Let $f_i := r_i(X_1 = T_1^{\varsigma_1}, X_2 = \varphi_1(T_1), \ldots, X_i = \varphi_{i-1}(T_{i-1}), X_{i+1})$.

(*ii*) Let $(T_{i-1} = T_i^{\varsigma_i}, X_{i+1} = \varphi_i(T_i))$ be a Puiseux parametrization of f_i.

Before stating our main result on the bound, we first present several lemmas.

Lemma 25. *For $0 \le i \le s-2$, define $g_i(T_{s-2}) := T_{s-2}^{\prod_{k=i+1}^{s-2}\varsigma_k}$. Let $T_0 := X_1$. Then we have $T_i = g_i(T_{s-2})$, $0 \le i \le s-2$.*

Proof. We prove it by induction. Clearly it holds for $i = s-2$. Suppose it holds for i. Then we have

$$T_{i-1} = T_i^{\varsigma_i} = \left(T_{s-2}^{\prod_{k=i+1}^{s-2}\varsigma_k}\right)^{\varsigma_i} = \left(T_{s-2}^{\prod_{k=i}^{s-2}\varsigma_k}\right).$$

Therefore it also holds for $i-1$. So it holds for all $0 \le i \le s-2$. □

Lemma 26. *There exist numbers $\tau_1, \ldots, \tau_{s-2} \in \mathbb{N}$ such that in order to make f_{s-1} general in X_s, it suffices to compute the polynomial parts of φ_i of accuracy τ_i, $1 \le i \le s-2$. Moreover, if we write the algorithm* AccuracyEstimate *for short as θ, the accuracies τ_i can be computed in the following manner:*

- *let $\tau_{s-2} := (\prod_{k=1}^{s-2}\varsigma_k)\delta_{s-1} + 1$*
- *let $\tau_{i-1} := \max(\theta(f_i, \tau_i), (\prod_{k=1}^{i-1}\varsigma_k)\delta_{s-1} + 1)$, for $2 \le i \le s-2$.*

Proof. By Lemma 25, we have $g_0(T_{s-2}) = T_{s-2}^{\prod_{k=1}^{s-2}\varsigma_k}$. Since $\mathrm{ord}(h_{s-1}(X_1)) = \delta_{s-1}$, we have

$$\mathrm{ord}(h_{s-1}(X_1 = g_0(T_{s-2}))) = \left(\prod_{k=1}^{s-2}\varsigma_k\right)\delta_{s-1}.$$

Let $\tau_{s-2} := (\prod_{k=1}^{s-2}\varsigma_k)\delta_{s-1} + 1$. By Lemma 21, to make f_{s-1} general in X_s, it suffices to compute the polynomial parts of the coefficients of f_{s-1} of accuracy τ_{s-2}.

By Lemma 23, we need to compute the polynomial parts of $\varphi_i(g_i(T_{s-2}))$, $1 \le i \le s-2$, of accuracy τ_{s-2}. Since $\mathrm{ord}(g_i(T_{s-2})) = \prod_{k=i+1}^{s-2}\varsigma_k$, to achieve this accuracy, it is enough to compute the polynomial parts of φ_i of accuracy $(\prod_{k=1}^{i}\varsigma_k)\delta_{s-1} + 1$, for $1 \le i \le s-2$.

Since we have $f_i = r_i(X_1 = T_1^{\varsigma_1}, X_2 = \varphi_1(T_1), \ldots, X_i = \varphi_{i-1}(T_{i-1}), X_{i+1})$ and $(T_{i-1} = T_i^{\varsigma_i}, X_{i+1} = \varphi_i(T_i))$ is a Puiseux parametrization of f_i, by Theorem 9 and Lemma 23, to compute the polynomial part of φ_i of accuracy τ_i, we need the polynomial part of φ_{i-1} of accuracy $\theta(f_i, \tau_i)$. Thus, $\tau_{s-2} := (\prod_{k=1}^{s-2}\varsigma_k)\delta_{s-1} + 1$ and $\tau_{i-1} = \max(\theta(f_i, \tau_i), (\prod_{k=1}^{i-1}\varsigma_k)\delta_{s-1} + 1)$ for $2 \le i \le s-2$ will guarantee f_{s-1} can be made general in X_s. □

Theorem 10. *One can compute positive integer numbers $\tau_1, \ldots, \tau_{s-1}$ such that, in order to compute* $\lim_0(W(R))$*, it suffices to compute Puiseux parametrizations of f_i of accuracy τ_i, for $i = 1, \ldots, s-1$. Moreover, generically, one can choose τ_i, $i = 1, \ldots, s-1$, as follows*

- $\tau_{s-1} := 1$,
- $\tau_{s-2} := (\prod_{k=1}^{s-2} \varsigma_k)\delta_{s-1} + 1$,
- $\tau_i = (\prod_{k=1}^{s-2} \varsigma_k)(\sum_{k=2}^{s-1} \delta_i) + 1$, *for* $i = 1, \ldots, s-3$,

and each index ς_k can be set to d_k, for $k = 1, \ldots, s-2$.

Proof. By Lemma 26, we know that $\tau_1, \ldots, \tau_{s-1}$ can be computed. By Lemma 25, we have $X_1 = T_{i-1}^{\prod_{k=1}^{i-1} \varsigma_k}$. Since $\mathrm{ord}(h_i(X_1)) = \delta_i$, we have

$$\mathrm{ord}(h_i(X_1 = T_{i-1}^{\prod_{k=1}^{i-1} \varsigma_k})) = \left(\prod_{k=1}^{i-1} \varsigma_k\right)\delta_i.$$

By Lemma 24, generically a Puiseux parametrization of f_i of accuracy τ_i can be computed from an approximation of f_i of accuracy $\tau_i + \delta_i$. In Lemma 26, let $\theta(f_i, \tau_i) = \tau_i + (\prod_{k=1}^{i-1} \varsigma_k)\delta_i$, $2 \leq i \leq s-2$, which implies the bound in the theorem. Finally we observe that $\varsigma_k \leq d_k$ holds, for $1 \leq k \leq s-2$. □

4.7 Algorithm

In this section, we provide a complete algorithm for computing the non-trivial limit points of the quasi-component of a one-dimensional strongly normalized regular chain based on the results of the previous sections.

Remark 8. *Note that line* 9 *of Algorithm 4 computes Puiseux parametrizations of f_i of accuracy τ_i. Thus $(\phi(T_i), \varphi(T_i))$ at line* 10 *cannot have negative orders.*

Proposition 5. *Algorithm 5 is correct and terminates.*

Proof. This follows from Theorem 7, Theorem 9, Theorem 10 and Lemma 22. □

Theorem 11. *Let $R \subset \mathbb{Q}[X_1, \ldots, X_n]$ be a regular chain such that* $\dim(\mathrm{sat}(R)) = 1$*. Then there exists an algorithm to compute regular chains $R_i \in \mathbb{Q}[X_1, \ldots, X_n]$, $i = 1, \ldots, e$, such that* $\lim(W(R)) = \cup_{i=1}^{e} W(R_i)$.

Proof. By Remark 5, we can assume that R is strongly normalized and X_1 is free w.r.t. R. By Proposition 5, there is an algorithm to compute $\lim(W(R))$. Thus, it suffices to prove that $\lim(W(R))$ can be represented by regular chains in $\mathbb{Q}[X_1, \ldots, X_n]$, whenever $R \subset \mathbb{Q}[X_1, \ldots, X_n]$ holds. By examining carefully Algorithms 2, 3, 4, 5, and their subroutines, one observes that only Algorithms 2 and 5 may introduce numbers that are in the algebraic closure $\overline{\mathbb{Q}}$ of $\mathbb{Q}$, and not in $\mathbb{Q}$ itself. In fact, for each $x = (x_1, \ldots, x_n) \in \lim(W(R))$, Algorithms 2 and 5 introduce a field extension $\mathbb{Q}(\xi_1, \ldots, \xi_m)$ such that we have $x_i \in \mathbb{Q}[\xi_1, \ldots, \xi_m]$. Let $Y_1, \ldots, Y_m$ be m

Algorithm 4: LimitPointsAtZero

Input: $R := \{r_1(X_1, X_2), \ldots, r_{s-1}(X_1, \ldots, X_s)\} \subset \mathbb{C}[X_1 < \cdots < X_s]$, $s > 1$, is a strongly normalized regular chain.

Output: The non-trivial limit points of $W(R)$ whose X_1-coordinates are 0.

```
1  begin
2      let S := {(T_0)};
3      compute the accuracy estimates τ_1, ..., τ_{s-2} by Theorem 10; let τ_{s-1} = 1;
4      for i from 1 to s − 1 do
5          S' := ∅;
6          for Φ ∈ S do
7              f_i := r_i(X_1 = Φ_1, ..., X_i = Φ_i, X_{i+1});
8              if i > 1 then
9                  let δ := ord(f_i, T_{i-1}); let f_i := f_i / T_{i-1}^δ;
10             E := NewtonPuiseux(f_i, τ_i);
11             for (T_{i-1} = φ(T_i), X_{i+1} = ϕ(T_i)) ∈ E do
12                 S' := S' ∪ {Φ(T_{i-1} = φ(T_i)) ∪ (ϕ(T_i))}
13         S := S'
14     if S = ∅ then
15         return ∅
16     else
17         return eval(S, T_{s-1} = 0)
18 end
```

Algorithm 5: LimitPoints

Input: A strongly normalized regular chain
$R := \{r_1(X_1, X_2), \ldots, r_{s-1}(X_1, \ldots, X_s)\} \subset \mathbb{C}[X_1 < \cdots < X_s]$, $s > 1$.

Output: All the non-trivial limit points of $W(R)$.

```
1  begin
2      let h_R := init(R); let L be the set of zeros of h_R in ℂ;
3      S := ∅;
4      for α ∈ L do
5          R_α := R(X_1 = X_1 + α);
6          S_α := LimitPointsAtZero(R_α);
7          update S_α by replacing the first coordinate of every point in S_α by α;
8          S := S ∪ S_α
9      return S
10 end
```

Sys	T	#(T)	d-1	d-0	R	#(R)
f-744	14.360	4	1	3	432.567	1
Liu-Lorenz	0.412	3	3	0	216.125	3
MontesS3	0.072	2	2	0	0.064	2
Neural	0.296	5	5	0	1.660	5
Solotareff-4a	0.632	7	7	0	32.362	7
Vermeer	1.172	2	2	0	75.332	2
Wang-1991c	3.084	13	13	0	6.280	13

Table 4.1: Removing redundant components.

new symbols. Let $G := \{g_1(Y_1), g_2(Y_1, Y_2), \ldots, g_m(Y_1, Y_2, \ldots, Y_m)\}$ be an irreducible regular chain (i.e. generating a maximal ideal over $\mathbb{Q}$) such that $G(Y_1 = \xi_1, \ldots, Y_m = \xi_m) = 0$ holds. Since $x_i \in \mathbb{Q}[\xi_1, \ldots, \xi_m]$, there exists $f_i \in \mathbb{Q}[Y_1, \ldots, Y_m]$, $i = 1, \ldots, n$, such that $x_i = f_i(Y_1 = \xi_1, \ldots, Y_m = \xi_m)$. Let $\mathcal{S}_x := \{X_1 = f_1(Y_1, \ldots, Y_m), \ldots, X_n = f_n(Y_1, \ldots, Y_m), G(Y_1, \ldots, Y_m) = 0\}$. The projection of the zero set of $\mathcal{S}_x$ on the $(X_1, \ldots, X_n)$-space is the zero set of an irregular chain $R_x \in \mathbb{Q}[X_1, \ldots, X_m]$ and we have $\lim(W(R)) = \cup_{x \in \lim(W(R))} W(R_x)$. □

4.8 Experimentation

We have implemented Algorithm 5 of Section 4.7, which computes the limit points of the quasi-component of a one-dimensional strongly normalized regular chain. The implementation is based on the `RegularChains` library and the command algcurves[puiseux] [95] of MAPLE. The code is available at `http://www.orcca.on.ca/~cchen/ACM13/LimitPoints.mpl`. This preliminary implementation relies on algebraic factorization, whereas, as suggested in [33], applying the D5 principle [32], in the spirit of triangular decomposition algorithms, for instance [25], would be sufficient when computations need to split into different cases. This would certainly improve performance greatly and this enhancement is work in progress.

As pointed out in the introduction, the computation of the limit points of the quasi-component of a regular chain can be applied to removing redundant components in a Kalkbrener triangular decomposition. In Table 4.1, we report on experimental results of this application.

The polynomial systems listed in this table are one-dimensional polynomial systems selected from the literature [24, 25]. For each system, we first call the Triangularize command of the library `RegularChains`, with the option "'normalized='strongly', 'radical'='yes'"'. For the input system, this process computes a Kalkbrener triangular decomposition $\mathcal{R}$ where the regular chains are strongly normalized and their saturated ideals are radical. Next, for each one-dimensional regular chain R in the output, we compute the limit points $\lim(W(R))$, thus deducing a set of regular chains $R_1, \ldots, R_e$ such that the union of their quasi-components equals the Zariski closure $\overline{W(R)}$. The algorithm Difference [24] is then called to test whether or not there exists a pair R, R' of regular chains of $\mathcal{R}$ such that the inclusion $\overline{W(R)} \subseteq \overline{W(R')}$ holds. In Table 4.1, the columns T and #(T) denote respectively the timings spent by Triangularize and the number of regular chains returned by this command; the columns d-1 and d-0 denote respectively the number of 1-dimensional and 0-dimensional regular chains; the columns R and #(R) denote respectively the timings spent on removing redundant components in the output of

Triangularize and the number of regular chains in the output irredundant decomposition. As we can see in the table, most of the decompositions are checked to be irredundant, which we could not do before this work by means of triangular decomposition algorithms. In addition, the three redundant 0-dimensional components in the Kalkbrener triangular decomposition of system f-744 are successfully removed in about 7 minutes, whereas we cannot draw this conclusion in more than one hour by a brute-force method computing the generators of the saturated ideals of regular chains. Therefore, we have verified experimentally the benefits provided by the proposed algorithms.

4.9 Concluding remarks

We conclude with a few remarks about special cases and a generalization of the algorithms presented in this chapter.

Reduction to strongly normalized chains. Using the hypotheses of Lemma 8, we observe that one can reduce the computation of $\lim(W(R))$ to that of $\lim(W(N))$. Indeed, under the assumption that $\mathrm{sat}(R)$ has dimension one, both $\lim(W(R))$ and $\lim(W(N))$ are finite. Once the set $\lim(W(N))$ is computed, one can easily check which points in $\lim(W(N))$ do not belong to $W(R)$ and then deduce $\lim(W(R))$. This reduction to strongly normalized regular chains has the advantage that h_N is a univariate polynomial in $\mathbb{C}[X_1]$, which simplifies the presentation of the basic ideas of our algorithms, see Section 4.2.1. However, it has two drawbacks. First the coefficients of N are generally much larger than those of R. Secondly, $\lim(W(N))$ may also be much larger than $\lim(W(R))$. A detailed presentation of a direct computation of $\lim(W(R))$, without reducing to $\lim(W(N))$, will be done in a future paper.

Shape lemma case. Here, by reference to the paper [13] (which deals with polynomial ideals of dimension zero) we assume that, for $2 \leq i \leq e$, the polynomial r_i involves only the variables X_1, X_2, X_i and that $\deg(r_i, X_i) = 1$ holds. In this case, computing Puiseux series expansions is required only for the polynomial of R of lower rank, namely r_1. In this case, the algorithms presented in this chapter are much simplified. However, for the specific purpose of solving polynomial systems via triangular decompositions, reducing to this Shape lemma case, via a random change of coordinates, has a negative impact on performance and software design, for many problems of practical interest. In contrast, the point of view of the work initiated in this chapter is two-fold: first, deliver algorithms that do not require any genericity assumptions; second develop criteria that take advantage of specific properties of the input systems in order to speedup computations. Yet, in our implementation, several tricks are used to avoid unnecessary Puiseux series expansions, such as applying the theorem (see [35] p.113) on the continuity of the roots of a parametric polynomial. In this chapter, we proposed an algorithm for computing the limit points of the quasi-component of a regular chain in dimension one by means of Puiseux series expansions. In the future, we will investigate how to compute the limit points in higher dimension with the help of the Abhyankar-Jung theorem [72].

Chapter 5

Real Limit Points of Space Curves

5.1 Introduction

The work reported here is motivated by problems arising in solving polynomial systems over the real numbers. Consider the following two polynomials over $\mathbb{Q}$;

$$f_1 = -y^3 + y^2 + z^5 \text{ and } f_2 = z^4 x + y^3 - y^2. \tag{5.1}$$

For each real value of z and each real value of y satisfying both $f_1 = 0$ and $z \neq 0$, there is a real value of x satisfying $f_2 = 0$. A natural question is what happens to x when z approaches 0? This type of questions arises in the study of parametrizations of algebraic sets [88, 10] and the computation of limits of rational functions [21, 98, 104, 8].

The following question generalizes the previous ones. Given a regular chain $T \subset \mathbb{Q}[X_1, \ldots, X_n]$, denoting by h_T the product of its initials, we are interested in computing the (non-trivial) *limit points* of the set $Z_{\mathbb{R}}(T)$ consisting of the real zeros of T which do not cancel h_T. In other words, denoting by $\overline{Z_{\mathbb{R}}(T)}$ the closure of $Z_{\mathbb{R}}(T)$ in the Euclidean topology, we want to determine the set $\mathsf{lim}(Z_{\mathbb{R}}(T)) := \overline{Z_{\mathbb{R}}(T)} \setminus Z_{\mathbb{R}}(T)$. On the above example, our first question asks to compute $\mathsf{lim}(Z_{\mathbb{R}}(\{f_1, f_2\}))$. Indeed, the set $\{f_1, f_2\}$ is a regular chain for the variable ordering $z < y < x$.

With T and h_T as above, it was shown in [6], how to compute the *limit points* of the quasi-component $W(T) := V(T) \setminus V(h_T)$, that is, the set of the complex zeros of T which do not cancel h_T. In other words, denoting by $\overline{W(T)}$ the closure of $W(T)$ in the Zariski topology, the algorithms of [6] determine the set $\mathsf{lim}(W(T)) := \overline{W(T)} \setminus W(T)$. The work of [6] requires that T is a one-dimensional regular chain, that is, $\overline{W(T)}$ is a one-dimensional algebraic set. However, techniques proposed in [4] replace that assumption with weaker ones, thus allowing to deal with regular chains in higher dimension under some circumstances.

For the above example, with $T = \{f_1, f_2\}$, the set $\mathsf{lim}(W(T))$ consists of two points, as shown in Figure 5.1, with our implementation in the `RegularChains` library `www.regularchains.org`. Notably, both points have real coordinates. But only one of them belongs to $\mathsf{lim}(Z_{\mathbb{R}}(T))$. In fact, the first two Puiseux parametrizations corresponding to regular chain T around $z = 0$, as shown in Figure 5.1 and in the output of RegularChainBranches command, have RootOf(_Z^2 + 1) in their coefficients. In Maple, the notation RootOf(_Z^2 + 1) is an encoding for the complex number $\sqrt{-1}$. These latter two Puiseux parametrizations result in the first limit point $(x = 0, y = 0, z = 0)$. Thus, the point $(x = 0, y = 0, z = 0)$ can not belong to $\mathsf{lim}(Z_{\mathbb{R}}(T))$.

Therefore, $\mathsf{lim}(Z_{\mathbb{R}}(T))$ cannot be obtained simply as $\mathsf{lim}(W(T)) \cap \mathbb{R}^n$. The goal of this chapter is to explain how to adapt the algorithm of [6] in order to compute $\mathsf{lim}(Z_{\mathbb{R}}(T))$, for a one-dimensional regular chain.

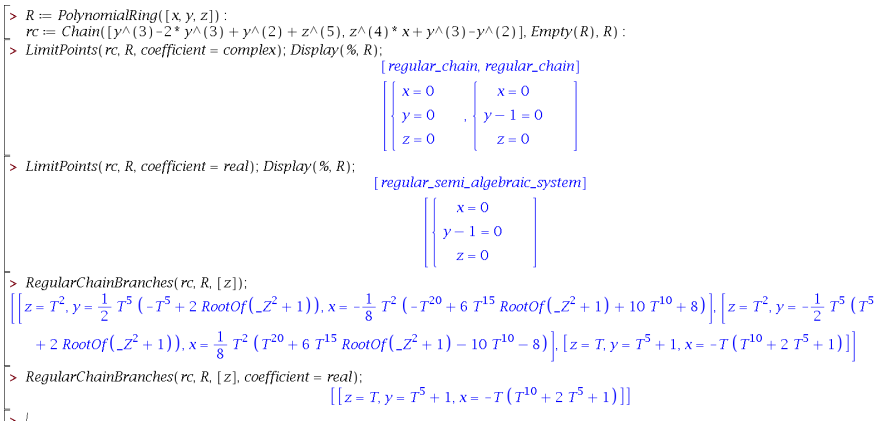


Figure 5.1: Computing the limit points of a regular chain: complex case vs real case.

The first step, made in Section 5.2.1, is to revisit the problem of determining the *real Puiseux expansions* of a bivariate polynomial. This problem is discussed in [21] with the objective of computing limits of *bivariate* rational functions. For our purpose of adapting the algorithms of [6], we need more information about the structure of Puiseux expansions of a bivariate polynomial. To this end, we rely on the *Extended Hensel Construction* of [82] as well as algorithms for computing splitting fields [93, 53]

In a second step, illustrated in Section 5.2.2 we explain how the ideas of Section 5.2.1 can help computing $\mathsf{lim}(Z_{\mathbb{R}}(T))$, that is, the real limit points of the regular chain T of dimension one.

Specially our latest work [8] (also presented in Chapter 8), the problem of computing limits of real multivariate rational functions highly depends on finding the real solutions of the closure of the regular-semi-algebraic systems in Euclidean topology. In fact, based on ideas of [98], we have proposed a method in [8] to reduce the problem of computing $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q(x_1,\ldots,x_n)$, to the problem of computing the limits of $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0),(x_1,\ldots,x_n)\in Z_{\mathbb{R}}(S_i)} q(x_1,\ldots,x_n)$ where S_i is regular semi-algebraic system of dimension one, for $i = 1,\ldots,e$, and q is a multivariate rational function. Based on Algorithm 4 in [8], one can compute $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0),(x_1,\ldots,x_n)\in Z_{\mathbb{R}}(S_i)} q(x_1,\ldots,x_n)$ for $i = 1,\ldots,e$, via computing the real limit points of regular chain part of S_i. This can be done by relying on the ideas of [6] for computing $\overline{W(T)} \setminus W(T)$ where T is a regular chain and $W(T) := V(T) \setminus V(h_T)$ with h_T being the product of the initials of the polynomials in T. As it can be seen in Example 9, it is not always possible to detect the real limit points of a regular semi-algebraic system by finding the real solutions among the limit points computed by the work described in Chapter 4 (see also [6]). The main issue with this approach is that the computations of finding the limit points of the constructible sets $W(T)$ are done with respect to Zariski topology. Thus more consideration is required in order to compute the real limit points of regular semi-algebraic systems with respect to Euclidean topology. In order to compute the non-trivial limit points of $Z_{\mathbb{R}}(S)$, for regular semi-algebraic system S of dimension one, one

needs to compute the real Puiseux parametrizations of $Z_{\mathbb{R}}(S)$ about some point, which are the parametrizations with coefficients in $\mathbb{R}$. To do so, it is enough to have a method for detecting the real Puiseux expansions of a single bivariate polynomial.

The idea of computing $\mathsf{lim}(Z_{\mathbb{R}}(T))$ is also inspired by the problem of finding a normal parametrization for surfaces. As it was mentioned in the Introduction chapter, parametric representations of surfaces are used so often specially in Computer Aided Geometric Design. However, working with parametric representation instead of the implicit representation will bring its own obstacles; it is possible that some information of the surface might be missed in the parametric representation due to missing to cover some points on the surface in the parametric representation. These missing points actually form a constructible set, since the image of the parametrization is also a constructible set. Therefore, for each parametrization, it is important to detect whether it covers all the points of the surface or "*some points*"' are actually missing. One way of dealing with this problem is to find parametrizations that cover the whole surface or in the other words a normal parametrization. For the case the surface is actually a curve, this problem is addressed in [85]. In fact, the problem of finding a normal parametrization turns to a complicated problem for general case. Up to our knowledge, this is still an open problem. Nevertheless, there are other alternatives to address this difficulty. One way to do so is the approach of [12], [86], and [87] in which the authors compute finitely many parametric representations to cover all the points on the surface, of some specific type, in their image.

Another approach to this problem is via triangular decomposition. As it was explained, triangular decomposition results in a parametric representation; however, some points might be missing in this representation. Therefore, computing the missing points of a regular chain helps to recover all the missing points in this kind of representation and encode them as new regular chains and consequently, parametric representation for the whole space curve. The methods in [6] and [4] can be used in order to compute the missing points in the parametrizations of surfaces, specially space curves.

5.2 Real limit points

Let $T \subset \mathbb{Q}[X_1, \ldots, X_n]$ be a one-dimensional regular chain; we denote by U its free variable. In [6], an algorithm is proposed for computing the non-trivial limit points of the quasi-component $W(T)$, that is, the set $\overline{W(T)} \setminus W(T)$ (where $\overline{W(T)}$ is the Zariski closure of $W(T)$). In Algorithm 4 of [8] , a similar, but different, computation is needed. In this case, we need the non-trivial limit points of $W_{\mathbb{R}}(T) := Z_{\mathbb{R}}(T) \setminus Z_{\mathbb{R}}(h_T)$, that is, the set $\overline{W_{\mathbb{R}}(T)} \setminus W_{\mathbb{R}}(T)$, where $\overline{W_{\mathbb{R}}(T)}$ is the closure of $W_{\mathbb{R}}(T)$ in $\mathbb{R}^n$ endowed with the Euclidean topology. Unfortunately, it is not true that the non-trivial limit points of $W_{\mathbb{R}}(T)$ are the non-trivial limit points of $W(T)$ with real coordinates, as shown by the example of Figure 3.2. However, the algorithm of [6], based on Puiseux series, can be adapted in order to compute the non-trivial limit points of $W_{\mathbb{R}}(T)$. This adaptation is explained hereafter. The LimitPoints command of the `RegularChains` library in MAPLE handles both cases, $\overline{W(T)} \setminus W(T)$ and $\overline{W_{\mathbb{R}}(T)} \setminus W_{\mathbb{R}}(T)$. The *Puiseux parametrizations* of the regular chain T in Definition 18 (see also [6], Definition 2) encode all the branches of $V(\mathrm{sat}(T))$ when the free variable U approaches zero. It is proved in [6] that the non-trivial limit points of $W(T)$ around $U = 0$, where $U = 0$ is a root of the product h_T of the initials of T, are obtained by letting U to be zero in all the Puiseux parametrizations of T around $U = 0$.

Therefore, for computing all the non-trivial limit points of $W(T)$, one needs to compute all the Puiseux parametrizations of T when U approaches any root of h_T.

Definition 18. *Let $T := \{t_1, \ldots, t_{n-1}\} \subset \mathbb{Q}[X_1 < \cdots < X_n]$ be a one-dimensional and strongly normalized regular chain whose free variable is $U = X_1$. Thus, the product h_T of the initials of T is a univariate polynomial in X_1. Assume that $X_1 = 0$ is a root of h_T. Let $\chi = (\chi_2(U), \ldots, \chi_n(U))$ be a vector of $\mathbb{C}((U^*))^{n-1}$ and let $\varsigma_1 = 1$. We assume that, for all $2 \leq j \leq n$, there exists a positive integer ς_j such that $(U^{\varsigma_j}, \chi_j(U))$ is a Puiseux parametrization of the univariate polynomial $t_{j-1}(U^{\varsigma_{j-1}}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j)$ around $U = 0$, where the minimum exponent of U in $\chi_j(U)$ is non-negative. Let $\varsigma := \operatorname{lcm}(\varsigma_2, \ldots, \varsigma_n)$ and $\phi_j = \chi_j(U^{\frac{\varsigma}{\varsigma_j}})$. Then $(U^{\varsigma}, \phi_2, \ldots, \phi_n)$ is called a* Puiseux parametrization *of T around $U = 0$.*

Remark 9. *If α is a root of h_T, one can define the Puiseux parametrizations of T at $X_1 = \alpha$, by reducing to the case $\alpha = 0$ via a change of coordinates and proceed as in Definition 18. For convenience, when we talk about any Puiseux parametrization of T at a root of h_T, we assume w.l.o.g that the given Puiseux parametrization is for $X_1 = 0$.*

Remark 10. *Definition 18 implies each Puiseux parametrization $(U^{\varsigma}, \phi_2, \ldots, \phi_n)$ of T belongs to $\mathbb{C}\langle U\rangle^n$.*

Example 8. *Let $T := \{t_1, t_2\} \subseteq \mathbb{Q}[X_1 < X_2 < X_3]$ be a regular chain where $t_1 := X_2^4 - 2\,X_2^3 + X_2^2 + X_1^5$ and $t_2 := X_1^4\,X_3 + X_2^3 - X_2^2$. We note that the product of the initials of T is $h_T := X_1^4$. We would like to compute the Puiseux parametrizations of the regular chain T around $X_1 = 0$. Using the* `ExtendedHenselConstruction` *command of our library* `PowerSeries` [1]*, one can compute the Puiseux parametrizations of t_1 around $X_1 = 0$ and obtain:*

$$\begin{aligned}
\Phi_1 &:= \left(X_1 = U^2, X_2 = 1 + \sqrt{-1}\,U^5 + U^{10} + O(U^{15})\right),\\
\Phi_2 &:= \left(X_1 = U^2, X_2 = 1 - \sqrt{-1}\,U^5 + U^{10} + O(U^{15})\right),\\
\Phi_3 &:= \left(X_1 = U^2, X_2 = \sqrt{-1}\,U^5 - U^{10} + O(U^{15})\right),\\
\Phi_4 &:= \left(X_1 = U^2, X_2 = -\sqrt{-1}\,U^5 - U^{10} + O(U^{15})\right).
\end{aligned} \tag{5.2}$$

The big-oh notation is used above to indicate at which degree the displayed power series are truncated.

Now by substituting Φ_1 into t_2, we obtain $t_{21} := U^8\,X_3 + (1 + \sqrt{-1}\,U^5 + U^{10})^3 - (1 + \sqrt{-1}\,U^5 + U^{10})^2$. Next, we compute Puiseux parametrizations of t_{21} around $U = 0$ and obtain:

$$\left(U = U, X_3 = -\frac{\sqrt{-1}}{U^4} + U^2 - 3\,\sqrt{-1}\,U^7 + O(U^8)\right).$$

Since there is a negative exponent of U appearing in the above Puiseux parametrization of t_{21}, this parametrization would not result in a Puiseux parametrization for the regular chain T. By repeating the same process with Φ_2, one obtains a Puiseux parametrization in which negative exponents of U appear as well. However, the scenario is different when substituting Φ_3 into

[1] This library is freely available from `www.regularchains.org`

t_2. Indeed, this substitution yields $t_{23} := U^8 X_3 + (\sqrt{-1}\, U^5 - U^{10})^3 - (\sqrt{-1}\, U^5 - U^{10})^2$, whose Puiseux parametrization around $U = 0$ is

$$X_3 = -U^2 \left(-U^{20} + 3\ \sqrt{-1}\, U^{15} + 2\, U^{10} + \ \sqrt{-1}\, U^5 + 1 + O(U^{25})\right).$$

Since there is no negative exponents of U in the latter Puiseux parametrization, we deduce the following Puiseux parametrization of the regular chain T:

$$\begin{aligned}\Phi_{2,3} := (&X_1 = U^2, X_2 = \sqrt{-1}\, U^5 - U^{10} + O(U^{15}),\\ &X_3 = U^{22} - 3\ \sqrt{-1}\, U^{17} - 2\, U^{12} - \ \sqrt{-1}\, U^7 - U^2 + O(U^{27}))\end{aligned}$$

Proceeding similarly with Φ_4, one obtains the second Puiseux parametrization of T:

$$\begin{aligned}\Phi_{2,4} := (&X_1 = U^2, X_2 = -\sqrt{-1}\, U^5 - U^{10} + O(U^{15}),\\ &X_3 = U^{22} + 3\ \sqrt{-1}\, U^{17} - 2\, U^{12} + \ \sqrt{-1}\, U^7 - U^2 + O(U^{27}))\end{aligned}$$

In both Puiseux parametrizations of regular chain T, the ramification index is $\varsigma = \mathrm{lcm}(2, 1) = 2$.

Definition 19. *Using the notations of Definition 18, the Puiseux parametrization $(U^\varsigma, \phi_2, \ldots, \phi_n)$ is called a* real Puiseux parametrization *of T if $\phi_i \in \mathbb{R}\langle U\rangle$, for $i = 2, \ldots, n$.*

Example 9. *Let T be again the regular chain in Example 8 with Puiseux parametrizations $\Phi_{2,3}, \Phi_{2,4}$ at $U = 0$. Then by substituting $U = 0$ in $\Phi_{2,3}, \Phi_{2,4}$, we obtain one non-trivial limit point for $W(T)$, namely $\{(X_1 = 0, X_2 = 0, X_3 = 0)\}$, for which its coordinates are real. However, none of the branches of the space curve defined by $W(T)$ is real. Hence, $W_{\mathbb{R}}(T)$ has no non-trivial limit points!*

As we shall see, one can obtain the non-trivial limit points of $Z_{\mathbb{R}}(T)$ by computing the real Puiseux parametrizations of $Z_{\mathbb{R}}(T)$ when its free variable approaches any root of h_T. To do so, it is enough to have a method for detecting the real Puiseux expansions of a single polynomial. As it is explained in [21], when T only contains one bivariate polynomial, one way of separating the real Puiseux expansions from the complex ones, is to detect for which initial factors of the method of computing the Puiseux expansions, complex coefficients will appear in the computations. However, no method is proposed for general cases when T has more than one polynomial. In Section 5.2.1, we propose Algorithm 6 for computing real Puiseux parametrizations of regular chains of dimension one.

5.2.1 Real branches of bivariate polynomials

Proposition 6. *Let $\mathbf{k}$ be an algebraic number field and $f(U, Y) \in \mathbf{k}\langle U\rangle[Y]$ be square-free, monic w.r.t Y, and of degree $s > 0$ in Y. Then, for each $\ell = 1, \ldots, s$, one can compute a positive integer σ_ℓ as well as algebraic numbers $\Theta_\ell^1, \ldots, \Theta_\ell^{\sigma_\ell}$ over $\mathbf{k}$ such that*

1. *for $i = 1, \ldots, \sigma_\ell$, the algebraic number Θ_ℓ^i has a minimal polynomial of the form $h_\ell^i(Y) \in \mathbf{k}(\Theta_\ell^1, \ldots, \Theta_\ell^{i-1})[Y]$,*
2. *$f(U, Y)$ factorizes as $(Y - \chi_1(U)) \cdots (Y - \chi_s(U))$ where $\chi_\ell(U) \in \mathbf{k}(\Theta_\ell^1, \ldots, \Theta_\ell^{\sigma_\ell})((U^*))$., $i = 1, \ldots, \sigma_\ell$.*

Proof. Based on Theorem 1, the existence of expansions $\chi_1(U), \ldots, \chi_s(U)$ is guaranteed.

To prove this proposition, we should show that there exist algebraic numbers $\Theta_\ell^1, \ldots, \Theta_\ell^{\sigma_\ell}$ over $\mathbf{k}$ such that $\chi_\ell(U) \in \mathbf{k}(\Theta_\ell^1, \ldots, \Theta_\ell^{\sigma_\ell})((U^*))$, for $\ell = 1, \ldots, s$.

Let $f_0 := F(U, Y)$. Then based on Theorem 1, there are $f_1, \ldots, f_{\sigma_\ell} \in \mathbb{C}((U^*))[Y]$ such that $f_i \in EHC(f_{i-1})$, for $i = 1, \ldots, \sigma_\ell$, and $f_{\sigma_\ell} = Y - \chi_\ell(U)$.

Based on Lemma 5 and Corollary 1, there is at most one algebraic number Θ_ℓ^1 with minimal polynomial $h_\ell^1(Y) \in \mathbf{k}[Y]$ (which is, indeed, computed by substituting $U = 1$ in Newton polynomial of f_0) such that $\mathbf{k}_1 := \mathbf{k}(\Theta_\ell^1)$ and $f_1 \in \mathbf{k}_1((U^*))[Y]$. Since EHC is applied recursively on polynomials f_i, thus there are algebraic numbers Θ_ℓ^i with minimal polynomials $h_\ell^i(Y) \in \mathbf{k}_{i-1}[Y]$, where $\mathbf{k}_i := \mathbf{k}_{i-1}(\Theta_\ell^i)$ and $f_i \in \mathbf{k}_i((U^*))[Y]$, for $i = 2, \ldots, \sigma_\ell$. Therefore, $f_{\sigma_\ell} \in \mathbf{k}_{\sigma_\ell}((U^*))[Y]$ and consequently, $\chi_\ell(Y) \in \mathbf{k}_{\sigma_\ell}((U^*)) = \mathbf{k}(\Theta_\ell^1, \ldots, \Theta_\ell^{\sigma_\ell})((U^*))$.

Note that EHC sometimes does a change of coordinates on the input polynomial, but this change of coordinates is within the coefficient ring of its input polynomial. Thus it does not introduce any new algebraic number. □

Proposition 6 follows from the *extended Hensel construction*. This proposition shows that there is a finite extension of $\mathbf{k}$ for which $f(U, Y)$ can be written as $(Y - \chi_1(U)) \cdots (Y - \chi_s(U))$, and therefore all the coefficients of the Puiseux expansions of f are determined. Especially, when $\mathbf{k} = \mathbb{Q}$, then determining whether or not $\chi_\ell(U)$ is a real Puiseux expansion is equivalent to the fact that each Θ_ℓ^{i-1} is a real algebraic number over $\mathbb{Q}(\Theta_\ell^1, \ldots, \Theta_\ell^{i-1})$, for $i = 1, \ldots, \sigma_\ell$.

Furthermore, based on the construction of $h_\ell^i(Y)$, all of the roots of each polynomial $h_\ell^i(Y)$ will appear in some of Puiseux expansions of $f(U, Y)$. Since some of those roots may not be real, it is necessary to use an encoding of the roots of $h_\ell^i(Y)$ that allows us to separate the real ones from the others. For simplicity of presentation, we do that by considering the splitting fields of $h_\ell^i(Y)$, see Remark 11. For computational efficiency, one should prefer techniques based on *real algebraic closures* as in [76].

Let $h(Y) \in \mathbf{k}[Y]$ be an irreducible and monic polynomial with degree s. Let also $\frac{\mathbf{k}[X_1, \ldots, X_n]}{\langle F \rangle}$ be the residue class ring of $\mathbf{k}[X_1, \ldots, X_n]$ with respect to F, where $F \subset \mathbf{k}[X_1, \ldots, X_n]$.

Remark 11. *To construct the splitting field* $\mathbf{L}$ *of* $h(Y)$ *and compute the factorization of* $h(Y)$ *into linear factors over* $\mathbf{L}$*, one can proceeds as follows.*

1. *Initialize* $i := 1$, $X_i := Y$, $\mathbf{L} := \mathbf{k}$, $R_0 := \{\}$, $\mathcal{P} := \{\}$ *and* $\mathcal{F} := \{h(Y)\}$*; the set* $\mathcal{F}$ *is assumed to maintain a list of univariate polynomials in* Y_i *irreducible over* $\mathcal{L}$ *and of degree at least two,*
2. *While* $\mathcal{F}$ *is not empty do*
   (a) *pick a polynomial* $f(X_i) \in \mathcal{F}$ *over* $\mathbf{L}$*,*
   (b) *let* α_i *be a root of* $f(X_i)$ *(in the algebraic closure of* $\mathbf{k}$*),*
   (c) *replace* $\mathbf{L}$ *by* $\mathbf{L}(\alpha_i)$*, that is, by adjoining* α_i *to* $\mathbf{L}$*,*
   (d) *replace* T *by* $R_i := R_{i-1} \cup \{r_i(X_1, \ldots, X_i)\}$*, where the multivariate* $r_i(X_1, \ldots, X_i)$ *is obtained from* $f(X_i)$ *after replacing the algebraic numbers* $\alpha_1, \ldots, \alpha_{i-1}$ *with the variables* $X_1, \ldots, X_{i-1}$*,*
   (e) *factor* $f(X_i)$ *into irreducible factors over* $\mathbf{L}$*, then add the obtained factors of degree 1 (resp. greater than 1) to* $\mathcal{P}$ *(resp.* $\mathcal{F}$*); when adding a factor to* $\mathcal{P}$ *replace* X_i *with* Y*;when adding a factor to* $\mathcal{F}$*, replace* X_i *with* X_{i+1} *and* $\alpha_1, \ldots, \alpha_{i-1}, \alpha_i$ *with* $X_1, \ldots, X_{i-1}, X_i$*,*
   (f) *if* $\mathcal{F}$ *is not empty then* $i := i + 1$.

3. Let $s' := i$.

At the end of this procedure, the set $R_{s'}$ *is a regular chain in the polynomial ring* $\mathbf{k}[X_1, \ldots, X_{s'}]$ *generating a maximal ideal such that* $\mathbf{k}[X_1, \ldots, X_{s'}]/\langle R_{s'} \rangle$ *is isomorphic to the splitting field* $\mathbf{k}(p)$ *of* $h(Y)$. *Furthermore,*

$$\mathbf{k}[Y] \subset \frac{\mathbf{k}[X_1]}{\langle R_1 \rangle}[Y] \subset \cdots \subset \frac{\mathbf{k}[X_1, \ldots, X_{s'}]}{\langle R_{s'} \rangle}[Y].$$

Algorithm 6: RealPuiseuxExpansions

Input: $f(U, Y) \in \mathbf{k}[U, Y]$.
Output: Real Puiseux expansions of f when $U \to 0$

```
1  begin
2      B := Puiseux expansions of f(U, Y) at U = 0;
3      R := {};
4      for χ(U) ∈ B do
5          let χ(U) ∈ k(Θ^1, ..., Θ^σ)((U*));
6          let R^i_{j_i} ⊂ k[X_{i,1}, ..., X_{i,j_i}] be the zero-dimensional regular chain encoding the
           algebraic number Θ^i, for i = 1, ..., σ;
7          let C be a regular chain encoding the field k;
8          F := C ∪ R^1_{j_1} ∪ ··· ∪ R^σ_{j_σ}
9          if RealTriangularize(F) ≠ ∅ then
10             R := R ∪ {χ(U)};
11     return R;
12 end
```

Using regular chains $R_1, \ldots, R_{s'}$ in Remark 11, one can encode all the solutions of polynomial $h(Y)$, "uniquely". It is worth mentioning that the `Split` command of the `PolynomialTools` package in MAPLE computes the regular chains $R_1, \ldots, R_{s'}$, implicitly.

Example 10. *Suppose* $h(Y) := Y^3 + Y^2 + 3$. *Using* `Split` *command in* MAPLE, *one obtains* $R_1 := \{X_1^3 + X_1^2 + 3\}$ *and* $R_2 := R_1 \cup \{X_2^2 + (1 + X_1) X_2 + X_1^2 + X_1\}$, *for which* $\frac{\mathbb{Q}[X_1, X_2]}{\langle R_2 \rangle}[Y]$ *is the splitting field of* $h(Y)$. *Using the command* `RealTriangularize` *of the* `RegularChains` *library, we can check that* $Z_{\mathbb{R}}(R_1)$ *contains one real solution, while the set* $Z_{\mathbb{R}}(R_2)$ *does not have any real solutions.*

Definition 20. *Let* Θ *be a root of* $h(Y)$. *Let also* j *be the smallest integer for which* $\Theta \in \frac{\mathbf{k}[X_1, \ldots, X_j]}{\langle R_j \rangle}$, *then* R_j *is called a* regular chain encoding of Θ.

Following up on Definition 20, determining whether or not Θ is a real algebraic number over $\mathbf{k}$ is equivalent to check whether or not $Z_{\mathbb{R}}(R_j)$ has a real solution or not over $\mathbf{k}$. Furthermore, $\mathbf{k}$ must be a real extension of $\mathbb{Q}$. To make sure that a polynomial system has real solutions, one can use the RealTriangularize command of the `RegularChains` Library in MAPLE. In fact, the command RealTriangularize computes the real solutions of the polynomial system defined by F, where $F \subset \mathbb{Q}[X_1, \ldots, X_n]$. Thus, for checking whether or not Θ is a real algebraic

number over $\mathbf{k}$, using RealTriangularize, more considerations are required due to the constraint imposed by the coefficient ring. To remove this constraint, since $\mathbf{k}$ is an algebraic extension of $\mathbb{Q}$, thus one can compute a zero-dimensional regular chain $C \subset \mathbb{Q}[Y_1, \ldots, Y_m]$ (for some m) such that $\frac{\mathbb{Q}[Y_1,\ldots,Y_m]}{\langle C\rangle}$ is isomorphic to $\mathbf{k}$. This means that one can apply RealTriangularize on the system defined by $C \cup R_j \subset \mathbb{Q}[Y_1, \ldots, Y_m, X_1, \ldots, X_j]$; if this system has real solutions, then we deduce that Θ is a real algebraic number over $\mathbf{k}$.

Algorithm 6 implements the above idea and computes the real Puiseux expansions of the bivariate polynomial f at $U = 0$. This algorithm, first, computes all of the Puiseux expansions of the polynomial f at $U = 0$ and then determines which one is a real expansion for f.

5.2.2 Real branches of space curves

Proposition 7. *With the notations of Definition 18, consider the Puiseux parametrization $(U^{s}, \phi_2, \ldots, \phi_n)$ of the regular chain T around $U = 0$. Then, for each $j = 2, \ldots, n$, one can compute algebraic numbers $\Theta_j^1, \ldots, \Theta_j^{\sigma_j}$ over $\mathbf{k}_{j-1}$ such that $\phi_j(U) \in \mathbf{k}_j[U]$, where $\mathbf{k}_1 := \mathbb{Q}$ and $\mathbf{k}_j := \mathbf{k}_{j-1}(\Theta_j^1, \ldots, \Theta_j^{\sigma_j})$ for some non-negative integer σ_j.*

Proof. To prove this proposition, it is enough to prove that $\chi_j(U) \in \mathbf{k}_j[U]$, for $j = 2, \ldots, n$. We prove this by induction on j. For $j = 2$, $(U^{s}, \chi_2(U))$ is a Puiseux parametrization of the bivariate polynomial $t_1(U, X_2)$ around $U = 0$. Since $t_1(U, X_2) \in \mathbb{Q}[U, X_2]$, thus according to Proposition 6, there exist algebraic numbers $\Theta_2^1, \ldots, \Theta_2^{\sigma_2}$ over $\mathbb{Q}$ such that $\chi_2(U) \in \mathbb{Q}(\Theta_2^1, \ldots, \Theta_2^{\sigma_2})$. Suppose $\chi_{j-1}(U) \in \mathbf{k}_{j-1}[U]$. Thus, $\chi_j(U)$ is a Puiseux expansion of bivariate polynomial

$$t_{j-1}(U^{s_{j-1}}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j)$$

which, in turn, belongs to $\mathbf{k}_{j-1}[U, X_j]$ by induction hypothesis step. Based on Proposition 6, there exists algebraic numbers $\Theta_j^1, \ldots, \Theta_j^{\sigma_j}$ over $\mathbf{k}_{j-1}$ such that $\chi_i(U) \in \mathbf{k}_{j-1}(\Theta_j^1, \ldots, \Theta_j^{\sigma_j})[U]$. Now if we let $\mathbf{k}_j := \mathbf{k}_{j-1}(\Theta_j^1, \ldots, \Theta_j^{\sigma_j})$, this completes the proof. □

Proposition 8. *Following up on Proposition 7, the Puiseux parametrization*

$$(U^{s}, \phi_2(U), \ldots, \phi_n(U))$$

is a real Puiseux parametrization of T if and only if $\mathbf{k}_n$ is a real extension of $\mathbb{Q}$.

Proof. The correctness of the relation $\mathbf{k}_1 := \mathbb{Q} \subseteq \mathbf{k}_2 \subseteq \cdots \subseteq \mathbf{k}_n$ is trivial based on the constructive proof of Proposition 7. Thus for determining whether or not the Puiseux parametrization $(U^{s}, \phi_2, \ldots, \phi_n)$ is real, it is enough to check if $\mathbf{k}_n$ is a real extension over $\mathbb{Q}$. □

Algorithm 7 computes the real Puiseux parametrizations corresponding to regular chain T. Based on Definition 18, for computing the Puiseux parametrizations of T, one needs to compute the Puiseux parametrizations $(U^{s_j}, \chi_j(U))$ of the bivariate polynomial

$$t_{j-1}(U^{s_{j-1}}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j),$$

for $j = 2, \ldots, n$. If any of such parametrization has complex coefficients, then it would not result in a real Puiseux parametrization for regular chain T. Thus, we should only consider the real Puiseux parametrizations of bivariate polynomials $t_{j-1}(U^{s_{j-1}}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j)$. To

Algorithm 7: RealRegularChainBranches

Input: one dimensional regular chain T with free variable U.
Output: Real Puiseux parametrizations of T when $U \to 0$

1 **begin**
2 $\quad \mathcal{R} := \{\}$;
3 $\quad \varsigma_1 = 1$;
4 $\quad$ **for** j **from** 2 **to** n **do**
5 $\quad\quad \mathcal{R}_j := \{\}$;
6 $\quad\quad$ **for** $(\chi_2(U), \ldots, \chi_{j-1}(U)) \in \mathcal{R}$ **do**
7 $\quad\quad\quad \mathcal{B} :=$ `RealPuiseuxExpansions`$(t_{j-1}(U^{\varsigma_{j-1}}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j), 0)$;
8 $\quad\quad\quad \varsigma_j =$ `RamificationIndex`$(t_{j-1}(U^{\varsigma_{j-1}}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j), 0)$;
9 $\quad\quad\quad$ **if** $\mathcal{B} \neq \emptyset$ **then**
10 $\quad\quad\quad\quad$ let $\mathcal{B} := \{\chi_j^1(U), \ldots, \chi_j^{\ell_j}(U)\}$;
11 $\quad\quad\quad\quad \mathcal{R}_j :=$
$\quad\quad\quad\quad \mathcal{R}_j \cup \left\{\left(\chi_2(U), \ldots, \chi_{j-1}(U), \chi_j^1(U)\right), \ldots, \left(\chi_2(U), \ldots, \chi_{j-1}(U), \chi_j^{\ell_j}(U)\right)\right\}$;
12 $\quad\quad \mathcal{R} := \mathcal{R}_j$;
13 $\quad$ **return** $\mathcal{R}$;
14 **end**

do so, in Algorithm 7 at line 7, we call Algorithm 6 for computing the real Puiseux expansions of bivariate polynomials to filter out the expansions that would not contribute in building a real Puiseux parametrization for regular chain T.

Based on Definition 18, for computing the Puiseux parametrizations of T, one needs to compute the Puiseux parametrizations $(U^{\varsigma_j}, \chi_j(U))$ of the bivariate polynomial

$$t_{j-1}(U^{\varsigma_{j-1}}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j),$$

for $j = 2, \ldots, n$. If any of such parametrization has complex coefficients, then it would not result in a real Puiseux parametrization for regular chain T. Thus, we should only consider the real Puiseux parametrizations of bivariate polynomials $t_{j-1}(U^{\varsigma_{j-1}}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j)$. To do so, one needs to use Algorithm 6, successively.

5.3 Experimentation

Table 5.1 demonstrates the time consumptions for computing real and complex limit points corresponding to the regular chains of dimension one in the triangular decomposition of the polynomial systems in the first column. The second and third columns are respectivly, the time spent for computing and the number of real limit points. The fourth and fifth columns are respectivly, the time spent for computing and the number of complex limit points. The command for computing real and complex limit points of regular chains is called LimitPoints and it is part of `RegularChains` library of MAPLE. The command LimitPoints has two different signatures, LimitPoints$(R, coefficient = real)$ and LimitPoints$(R, coefficient = complex)$ for computing, respectively, the real and complex limit points of the input regular chain R.

Sys	RealLimit (sec)	#LM	ComplexLimit (sec)	#LM
Liu-Lorenz	777.300	4	1708.829	9
MontesS3	0.015	0	0.015	0
Neural	1.538	3	2.368	3
cox-issac07	0.438	0	0.575	1

Table 5.1: Complex limit points vs real limit points

Chapter 6

Computing Limit Points via Changes of Coordinates

6.1 Introduction

Applying a change of coordinates to the algebraic or differential representation of a geometrical object is a fundamental technique to obtain a more convenient representation and reveal properties. For instance, random linear change of coordinates are performed in algorithms for solving systems of polynomial equations and inequations in the algorithms of Krick and Logar [50], Rouillier [79], Verschelde [92] and Lecerf [56].

For polynomial ideals, one desirable representation is Noether normalization, which was introduced by Emmy Noether in 1926. This basic tool in commutative algebra and algebraic geometry is widely used to study different topics such as affine K-algebras [39], dimension theory [38], solving systems of polynomial equations and inequations [56] and radical of ideals [50]. In the last two decades, several algorithms have been proposed to compute Noether normalization of a quotient ring (see [60, 39, 14, 42, 78]). However, all these algorithms use random changes of coordinates and most of them use Gröbner bases w.r.t. lexicographical ordering which may lead to large coefficient growth. In this direction, a general algorithm was given by Vasconcelos [97]. Logar [60] proposed a probabilistic algorithm based on Gröbner bases calculations to compute Noether normalization corresponding to prime ideals. Further, Greuel et al. [39] proposed a general algorithm using random triangular linear changes of variables and also Gröbner bases. Bermejo et al. [14] presented another method by applying sparser random triangular linear changes of variables and Gröbner bases. In [42], Hashemi provided an algorithm which uses an incremental random linear changes of variables without computing the dimension of the input ideal. Finally, Robertz [78] described a new approach based on Janet bases and Satnley decompositions. We give also the references [84, 43] to deterministic approaches to compute Noether normalization. We shall note that in this chapter, we apply the algorithm presented in [43]. For more details on Noether normalization, we refer to the books [39, 34].

In regular chain theory, one desirable and challenging objective is, given a regular chain T, to obtain the (non-trivial) limit points of its quasi-component $W(T)$, or equivalently, computing the variety of its saturated ideal $\mathrm{sat}(T)$. The set $\lim(W(T))$ of the non-trivial limit points of

$W(T)$ satisfies $V(\mathrm{sat}(T)) = \overline{W(T)} = W(T) \cup \lim(W(T))$. Hence, $\lim(W(T))$ is the set-theoretic difference $V(\mathrm{sat}(T)) \setminus W(T)$. Deducing $\lim(W(T))$ or $V(\mathrm{sat}(T))$ from T is a central question which has theoretical applications (like the so-called *Ritt Problem*) and practical ones (like removing redundant components in triangular decomposition).

Of course $V(\mathrm{sat}(T))$ can be computed from T via Gröbner basis techniques. But this approach is of limited practical interest. In fact, considering the case where the base field is $\mathbb{Q}$, we are looking for approaches that would run in polynomial time w.r.t. the degrees and coefficient heights of T. Thanks to the work of [31], algorithms for change of variable orders (and more generally, algorithms for linear changes of coordinates) are good candidates. Indeed [31] shows that change of variable orders for regular chains can be done in polynomial time w.r.t input data size. The articles [18, 20] propose an alternative method for the same task, called PALGIE. Both algorithms are part of the `RegularChains` library in MAPLE and experimentally, the second one performs better, while its algebraic complexity is unknown.

Returning to Noether normalization[1], we ask in Section 6.4 how "simple" can T be if we assume that $\mathrm{sat}(T)$ is in Noether position. Unfortunately, an additional hypothesis is needed in order to obtain a satisfactory answer like "all initials of T are constant", see Theorem 13 and Remark 12.

In Section 6.5 and 6.6, we develop a few criteria for computing $\lim(W(T))$ or $V(\mathrm{sat}(T))$. Our techniques (see Proposition 18, Theorem 14, Theorem 15, Theorem 16 and Algorithm 15) rely on linear changes of coordinates and allow us to relax the "dimension one" hypothesis in our previous paper [6], where $\lim(W(T))$ was computed via Puiseux series.

Therefore, the techniques proposed in this chapter can be used to compute $\lim(W(T))$ or $V(\mathrm{sat}(T))$ without Gröbner basis or Puiseux series calculations. Moreover, these new techniques can handle cases where the results of our previous paper [6] could not apply. One of the main ideas of our new results (see for instance Theorem 14) is to use a linear change of coordinates so as to replace the description of $\overline{W(T)}$ by one for which $\overline{W(T)} \cap V(h_T)$ can be computed by means of standard operations on regular chains. Nevertheless, our proposed techniques do not cover all possible cases and the problem of finding a "Gröbner-basis-free" general algorithm for $\lim(W(T))$ or $V(\mathrm{sat}(T))$ remains unsolved.

6.2 Preliminaries

Throughout this chapter, polynomials have coefficients in a field $\mathbf{k}$ and variables in a set $\mathbf{x}$ of n ordered variables $x_1 < \cdots < x_n$. The corresponding polynomial ring is denoted by $\mathbf{k}[\mathbf{x}]$. Let F be a subset of $\mathbf{k}[\mathbf{x}]$. We denote by $\langle F \rangle$ the ideal generated by F in $\mathbf{k}[\mathbf{x}]$. Recall that a polynomial $f \in \mathbf{k}[\mathbf{x}]$ is *regular* modulo the ideal $\langle F \rangle$ whenever f does not belong to any prime ideals associated with $\langle F \rangle$, thus, whenever f is neither null nor a zero-divisor modulo $\langle F \rangle$. Further, $\overline{\mathbf{k}}$ stands for the algebraic closure of $\mathbf{k}$ and $V(F) \subset \overline{\mathbf{k}}^n$ for the algebraic set consisting of all common zeros of all $f \in F$. For a set $W \subset \overline{\mathbf{k}}^n$, we denote by $\overline{W}$ the *Zariski closure* of W, that is, the intersection of all algebraic sets containing W.

We briefly review standard notions and concepts related to regular chains and we refer to [11, 25] for details. For a non-constant $f \in \mathbf{k}[\mathbf{x}]$, we denote by $\mathrm{mvar}(f)$, $\mathrm{mdeg}(f)$ and

[1] Section 6.4 contains a brief review of Noether normalization which makes this chapter self-contained.

init(f), the variable of greatest rank appearing in f, the degree of f w.r.t. that variable and the leading coefficient of f w.r.t. that same variable. The quantities mvar(f), mdeg(f) and init(f) are called respectively the *main variable*, *main degree* and *initial* of f. A set T of non-constant polynomials from $\mathbf{k}[\mathbf{x}]$ is called *triangular* if no two polynomials from T have the same main variable. Let $T \subset \mathbf{k}[\mathbf{x}]$ be a triangular set. Observe that T is necessarily finite and that every subset of T is itself triangular. For a variable $v \in \mathbf{x}$, if there exists $f \in T$ such that mvar(f) = v, we denote this polynomial by T_v and say that v is *algebraic* w.r.t. T, otherwise we say that v is *free* w.r.t. T; in all cases, we define $T_{<v} := \{g \in T \mid \text{mvar}(g) < v\}$ and denote by $\mathsf{free}(T)$ the set of the variables from $\mathbf{x}$ which are free w.r.t. T. We denote by h_T the product of the polynomials init(f), for $f \in T$. We say that T is *strongly normalized* if all variables occurring in h_T are in $\mathsf{free}(T)$; when this holds, it is easy to check that T is a Gröbner basis of the ideal that T generates in $\mathbf{k}(\mathbf{u})[\mathbf{x} \setminus \mathbf{u}]$ where $\mathbf{u} := \mathsf{free}(T)$ and $\mathbf{k}(\mathbf{u})$ is the field of rational functions over $\mathbf{k}$ and with variables in $\mathbf{u}$. Moreover, we say that T is *monic* whenever $h_T \in \mathbf{k}$ holds. The *saturated ideal* of T, written sat(T), is defined as the colon ideal $\text{sat}(T) = \langle T \rangle : h_T^\infty$. The *quasi-component* of T is the basic constructible set given by $W(T) := V(T) \setminus V(h_T)$. The following two properties are easy to prove:

$$\overline{W(T)} \;=\; V(\text{sat}(T)) \quad \text{and} \quad \overline{W(T)} \;=\; W(T) \;\cup\; \lim(W(T)), \tag{6.1}$$

where $\lim(W(T)) := \overline{W(T)} \cap V(h_T)$ holds and the points of that latter set are called the *(non-trivial) limit points* of $W(T)$, for reasons explained in [6]. We say that T is a *regular chain* whenever T is empty or $T_{<w}$ is a regular chain and the initial of T_w is regular modulo $\text{sat}(T_{<w})$, where w is the largest main variable of a polynomial in T. If T consists of $n - d$ polynomials, for $0 \leq d < n$, then sat(T) has dimension d and either $\lim(W(T))$ is empty or has dimension $d - 1$; moreover, we have $\mathbf{k}[\mathbf{u}] \cap \text{sat}(T) = \langle 0 \rangle$, where $\mathbf{u} := \mathsf{free}(T)$.

Let $F \subset \mathbf{k}[\mathbf{x}]$ be finite. Let $T_1, \ldots, T_e$ be finitely many regular chains of $\mathbf{k}[\mathbf{x}]$. We say that $\{T_1, \ldots, T_e\}$ is a *Kalkbrener triangular decomposition* of $V(F)$ if we have $V(F) = \cup_{i=1}^e \overline{W(T_i)}$. We say that $\{T_1, \ldots, T_e\}$ is a *Lazard-Wu triangular decomposition* of $V(F)$ if we have $V(F) = \cup_{i=1}^e W(T_i)$.

We call *affine change of coordinates in* $\overline{\mathbf{k}}^n$ any bijective map A of the form

$$\begin{array}{rccc} A: & \overline{\mathbf{k}}^n & \to & \overline{\mathbf{k}}^n \\ & \mathbf{x} & \longmapsto & (A_1(\mathbf{x}), \ldots, A_n(\mathbf{x})) \end{array} \tag{6.2}$$

where $A_1, \ldots, A_n$ are affine forms over $\overline{\mathbf{k}}$. Hence $A(\mathbf{x})$ can be written as $M\mathbf{x} + \mathbf{b}$ where M is an invertible matrix over $\overline{\mathbf{k}}$ and $\mathbf{b} \in \overline{\mathbf{k}}^n$. If $\mathbf{b}$ is null, we call A a *linear change of coordinates in* $\overline{\mathbf{k}}^n$. For the algebraic set $V(F)$, we denote

$$V^A(F) \;:=\; V(\{f^A \mid f \in F\}), \tag{6.3}$$

where $f^A(\mathbf{x}) := f(A_1(\mathbf{x}), \ldots, A_n(\mathbf{x}))$. Observe that if $V(F)$ is irreducible, then so is $V^A(F)$. Similarly, the image of $W(T)$ under A is

$$W^A(T) \;=\; V^A(T) \setminus V^A(h_T). \tag{6.4}$$

6.3 Algorithm for linear change of coordinates

The goal of this section is to present a practically efficient algorithmic solution to the following problem.

Problem 1. *Given a regular chain $T \subset \mathbf{k}[\mathbf{x}]$ and given a linear change of coordinates A in $\overline{\mathbf{k}}^n$, compute finitely many regular chains $C_1, \ldots, C_e$ such that*

$$\overline{W^A(T)} \;=\; \overline{W(C_1)} \cup \cdots \cup \overline{W(C_e)}.$$

In the literature, see [18, 20, 31], the following related problem has been addressed.

Problem 2. *Given two total orderings $\mathcal{R}$ and $\overline{\mathcal{R}}$ on $\{x_1, \ldots, x_n\}$, given $T \subset \mathbf{k}[x_1, \ldots, x_n]$, assuming that*

1. *T is a regular chain for the ordering $\mathcal{R}$ on $\{x_1, \ldots, x_n\}$ and,*
2. *the saturated ideal $\mathrm{sat}(T, \mathcal{R})$ (which is an alias of $\mathrm{sat}(T)$ with a second argument recalling the ordering) of T of $\mathbf{k}[x_1, \ldots, x_n]$ is prime,*

compute $C \subset \mathbf{k}[x_1, \ldots, x_n]$ such that

3. *C is a regular chain for the ordering $\overline{\mathcal{R}}$ on $\{x_1, \ldots, x_n\}$ and,*
4. *the saturated ideal $\mathrm{sat}(C, \overline{\mathcal{R}})$ of C in $\mathbf{k}[x_1, \ldots, x_n]$ is equal to $\mathrm{sat}(T, \mathcal{R})$.*

We call this second problem *change of variable order*. The articles [18, 20] are actually dedicated to the case of differential regular chains, where a differential counterpart of Problem 2 is termed *ranking conversion*. However, these articles suggest that, from the differential case, a solution to Problem 2 could be derived and they call it `PALGIE`, which is an acronym for Prime ALGebraic IdEal. We present such an algebraic derivation in Section 6.3.1. Then, towards solving Problem 1, we consider the following.

Problem 3. *Given two total orderings $\mathcal{R}$ and $\overline{\mathcal{R}}$ on $\{x_1, \ldots, x_n\}$, given $T \subset \mathbf{k}[x_1, \ldots, x_n]$, assuming that T is a regular chain for the ordering $\mathcal{R}$ on $\{x_1, \ldots, x_n\}$, compute finitely many regular chains $C_1, \ldots, C_e$ such that the radical of the saturated ideal $\mathrm{sat}(T, \mathcal{R})$ of T in $\mathbf{k}[x_1, \ldots, x_n]$ is equal to the intersection of the radicals of the saturated ideals $\mathrm{sat}(C_i, \overline{\mathcal{R}})$ of C_i in $\mathbf{k}[x_1, \ldots, x_n]$, for $1 \leq i \leq e$.*

Extending the `PALGIE` algorithm (as suggested in [18]) to a solution of Problem 3 can be achieved by standard techniques from regular chain theory, see [25].

Finally, the `PALGIE` algorithm is further extended to a solution of Problem 1 in Section 6.3.3.

Before entering Section 6.3.1, we argue that Problem 2 deals with a special case of Problem 1, that is, ranking conversions are, indeed, a special case of linear change of coordinates.

As in the statement of Problem 2, consider two total orderings $\mathcal{R}$ and $\overline{\mathcal{R}}$ on $\{x_1, \ldots, x_n\}$ as well as a regular chain $T \subset \mathbf{k}[x_1, \ldots, x_n]$ for the order $\mathcal{R}$ such that its saturated ideal $\mathrm{sat}(T, \mathcal{R})$ is prime. W.l.o.g. we can assume that the order $\mathcal{R}$ on $\{x_1, \ldots, x_n\}$ is given by $x_1 < \cdots < x_n$. Then, the change of variable order from $\mathcal{R}$ to $\overline{\mathcal{R}}$ can be interpreted as a permutation σ of the sequence $(x_1, \ldots, x_n)$. Let A be the linear change of coordinates replacing the column vector $(x_1, \ldots, x_n)^t$ with $M_\sigma(y_1, ..., y_n)^t$ where $(y_1, \ldots, y_n)$ stand for the new coordinates and M_σ is the matrix of σ w.r.t. the canonical basis of $\overline{\mathbf{k}}^n$ as a vector space over $\overline{\mathbf{k}}$. Running the extended version of the

PALGIE algorithm solving Problem 1 and presented in Section 6.3.3, we obtain a regular chain C such that we have

$$\text{sat}(C) \;=\; \text{sat}(T)^A.$$

Then simply renaming y_i with $x_{\sigma(i)}$, for $1 \leq i \leq n$, in C produces a regular chain D satisfying the output specifications of the original version of the PALGIE algorithm (see Section 6.3.1) whose purpose is to perform change of variable order. To make the proof strict, requiring that T and D be strongly normalized (and reduced Gröbner bases over the field of rational functions $\mathbf{k}(\mathsf{free}(T))$) make them unique which completes the proof.

6.3.1 The PALGIE algorithm for the prime case

We consider two total orderings $\mathcal{R}$ and $\overline{\mathcal{R}}$ on $\mathbf{x}$. Let $F \subset \mathbf{k}[\mathbf{x}]$ be a finite set of polynomials, let $\overline{\mathbf{k}}$ be the algebraic closure of $\mathbf{k}$ and $V(F)$ the zero set of F in the affine space $\overline{\mathbf{k}}^n$. We assume that the ideal $\langle F \rangle$ generated by F in $\mathbf{k}[\mathbf{x}]$ is prime. We also assume that we are given another finite polynomial set $C \subset \mathbf{k}[\mathbf{x}]$ which is a regular chain for the total ordering $\mathcal{R}$ and whose saturated ideal is $\langle F \rangle$.

The goal of this section is to describe an algorithm, namely Algorithm 14, which given C, $\mathcal{R}$ and $\overline{\mathcal{R}}$, computes a polynomial set $\overline{C} \subset \mathbf{k}[\mathbf{x}]$ which is a regular chain for the total ordering $\overline{\mathcal{R}}$ and whose saturated ideal is $\langle F \rangle$.

Our main procedure is Algorithm 14. It relies on five other procedures for which pseudo-code is given through Algorithm 8, 9, 10, 11 and 13. It also relies on standard procedures for which no pseudo-code is given. For instance, in Algorithm 14, for $p \in \mathbf{k}[\mathbf{x}]$ and for a regular chain $\overline{C} \subset \mathbf{k}[\mathbf{x}]$ w.r.t. $\overline{\mathcal{R}}$, the function call $\text{red}(p, \overline{C}, \overline{\mathcal{R}})$ returns a polynomial $r \in \mathbf{k}[\mathbf{x}]$ such that $p - r \in \text{sat}(\overline{C}, \overline{\mathcal{R}})$

Proving the termination of Algorithm 8, 9, 10, 11, 13 and Algorithm 14 can be done following techniques that are standard in the literature dedicated to the theory of regular chains. Hence, we focus on the correctness, which is rather easy to prove and follows from Theorem 12.

Proposition 9. *Algorithm 8 satisfies its specifications.*

PROOF. The termination of Algorithm 8 follows from the fact that, at each recursive call, either the rank of the first argument or the rank of the second decreases. The correctness follows essentially from the following result: p is regular w.r.t. $\text{sat}(C, \mathcal{R})$ if and only if the iterated resultant of p w.r.t. C is not zero. The key feature of Algorithm 8 is the fact that at most one Boolean value b_i is true; this is easily verified by induction. □

Proposition 10. *Algorithm 9 satisfies its specifications.*

PROOF. This follows easily from the specifications of Algorithm 8. □

Proposition 11. *Algorithm 10 satisfies its specifications.*

PROOF. This follows easily from the specifications of Algorithm 8. □

Proposition 12. *Algorithm 11 satisfies its specifications.*

Algorithm 8: IsRegular($p, C, \mathcal{R}$)

Input: $C \subset \mathbf{k}[\mathbf{x}]$ a regular chain for the variable order $\mathcal{R}$ and a polynomial $p \in \mathbf{k}[\mathbf{x}]$.

Output: pairs $(b_1, C_1), \ldots, (b_e, C_e)$ where each b_i is a Boolean and each $C_i \subset \mathbf{k}[\mathbf{x}]$ a regular chain for $\mathcal{R}$ s.t. (1) the intersection of the $\sqrt{\text{sat}(C_i, \mathcal{R})}$ equals $\sqrt{\text{sat}(C, \mathcal{R})}$, (2) p is regular w.r.t. $\text{sat}(C_i, \mathcal{R})$ if and only if b_i is true, (3) p is zero modulo $\text{sat}(C_i, \mathcal{R})$ if and only if b_i is false, and (4) at most one b_i is true.

```
1  begin
2      if p = 0 then
3          return (false, C) ;
4      if p ∈ k then
5          return (true, C) ;
6      if C = ∅ then
7          return (true, C) ;
8      if (∀q ∈ C) deg(p, mvar(q)) = 0 then
9          return (true, C) ;
10     Let v be the largest variable in p which is algebraic in C ;
11     r := res(p, C_v, v) ;
12     if r = 0 then
13         g := gcd(p, C_v) ;
14         C_1 := C_v^- ∪ {g} ∪ C̄_v^+ ;
15         q := C_v/g ;
16         C_2 := C_v^- ∪ {q} ∪ C̄_v^+ ;
17         output (false, C_1) ;
18         return IsRegular(p, C_2, R) ;
19     if r ∈ k then
20         return (true, C) ;
21     return IsRegular(r, C, R) ;
22 end
```

Algorithm 9: Saturate($C, H, \mathcal{R}$)

Input: $C \subset \mathbf{k}[\mathbf{x}]$ a regular chain for the variable order $\mathcal{R}$ and a polynomial set $H \subset \mathbf{k}[\mathbf{x}]$.
Output: $D \subset \mathbf{k}[\mathbf{x}]$ a regular chain for the variable order $\mathcal{R}$ s.t. $\text{sat}(C, \mathcal{R}) : H^\infty = \text{sat}(D, \mathcal{R})$ and each polynomial $h \in H$ is regular w.r.t. $\text{sat}(D)$, thus we have $\text{sat}(D, \mathcal{R}) : H^\infty = \text{sat}(D, \mathcal{R})$.

```
1 begin
2     for h ∈ H do
3         for (bool, D) ∈ IsRegular(h, C, R) do
4             if bool then
5                 C := D;
6                 break;
7     return C;
8 end
```

Algorithm 10: Extend($C, D, \mathcal{R}$)

Input: $C, D \subset \mathbf{k}[\mathbf{x}]$ two regular chains for the variable order $\mathcal{R}$ such that for every $f \in C$ and every $g \in D$, we have $\text{mvar}(f) < \text{mvar}(g)$.
Output: $E \subset \mathbf{k}[\mathbf{x}]$ a regular chain for the variable order $\mathcal{R}$ s.t. $\text{sat}(C \cup D, \mathcal{R}) : H^\infty = \text{sat}(E, \mathcal{R})$ where $H = \{\text{init}(p, \mathcal{R}) \mid p \in D\}$.

```
1  begin
2      E := C;
3      repeat
4          Let p ∈ D with minimum rank in R ;
5          D := D \ {p} ;
6          for (bool, F) ∈ IsRegular(init(p, R), E, R) do
7              if bool then
8                  p := PrimitivePart(red(p, F, R), mvar(p, R));
9                  E := F ∪ {p};
10                 break;
11     until D = ∅ ;
12     return E;
13 end
```

Algorithm 11: EnsureRank($p, \overline{\mathcal{R}}, C, \mathcal{R}$)

Input: $\mathcal{R}$ and $\overline{\mathcal{R}}$ total orders on $\mathbf{x}$; $C \subset \mathbf{k}[\mathbf{x}]$ a regular chain for $\mathcal{R}$ and a polynomial $p \in \mathbf{k}[\mathbf{x}]$ such that $p \in \mathcal{I}$.

Output: a polynomial $r \in \mathbf{k}[\mathbf{x}]$ and two polynomial sets $P', H' \subset \mathbf{k}[\mathbf{x}]$ such that $p - r \in \langle P' \rangle$, $P' \subset \mathcal{I}$ and $H' = \{\text{init}(r, \overline{\mathcal{R}})\}$, if $r \notin \mathbf{k}$, otherwise $H' = \emptyset$.

```
1 begin
2     P' := ∅ ;
3     while p ≠ 0 and init(p, R̄) ∈ sat(C, R) do
4         P' := P' ∪ {init(p, R̄)} ;
5         p := tail(p, R̄) ;
6     if p ≠ 0 then
7         H' := {init(p, R̄)} ;
8     return(p, P', H') ;
9 end
```

Algorithm 12: EnsureLeadingCoefficient($p, v, \overline{\mathcal{R}}, C, \mathcal{R}$)

Input: $\mathcal{R}$ and $\overline{\mathcal{R}}$ total orders on $\mathbf{x}$; $C \subset \mathbf{k}[\mathbf{x}]$ a regular chain for $\mathcal{R}$ and a polynomial $p \in \mathbf{k}[\mathbf{x}]$ such that $p \in \mathcal{I}$ and $v \in \mathbf{x}$ a variable.

Output: a polynomial $r \in \mathbf{k}[\mathbf{x}]$ and two polynomial sets $P', H' \subset \mathbf{k}[\mathbf{x}]$ such that $p - r \in \langle P' \rangle$, $P' \subset \mathcal{I}$ and $H' = \{\text{lc}(r, v)\}$, if $r \notin \mathbf{k}$, otherwise $H' = \emptyset$.

```
1 begin
2     P' := ∅ ;
3     while p ≠ 0 and lc(p, v) ∈ sat(C, R) do
4         P' := P' ∪ {lc(p, v)} ;
5         p := reductum(p, v) ;
6     if p ≠ 0 then
7         H' := {lc(p, v)} ;
8     return(p, P', H') ;
9 end
```

Algorithm 13: $\mathsf{Gcd}_n(q, p, v, \overline{C}_v^-, \overline{\mathcal{R}}, C, \mathcal{R})$

Input: $\mathcal{R}$ and $\overline{\mathcal{R}}$ total orders on $\mathbf{x}$; $v \in \mathbf{x}$ a variable; $q, p \in \mathbf{k}[\mathbf{x}]$ two non-constant polynomials with common main variable v w.r.t. $\overline{\mathcal{R}}$ s.t. $\deg(q, v) > \deg(p, v)$ holds; $C \subset \mathbf{k}[\mathbf{x}]$ a regular chain for $\mathcal{R}$ s.t. $\mathcal{I} := \mathrm{sat}(C, \mathcal{R})$ is prime; $\overline{C}_v^- \subset \mathbf{k}[\mathbf{x}]$ a regular chain for $\overline{\mathcal{R}}$ where all variables are less than v w.r.t. $\overline{\mathcal{R}}$ and both $\mathrm{init}(p, \overline{\mathcal{R}})$, $\mathrm{init}(q, \overline{\mathcal{R}})$ are regular w.r.t. $\mathrm{sat}(\overline{C}_v^-, \overline{\mathcal{R}})$.

Output: a polynomial $g \in \mathbf{k}[\mathbf{x}]$ s.t. g is a GCD of p, q in $\mathbf{L}[v]$ where $\mathbf{L}$ is the total ring of fractions of $\mathbf{k}[x_i \in \mathbf{x} \mid x_i <_{\overline{R}} v]/\mathrm{sat}(\overline{C}_v^-, \overline{\mathcal{R}})$ s.t. if both q, p belong to $\mathcal{I}$ then $g \in \mathcal{I}$ and $\deg(g, v) > 0$ both hold; two polynomial sets $P, H \subset \mathbf{k}[\mathbf{x}]$ s.t. (1) g belongs to the ideal generated by q, p and P, (2) we have $P \subset \mathcal{I}$ and (3) H is the set of the initials of the intermediate pseudo-remainders from q, p to g.

```
1  begin
2      (p1, p2) := (q, p) ;
3      (P, H) := (∅, ∅) ;
4      while true do
5          δ := deg(p1, v) − deg(p2, v) ;
6          ψ := −1 ;
7          β := (−1)^(δ+1) ;
8          α := lc(p2, v) ;
9          while deg(p2, v) > 0 do
10             p3 := exquo(prem(p1, p2, v), β) ;
11             d3 := deg(p3, v) ;
12             (p3, P′, H′) := EnsureLeadingCoefficient(p3, v, R̄, C, R) ;
13             (P, H) := (P ∪ P′, H ∪ H′) ;
14             p3 := red(p3, C̄v^−, R̄) ;
15             if p3 = 0 then
16                 return(p2, P, H)
17             if deg(p3, v) < d3 then
18                 (p1, p2) := (p2, p3) ;
19                 break ;
20             if δ > 0 then
21                 ψ := exquo((−α)^δ, ψ^(δ−1)) ;
22             (p1, p2) := (p2, p3) ;
23             if deg(p2, v) > 0 then
24                 δ := deg(p1, v) − deg(p2, v) ;
25                 β := −αψ^δ ;
26                 α := lc(p2, v) ;
27         if p2 = 0 then
28             return(p1, P, H)
29         else
30             return(p2, P, H)
31 end
```

Algorithm 14: Palgie($C, \mathcal{R}, \overline{\mathcal{R}}$)

Input: $\mathcal{R}$ and $\overline{\mathcal{R}}$ total orders on $\mathbf{x}$; $C \subset \mathbf{k}[\mathbf{x}]$ a regular chain for $\mathcal{R}$ s.t. $\mathcal{I} := \mathrm{sat}(C, \mathcal{R})$ is prime.

Output: $\overline{C} \subset \mathbf{k}[\mathbf{x}]$ a regular chain for $\overline{\mathcal{R}}$ s.t. $\mathrm{sat}(C, \mathcal{R}) = \mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$.

begin
  $P := C$;
  $H := \{\mathrm{init}(p, \mathcal{R})$ **for** $p \in C\}$;
  $\overline{C} := \emptyset$;
  repeat
    Let $p \in P$ with minimum rank in $\overline{\mathcal{R}}$;
    $P := P \setminus \{p\}$;
    $p := \mathrm{red}(p, \overline{C}, \overline{\mathcal{R}})$;
    $(p, P', H') := \mathsf{EnsureRank}(p, \overline{\mathcal{R}}, C, \mathcal{R})$;
    $(P, H) := (P \cup P', H \cup H')$;
    if $p \neq 0$ **then**
      $v := \mathrm{mvar}(p)$;
      $\overline{C} := \mathsf{Saturate}(\overline{C}, \{\mathrm{init}(p, \overline{\mathcal{R}})\}, \overline{\mathcal{R}})$;
      if $(\forall q \in \overline{C})\ \mathrm{mvar}(q) \neq v$ **then**
        $\overline{C} := \mathsf{Extend}(\overline{C}_v^- \cup \{p\}, \overline{C}_v^+, \overline{\mathcal{R}})$;
      else
        $(g, P', H') := \mathsf{Gcd}_n(\overline{C}_v, p, \overline{C}_v^-, \overline{\mathcal{R}}, C, \mathcal{R})$;
        $(P, H) := (P \cup P', H \cup H')$;
        $\overline{C} := \mathsf{Saturate}(\overline{C}, \{\mathrm{init}(g, \overline{\mathcal{R}})\}, \overline{\mathcal{R}})$;
        $\overline{C} := \overline{C} \setminus \{\overline{C}_v\} \cup \{g\}$;
      $\overline{C} := \mathsf{Saturate}(\overline{C}, H, \overline{\mathcal{R}})$;
  until $P = \emptyset$;
  return$(\overline{C})$;
end

PROOF. This is clear. □

Proposition 13. *Algorithm 13 satisfies its specifications.*

PROOF. Let us first ignore Line **(14)** in the first place and assume that at each execution of Line **(17)** we have $d_3 = \deg(p_3, v)$. Under these assumptions, Algorithm 13 computes a GCD of q, p in the ring $\mathbf{L}[v]$. Note that $\mathbf{L}$ may not be a field, not even a direct product of fields and this GCD is in the sense of [67].

Now let us assume that at some iteration of the inner **while**-loop the condition $d_3 = \deg(p_3, v)$ becomes false at Line **(17)**. Then, the inner **while**-loop breaks out and the pair (q, p) is replaced by the current value of (p_2, p_3) at the beginning of the outer **while**-loop. It is not difficult to see that if Algorithm 13 satisfies its specifications with (p_2, p_3) as input then it does with (q, p) as well.

Now let us take Line **(14)** into account. The question is whether the exact divisions performed at Lines **(10)** and **(21)** in the polynomial ring $\mathbf{k}[x_i \in \mathbf{x} \mid x_i <_{\overline{R}} v][v]$ are indeed exact when p_3 is replaced by $\mathrm{red}(p_3, \overline{C}_v^-, \overline{\mathcal{R}})$. The answer is *yes*, thanks to the main theorem of [67] which allows to replace p_3 as computed at Line **(10)** by $\lambda_3 p_3$ where λ_3 is any regular element in the ring $\mathbf{L}$.

The last (non-trivial) point that remains to be explained is the following claim: if both q, p belong to $\mathcal{I}$ then $\deg(g, v) > 0$ holds. Hence we assume that $q, p \in \mathcal{I}$ holds. Observe that, by construction, the polynomial g belongs to $\mathcal{I}$ as well. Assume, by contradiction, that $\deg(g, v) = 0$ holds. At Line **(12)**, the input polynomial p_3 of EnsureRank belongs to $\mathcal{I}$; hence the specifications of EnsureRank imply that the output polynomial p'_3 is null if $\deg(p'_3, v) = 0$ holds. Hence, we can assume that, at Line **(13)**, $\deg(p_3, v) > 0$ holds. Now, if the output polynomial p'_3 at Line **(14)** satisfies $\deg(p'_3, v) = 0$, we would have $\mathrm{init}(p'_3, \overline{\mathcal{R}}) \in \mathrm{sat}(\overline{C}_v^-, \overline{\mathcal{R}})$. However, by construction, the polynomials of $\overline{C}$ belong to $\mathcal{I}$ while each of their initials does not (see Proposition 16). Hence, we have $\mathrm{sat}(\overline{C}_v^-, \overline{\mathcal{R}}) \subseteq \mathcal{I}$. Therefore, we would have $\mathrm{init}(p'_3, \overline{\mathcal{R}}) \in \mathcal{I}$, contradicting with the specifications of EnsureRank. □

Proposition 14. *The following invariant properties hold at the beginning and at the end of each iteration of the* **while***-loop of the execution of Algorithm 14.*

(*i*) $P \subset \mathcal{I}$,

(*ii*) $\overline{C}$ *is a regular chain for* $\overline{\mathcal{R}}$,

(*iii*) *every polynomial* $h \in H$ *is regular w.r.t.* $\mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$.

PROOF. We first prove (*i*). At Line **(2)**, the set P is initialized with C which satisfies $C \subset C : H^\infty = \mathcal{I}$. At Lines **(10)** and **(18)**, the set P is incremented directly or indirectly by the procedure EnsureRank which adds to P polynomials belonging to $\mathrm{sat}(C, \mathcal{R}) = \mathcal{I}$. Thus, Claim (*i*) is proved.

Next, we prove (*ii*) proceeding by induction. At the beginning of the first iteration of the **while**-loop, we have $\overline{C} = \emptyset$; hence $\overline{C}$ is a regular chain. Assume that, at the beginning of each subsequent iteration of the **while**-loop, $\overline{C}$ is a regular chain for $\overline{\mathcal{R}}$. If at Line **(11)**, the polynomial p is null, then the conclusion is clear. From now on, we assume $p \neq 0$. Observe that p cannot be a non-zero constant polynomial; indeed we have $p \in \mathcal{I}$ from (*i*) and $\mathcal{I}$ is not the entire

polynomial ring $\mathbf{k}[\mathbf{x}]$. Hence, the main variable v of p is well-defined. After executing Line **(13)**, the initial of p w.r.t. $\overline{R}$ is regular w.r.t. $\mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$, thus $\overline{C_v} \cup \{p\}$ is a regular chain for $\overline{R}$. If v is not algebraic in $\overline{C}$, then the specifications of Extend imply that $\overline{C}$ is still a regular chain after executing **(15)**. If v is algebraic in $\overline{C}$, then the specifications of RegularGcd imply that $\overline{C}$ is still a regular chain after executing **(20)**. For understanding this latter claim, one should note that g, as computed at Line **(17)**, has a positive degree in v but its initial in $\overline{\mathcal{R}}$ may not be regular w.r.t. $\overline{C_v}$. Indeed, Algorithm 13 does not perform any regularity test. For this reason, Line **(19)** is essential. Therefore, Claim (*ii*) is proved.

Finally, Claim (*iii*) follows easily from the specifications of Saturate. One should note that the instruction at Line **(21)** could be moved just after Line **(22)**. Then, Algorithm 14 would still satisfy its specifications. However, enforcing the fact that every polynomial $h \in H$ is regular w.r.t. $\mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$ at the beginning of the each iteration of the **while**-loop has the following benefit in terms of performance: it keeps the main degrees in $\overline{C}$ as small as possible. Since this implies to perform lots of regularity tests, the optimization described in Section 6.3.2 is crucial. □

Lemma 27. *Considering the i-th iteration of the* **while***-loop of the execution of Algorithm 14, we denote by p_i the polynomial p which is selected from P at Line* **(6)** *and by $\overline{C_i}$ the value of the regular chain $\overline{C}$ at the end of that iteration. Let ℓ be the index of the last iteration the* **while***-loop of the execution of Algorithm 14. Then, the following properties hold:*

(*i*) *for all $1 \leq i < \ell$, we have* $\mathrm{sat}(\overline{C_i}, \overline{\mathcal{R}}) \subseteq \mathrm{sat}(\overline{C_{i+1}}, \overline{\mathcal{R}})$,

(*ii*) *for all $1 \leq i \leq \ell$, we have* $p_i \in \mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$.

PROOF. We first prove (*i*). It clearly follows from the specifications of Saturate that at each of the Lines **(13), (19), (21)** the input and output values of $\overline{C}$, say $\overline{C_{\mathrm{in}}}$ and $\overline{C_{\mathrm{out}}}$, satisfy $\mathrm{sat}(\overline{C_{\mathrm{in}}}, \overline{\mathcal{R}}) \subseteq \mathrm{sat}(\overline{C_{\mathrm{out}}}, \overline{\mathcal{R}})$. Similarly, if we denote by $\overline{C_{\mathrm{in}}}$ and $\overline{C_{\mathrm{out}}}$, the regular chains computed at Line **(13)** and **(15)** respectively, we have again $\mathrm{sat}(\overline{C_{\mathrm{in}}}, \overline{\mathcal{R}}) \subseteq \mathrm{sat}(\overline{C_{\mathrm{out}}}, \overline{\mathcal{R}})$. This proves Claim (*i*). Next, we prove (*ii*). The proof is by induction on $j := \ell - i$, thus for $0 \leq j \leq \ell$. So, we first consider $j = 0$, that is, the last iteration. During this iteration, the set P is not incremented. Then, three cases arise:

- either p_ℓ is proved to be zero modulo $\mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$ at Line **(8)**,
- or p_ℓ is added to $\overline{C}$ at Line **(15)**,
- or $\overline{C}_v$ is replaced by $g = \mathsf{Gcd}_n(\overline{C}_v, p_\ell, \overline{C_v}, \overline{\mathcal{R}}, C, \mathcal{R})$ at Line **(20)**.

In each of these three cases, it is clear that we have $p_\ell \in \mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$. Now let $0 \leq j < \ell$ and assume that all polynomials p selected at iterations $\ell - (j+1), \ldots, \ell$ will be proved to belong to $\mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$. We shall prove that p_i, with $i = \ell - j$, also satisfies $p_i \in \mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$. Observe that after Line**(9)**:

- either p_i was proved to be zero modulo $\mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$ at Line **(8)**,
- or p_i was replaced by a polynomial $(p')_i$ and polynomials added to P at Line **(9)**.

Our induction hypothesis implies that those polynomials added to P will be proved later on to belong to $\mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$. Hence, it is enough to prove that $(p')_i \in \mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$. This can be done easily using the same type of arguments that have been using so far in this proof. Hence, we can conclude that Claim (*ii*) holds. □

Proposition 15. *At the end of last iteration of the* **while***-loop of the execution of Algorithm 14, the properties hold:*

(*i*) $C \subset \mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$,

(*ii*) $\mathrm{sat}(C, \mathcal{R}) \subseteq \mathrm{sat}(\overline{C}, \overline{\mathcal{R}})$.

PROOF. Claim (*i*) follows from Property (*ii*) of Lemma 27 and the fact that the set P is initialized with C at Line **(2)** in Algorithm 14. Claim (*ii*) from Claim (*i*), the fact that H is initialized with the set of the initials of C at Line **(3)** in Algorithm 14, as well as Property (*iii*) of Proposition 14. □

Proposition 16. *At the end of each iteration of the* **while***-loop of the execution of Algorithm 14, the following properties hold:*

(*i*) *every initial of a polynomial* $p \in \overline{C}$ *is regular w.r.t.* $\mathrm{sat}(C, \mathcal{R})$,

(*ii*) $\overline{C} \subset \mathrm{sat}(C, \mathcal{R})$,

(*iiii*) $\mathrm{sat}(\overline{C}, \overline{\mathcal{R}}) \subseteq \mathrm{sat}(C, \mathcal{R})$.

PROOF. We prove (*i*) and (*ii*). Both properties trivially hold at the beginning of the first iteration of the **while**-loop of Algorithm 14. In the course of the execution of Algorithm 14, the regular chain $\overline{C}$ is modified in two ways:

- either by adding to $\overline{C}$ a polynomial which belongs to $\mathcal{I}$ and whose initial does not belong to $\mathcal{I}$, at Lines **(15)** and **(19-20)**,
- or by *saturation* at Lines **(13)**, **(15)** and **(21)**.

The second way preserves Properties (*i*) and (*ii*) since those saturations are performed w.r.t. elements which do not belong to $\mathcal{I}$ and $\mathcal{I}$ is prime. To be more precise, when a polynomial is not regular modulo $\overline{C}$, Saturate function should preserves two properties

1. choosing a branch where the polynomials still belong to $\mathcal{I}$ where $\mathcal{I}$ is prime,
2. choosing a branch where polynomials still form a regular chain.

As one can see, the second property has been taken care of in Algorithm 9. But when we want to add a new polynomial p to $\overline{C}$ which is not regular and therefore we need to choose the right branch of $\overline{C}$, we do not check either in Saturate or IsRegular, if that branch still belongs to $\mathcal{I}$ or not. In fact, when we pass p to IsRegular, we discard the branch C_1 at line **(14)** without checking whether it belongs to $\mathcal{I}$ or not. But actually it is not really necessary thank to the prime property of $\mathcal{I}$. Since polynomial p is regular modulo $\mathcal{I}$, thus polynomial g in line **(13)**

is also regular and subsequently does not belong to $\mathcal{I}$; which implies that Algorithm 8 chooses the right branch of $\overline{C}$ automatically.

The fact that the first way preserves Properties (*i*) and (*ii*) follows from Theorem 6.1 in [11] and the fact that the ideal $\mathcal{I}$ is prime. It is worth noting that in Line **(15)**, when we add polynomial p to $\overline{C}$, property (*ii*) might not be valid since some factors of p might not belong to $\mathcal{I}$, but then by applying the line **(21)** , we discard those unnecessary factors, and so property (*ii*) still remains valid at the end of the **while**-loop. Finally Claim (*iii*) follows immediately from Claims (*i*) and (*ii*). □

Theorem 12. *Algorithm 14 satisfies its specifications.*

PROOF. From Proposition 15 we have $\text{sat}(C, \mathcal{R}) \subseteq \text{sat}(\overline{C}, \overline{\mathcal{R}})$, while from Proposition 16 we have $\text{sat}(\overline{C}, \overline{\mathcal{R}}) \subseteq \text{sat}(C, \mathcal{R})$. Moreover, from Proposition 14, the triangular set $\overline{C}$ is a regular chain, which completes the proof. □

6.3.2 Regularity test in IsRegular($p, \overline{C}, \overline{\mathcal{R}}$)

The motivation is to prevent from checking that p is regular w.r.t. $\text{sat}(\overline{C}, \overline{\mathcal{R}})$ whenever this could be deduced from previous computations. To this end, let us assume that IsRegular($p, \overline{C}, \overline{\mathcal{R}}$) is called by a wrapper-function IsRegular($p, \overline{C}, \overline{\mathcal{R}}, \mathsf{T}$) where the fourth argument T is a table whose keys are polynomials and values are bit vectors. The wrapper-function IsRegular($p, \overline{C}, \overline{\mathcal{R}}, \mathsf{T}$) works as follows:

1. if p is not a key of T, then
   (a) IsRegular($p, \overline{C}, \overline{\mathcal{R}}$) is run; let (true, $\overline{D}$) be the returned pair whose first item is true.
   (b) let $V(p)$ be the vector of size n defined as follows: the i-th bit of $V(p)$ is 1 if (and only if) the i-th variable of $\mathbf{x}$ w.r.t. $\overline{\mathcal{R}}$ is algebraic in $\overline{D}$.
   (c) we add $(p, V(p))$ to T and return what IsRegular($p, \overline{C}, \overline{\mathcal{R}}$) computed.
2. if p is a key of T, then
   (a) we compute the current value of the vector $V(p)$ and compare it with the one stored in T, that we denote by $V_0(p)$;
   (b) let v be the main variable of p in $\overline{\mathcal{R}}$ and ℓ be its rank in $\overline{\mathcal{R}}$;
   (c) if the first ℓ bits of $V(p)$ and $V_0(p)$ agree, then no computation is needed and (true, $\overline{C}$) is returned;
   (d) otherwise, IsRegular($p, \overline{C}, \overline{\mathcal{R}}$) is run; let (true, $\overline{D}$) be the returned pair item whose first item is true; we update $V(p)$ into T using $\overline{D}$ and we return what IsRegular($p, \overline{C}, \overline{\mathcal{R}}$) computed.

6.3.3 The PALGIE algorithm for linear change of coordinates

We turn our attention back to Problem 1 and suggest how a solution of Problem 3 can lead to a solution of Problem 1. Let $T \subset \mathbf{k}[\mathbf{x}]$ be a regular chain and let A be a linear change of coordinates in $\overline{\mathbf{k}}^n$. We denote by d the dimension of $\mathrm{sat}(T)$. W.l.o.g. we assume that the variables $x_1 < \cdots < x_d$ are algebraically independent modulo $\mathrm{sat}(T)$, that is, $\mathsf{free}(T) = \{x_1, \ldots, x_d\}$. Let us write $T = \{t_{d+1}, \ldots, t_n\}$ such that t_i has main variable x_i and initial h_i. We apply the extended version of the PALGIE algorithm (that is, the one solving Problem 3) to the solving of the polynomial system S below

$$\left\{ \begin{array}{rcl} t_n^A(\mathbf{x}) & = & 0 \\ \vdots & \vdots & \vdots \\ t_{d+1}^A(\mathbf{x}) & = & 0 \\ h_{d+1}^A(\mathbf{x}) \cdots h_n^A(\mathbf{x}) & \neq & 0 \end{array} \right. \tag{6.5}$$

We denote by $Z(S) \subset \overline{\mathbf{k}}^n$ the zero set of S. Observe that for all polynomials $f \in \mathbf{k}[\mathbf{x}]$, we have

$$f \in \langle Z(S) \rangle \iff f^{A^{-1}} \in \sqrt{\mathrm{sat}(T)}. \tag{6.6}$$

where $\langle Z(S) \rangle$ is the ideal of $\mathbf{k}[\mathbf{x}]$ consisting of all polynomials vanishing on $Z(S)$. Relation (6.6) allows one to easily adapt the *master - student relationship* described in Section 3.2 of [20] and thus to adapt the (extended version of the) PALGIE algorithm so as to solve Problem 1.

6.4 Noether normalization and regular chains

In this section, we study the relation between Noether normalization and regular chains. Our initial quest was to determine whether, for a prime ideal $\mathcal{P} \subset \mathbf{k}[\mathbf{x}]$ in Noether position, one could find a monic regular chain T whose saturated ideal is precisely $\mathcal{P}$. For this purpose, we start by reviewing basic properties of Noether normalization, following Logar's paper [60].

Let $\mathcal{P} \subset \mathbf{k}[\mathbf{x}]$ be a (proper) prime ideal and G the reduced lexicographical Gröbner basis of $\mathcal{P}$. Recall that $\mathbf{x}$ counts n variables ordered as $x_1 < \cdots < x_n$. We assume that $\mathbf{k}$ is an infinite field. We denote by $T_{\mathcal{P}}$ the set defined by

$$T_{\mathcal{P}} \; = \; \{v \in \mathbf{x} \mid (\forall g \in G)\, \mathrm{mvar}(g) \neq v\}. \tag{6.7}$$

This set satisfies two important properties:

- $T_{\mathcal{P}}$ is algebraically independent modulo $\mathcal{P}$ that is, $\mathcal{P} \cap \mathbf{k}[T_{\mathcal{P}}] = \langle 0 \rangle$,
- the number of elements in $T_{\mathcal{P}}$ gives the dimension of $\mathcal{P}$, that is, $\dim(\mathcal{P}) = \mathrm{card}(T_{\mathcal{P}})$.

A variable $x_s \in \mathbf{x}$ is said *integral* over $\mathbf{k}[x_1, \ldots, x_{s-1}]$ modulo $\mathcal{P}$ if there exists $f \in \mathcal{P} \cap \mathbf{k}[x_1, \ldots, x_{s-1}, x_s]$ such that $\mathrm{mvar}(f) = x_s$ and $\mathrm{init}(f) \in \mathbf{k}$. Integral variables satisfy two important properties:

- A variable $x_s \in \mathbf{x}$ is integral over $\mathbf{k}[x_1, \ldots, x_{s-1}]$ modulo $\mathcal{P}$ if and only if there exists $g \in G$ such that $\mathrm{lm}(g) = x_s^{d_s}$ for some positive integer d_s,
- if a variable $x_s \in \mathbf{x}$ is integral over $\mathbf{k}[x_1, \ldots, x_{s-1}, \mathbf{u}]$ modulo $\mathcal{P}$, with $\mathbf{u} \subseteq T_{\mathcal{P}}$ disjoint from $\{x_1, \ldots, x_s\}$, then x_s is also integral over $\mathbf{k}[x_1, \ldots, x_{s-1}]$ modulo $\mathcal{P}$.

Thanks to the above properties, we may assume w.l.o.g. that if $d = \dim(\mathcal{P})$ then we have $T_{\mathcal{P}} = \{x_1, \ldots, x_d\}$. Consider a linear change of coordinates A in $\overline{\mathbf{k}}^n$ defined by a matrix M of the following form:

$$M = \left(\begin{array}{c|cccc} & a_{1,d+1} & \ldots & a_{1,n} \\ \mathrm{I}_{d\times d} & \vdots & \vdots & \vdots \\ & a_{d,d+1} & \ldots & a_{d,n} \\ \hline \mathbf{0} & \multicolumn{3}{c}{\mathrm{I}_{(n-d)\times(n-d)}} \end{array}\right) \tag{6.8}$$

where $a_{i,j} \in \mathbf{k}$. We denote by $\mathcal{P}^A$ the ideal generated by f^A for all $f \in \mathcal{P}$. Then, by Noether normalization lemma, for a generic choice of $a_{1,d+1}, \ldots, a_{d,n}$ the following properties hold:

1. $x_1, \ldots, x_d$ are algebraically independent modulo $\mathcal{P}^A$,
2. x_{d+i} is integral over $\mathbf{k}[x_1, \ldots, x_d]$ modulo $\mathcal{P}^A$ for all $i = 1, \ldots, n-d$.

In this case, we say that $\mathcal{P}^A$ is in *Noether position*.

We turn our attention to the regular chain representation of the prime ideal $\mathcal{P}$. To this end, using Theorem 3.3 of [11], one can extract, in an algorithmic fashion, a subset T of G such that T is a regular chain whose saturated ideal is precisely $\mathcal{P}$. Let H be the reduced lexicographical Gröbner basis of $\mathcal{P}^A$ and C be the regular chain extracted from H using the same theorem from [11].

Theorem 13. *If T generates its saturated ideal, then the regular chain C is monic, that is, for each polynomial $f \in C$ we have* $\mathrm{init}(f) \in \mathbf{k}$.

Proof. Assume by contradiction that there exists $f \in C$ such that $\mathrm{init}(f) \notin \mathbf{k}$ and let us choose such an f with minimum main variable. Since $x_1, \ldots, x_d$ are algebraically independent modulo $\mathcal{P}^A$ and since C is a regular chain, one can compute a polynomial f' such that $\mathrm{init}(f') \in \mathbf{k}[x_1, \ldots, x_d]$ and $\mathrm{sat}(C') = \mathrm{sat}(C)$ holds with $C' = C \setminus \{f\} \cup \{f'\}$.

Let $\mathrm{mvar}(f) = x_r$. Since $\mathcal{P}^A$ is in Noether position, it follows from [60] that there exists a polynomial $H_{x_r} \in H$ whose leading monomial is of the form $x_r^{d_r}$. Since $\mathrm{init}(H_{x_r}) \in \mathbf{k}$, we have

$$\deg(f', x_r) = \deg(f, x_r) < d_r = \deg(H_{x_r}, x_r). \tag{6.9}$$

Indeed, otherwise the polynomial H_{x_r} would have been selected as an element of the regular chain C.

From the choice of f and the assumption on T, the regular chain $C' \cap \mathbf{k}[x_1, \ldots, x_r]$ is a basis of $\mathcal{P}^A \cap \mathbf{k}[x_1, \ldots, x_r]$. Therefore, the polynomial H_{x_r} reduces to zero through multivariate division by $C' \cap \mathbf{k}[x_1, \ldots, x_r]$ and thus by $C \cap \mathbf{k}[x_1, \ldots, x_r]$. This contradicts the fact that H is a reduced Gröbner basis. □

Remark 12. *Theorem 13 states that if T generates* $\mathrm{sat}(T)$ *and $\mathcal{P}^A$ is in Noether position, then C is monic. Unfortunately, if T does not generate* $\mathrm{sat}(T)$*, then the previous conclusion may not hold as shown by the following example.*

Example 11. *Consider the regular chain $T := \{x_2^5 - x_1^4, x_1x_3 - x_2^2\} \subset \mathbb{Q}[x_1 < x_2 < x_3]$ which does not generate its saturated ideal. Consider also the linear change of coordinates A defined by the matrix below*

$$M = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Then $\langle T \rangle^A$ *is in Noether position and under this new change of coordinates we can compute the regular chain* $C = \{c_1, c_2\}$ *such that* $\sqrt{\text{sat}(C)} = \sqrt{\text{sat}(T)^A}$ *where* $c_1 = x_2^5 - 2x_2^4 + x_2^3 + 4x_1^2x_2^2 - x_1^4$ *and* $c_2 = \left(-x_1^3 + 2x_2^2x_1\right)x_3 + x_1^2x_2^2 - x_2^4 + x_2^3$. *As you can see* $\text{init}(c_2) \notin \mathbb{Q}$.

6.5 Applications of random linear changes of coordinates

Let $T \subset \mathbf{k}[\mathbf{x}]$ be a regular chain whose saturated ideal has dimension d. Let $\mathbf{u}$ be the free variables of T. Recall that h_T stands for the product of the $\text{init}(f)$ for $f \in T$. Let A be a linear change of coordinates in $\overline{\mathbf{k}}^n$. Assume that the extended version of the PALGIE algorithm (see Problem 3 in Section 6.3) applied to T and A produces a single regular chain $C \subset \mathbf{k}[\mathbf{x}]$, thus satisfying

$$\overline{W^A(T)} = \overline{W(C)}. \tag{6.10}$$

Let h_T and h_C be the products of the initials of T and C, respectively. Let r_T^A and r_C be the iterated resultants (see [25] for this term) of h_T^A and h_C w.r.t. C.

Proposition 17 gathers elementary properties of r_T^A and r_C. Proposition 18 provides conditions for deriving a basis of $\text{sat}(T)$ from the calculation of C while Theorem 14 provides a condition for deriving $\lim(W(T)) = \overline{W(T)} \cap V(h_T)$ from the calculation of C. The basic idea of Theorem 14 is to use a linear change of coordinates so as to replace the description of $\overline{W(T)}$ by one for which $\overline{W(T)} \cap V(h_T)$ can be computed by set-theoretic operations on constructible sets (represented by regular chain as in [24]). Moreover, Corollary 2 shows that, if T generates $\text{sat}(T)$, then the computation of $\lim(W(T))$ can always be achieved by the techniques of [24].

Proposition 17. *The following properties hold:*

(*i*) *the polynomial* h_T^A *is regular w.r.t.* $\text{sat}(C)$,

(*ii*) *the polynomials* r_T^A *and* r_C *belong to* $\mathbf{k}[\mathbf{u}]$ *and are non-zero.*

PROOF. Property (*i*) is by construction, that is, following the extended PALGIE algorithm applied to T and A. Property (*ii*) follows from (*i*) and the relations between regular chains and iterated resultants, see [25]. □

Proposition 18. *The following properties hold:*

(*i*) *if* $\text{sat}(T)$ *is radical and if the ideal* $\langle h_T, (h_C^{A^{-1}})\rangle$ *equals the whole ring* $\mathbf{k}[\mathbf{x}]$, *then* $T \cup C^{A^{-1}}$ *generates* $\text{sat}(T)$,

(*ii*) *if the regular chain* C *is monic, then* $C^{A^{-1}}$ *generates* $\text{sat}(T)$.

PROOF. We prove Property (*i*). Since $\text{sat}(T)$ is radical, the relations $\overline{W^A(T)} = \overline{W(C)}$ implies $C^{A^{-1}} \subset \text{sat}(T)$. Hence, we "only" need to prove that if a polynomial f belongs to $\text{sat}(T)$, then f is generated by $T \cup C^{A^{-1}}$. So let $f \in \text{sat}(T)$. On one hand, there exists a non-negative integer e such that $h_T^e f \in \langle T \rangle$. On the other, there exists a non-negative integer d such that $(h_C^{A^{-1}})^d f \in \langle C^{A^{-1}} \rangle$. Since the ideal $\langle h_T^e, (h_C^{A-1})^d \rangle$ is the whole ring $\mathbf{k}[\mathbf{x}]$, then we can write f as an element of $\langle T, C^{A^{-1}} \rangle$. Now we prove (*ii*). Since C is monic, it is a Gröbner basis of

$\mathrm{sat}(C)$, and, from the specifications of the PALGIE algorithm, a basis of $\mathrm{sat}(T)^A$ as well. Thus $C^{A^{-1}} := \{f^{A^{-1}} \mid f \in C\}$ is a basis of $\mathrm{sat}(T)$. □

From now on, we assume that the coefficients of the matrix $M = (m_{ij})$ are pairwise different variables. We view the coefficients of M, as well as the coefficients of all polynomials, as elements of the field of rational functions $\mathbf{k}(m_{ij})$. Moreover, the base field $\mathbf{k}$ is either $\mathbb{R}$ or $\mathbb{C}$ so that the affine space $\overline{\mathbf{k}}^n$ is endowed with the Euclidean topology. In this context, we recall from [6] that the quasi-component $W(T)$ has the same closure in both the Euclidean and the Zariski topologies.

Theorem 14. *For all values of (m_{ij}) such that $V(r_T^A, r_C)$ is empty, we have*

$$\lim(W(T)) \quad = \quad \{A^{-1}(\mathbf{y}) \mid \mathbf{y} \in V(h_T^A) \cap W(C)\}. \tag{6.11}$$

Proof. Observe first that $V(r_T^A, r_C)$ is empty if and only if $V(r_T, r_c^{A^{-1}})$ is empty. Observe next that any zero $\zeta \in \overline{\mathbf{k}}^n$ of h_T extends a zero $\zeta' \in \overline{\mathbf{k}}^d$ of r_T, see [24]. Therefore, for any choice of the parameters (m_{ij}) such that $V(r_T^A, r_C)$ is empty, one can let $(x_1, \ldots, x_n)$ approach a given root of h_T while staying within a bounded open set of $W^{A^{-1}}(C)$ leading to finitely many (possibly zero) finite limits for $(x_1, \ldots, x_n)$. Since, by construction, the constructible sets $W^{A^{-1}}(C)$ and $W(T)$ have the same Zariski closure, it follows that the points of $V(h_T^A) \cap W(C)$ are the images by A of the desired limit points of $W(T)$. □

Example 12. *Consider the regular chain $T := \{x_4, x_2x_3 + x_1^2\} \subset \mathbb{Q}[x_1 < x_2 < x_3 < x_4]$ and the linear change of coordinates A corresponding to the matrix*

$$M = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

Using the extended of PALGIE*, we can compute $C := \{x_4, x_3^2 + x_2x_3 + x_1^2\}$ and consequently, $r_T^A = x_1^2$ and $r_C = 1$. Then $\langle r_T^A, r_C \rangle = \langle 1 \rangle$ holds. Using* Triangularize *command of Maple, one can get*

$$\langle C, h_T^A \rangle^{A^{-1}} \quad = \quad \langle x_4, x_2, x_1 \rangle \quad = \quad \lim(W(T)).$$

Corollary 2. *Assume that T generates* $\mathrm{sat}(T)$*. Then we have*

$$\lim(W(T)) = V(T) \setminus W(T) \tag{6.12}$$

Hence, $\lim(W(T))$ *can be obtained by set-theoretic operations on constructible sets. Moreover, generically, the set* $\lim(W(T))$ *is determined by $V(h_T^A) \cap W(C)$.*

Proof. We prove the first claim. The hypothesis implies $V(T) = \overline{W(T)}$. Since $V(T) = W(T) \cup (V(T) \cap V(h_T))$, the conclusion follows. The second claim follows immediately from Theorems 14 and 13. □

6.6 On the computation of $\lim(W(T))$ and $\mathrm{sat}(T)$

Let T be a regular chain whose saturated ideal has dimension d. A driving application of this chapter is the computation of $\lim(W(T))$. Section 6.5 was primarily dedicated to the case where T is a basis of its saturated ideal, while in the present section we replace this assumption by others. Recall that we have the follow equalities:

$$V(\mathrm{sat}(T)) = \overline{W(T)} = \left(\overline{W(T)} \cap V(h_T)\right) \cup W(T) = \lim(W(T)) \cup W(T).$$

Therefore, computing $\lim(W(T))$ and computing $V(\mathrm{sat}(T))$ are equivalent problems. Theorems 15, 16 and Lemma 29 below deal with the latter problem while Proposition 19 is concerned with the former. All these results make some assumption on T and we do not know a general procedure for computing either $\lim(W(T))$ or $V(\mathrm{sat}(T))$ that would avoid Gröbner basis calculation.

Lemma 28. *Let $\mathcal{I}$ be a radical ideal of $\mathbf{k}[\mathbf{x}]$. Let $h \in \mathbf{k}[\mathbf{x}]$. Assume that the dimension of any associated prime $\mathfrak{p}$ of $\mathcal{I}$ is at least d. Then $\dim(V(\mathcal{I}, h)) < d$ implies that h is regular modulo $\mathcal{I}$. If the dimension of any associated prime $\mathfrak{p}$ of $\mathcal{I}$ is d, that is, if $\mathcal{I}$ is an unmixed ideal of dimension d, then $\dim(V(\mathcal{I}, h)) < d$ holds if and only if h is regular modulo $\mathcal{I}$.*

PROOF. Let $\mathcal{I} = \cap_{i=1}^{s} \mathfrak{p}_i$, where $\mathfrak{p}_i$ are the associated prime of $\mathcal{I}$. Assume that $\dim(V(\mathcal{I}, h)) < d$, it is enough to show that h does not belong to any $\mathfrak{p}_i$. On the other hand, we have $V(\mathcal{I}, h) = \cup_{i=1}^{s} V(\mathfrak{p}_i, h)$. If h belongs to some $\mathfrak{p}_i$, then $V(\mathfrak{p}_i, h) = V(\mathfrak{p}_i)$. Since $\dim(\mathfrak{p}_i) \geq d$, we know that $\dim(V(\mathcal{I}, h)) \geq d$, which is a contradiction to the assumption that $\dim(\mathcal{I}, h) < d$.

If $\mathcal{I}$ is an unmixed ideal of dimension d, by the above argument, $\dim(V(\mathcal{I}, h)) < d$ implies that h is regular modulo $\mathcal{I}$. On the other hand, if h is regular modulo $\mathcal{I}$, then h does not belong to any $\mathfrak{p}_i$. Thus $\dim(V(\mathcal{I}, h)) = \max(\dim(V(\mathfrak{p}_i, h))) < \max(\dim(\mathfrak{p}_i)) = d$. □

Theorem 15. *Let $T \subset \mathbf{k}[\mathbf{x}]$ be a regular chain with free variables $x_1, \ldots, x_d$. Let h_T be the product of the initials of the polynomials in T. Then, we have $\sqrt{\langle T \rangle} = \sqrt{\mathrm{sat}(T)}$ if and only if $\dim(V(T, h_T)) < d$ holds.*

PROOF. First we claim that for any associated prime $\mathfrak{p}$ of $\sqrt{\langle T \rangle}$, we have $\dim(\mathfrak{p}) \geq n - d$. To prove this, we first notice that the associated primes $\mathfrak{p}$ of $\sqrt{\langle T \rangle}$ are exactly the minimal associated primes $\mathfrak{p}$ of $\langle T \rangle$. On the other hand, since $\langle T \rangle \subseteq \mathrm{sat}(T)$ and $V(\mathrm{sat}(T)) \neq \emptyset$ hold, we know that $\langle T \rangle$ generates a proper ideal. By Krull's principle ideal theorem, for any minimal associated prime $\mathfrak{p}$ of $\langle T \rangle$, the height of $\mathfrak{p}$ is less than or equal to $|T|$. Since $|T| = n - d$, we have $\dim(\mathfrak{p}) \geq n - d$. The claim is proved.

Now we prove that we have $\sqrt{\langle T \rangle} = \sqrt{\mathrm{sat}(T)}$ if and only if $\dim(V(T, h_T)) < d$ holds. First, we show that the condition is sufficient. If $\dim(V(T, h_T)) < d$ holds, with the previous claim and Lemma 28, we deduce that h_T is regular modulo $\sqrt{\langle T \rangle}$. Thus, we have $\sqrt{\mathrm{sat}(T)} = \sqrt{\langle T \rangle : h_T^{\infty}} = \sqrt{\langle T \rangle} : h_T^{\infty} = \sqrt{\langle T \rangle}$. Next, we show that the condition is necessary. If $\sqrt{\langle T \rangle} = \sqrt{\mathrm{sat}(T)}$, then $\sqrt{\langle T \rangle}$ is an unmixed ideal and h_T is regular modulo $\sqrt{\langle T \rangle}$. Thus, $\dim(V(T, h_T)) < d$ holds by Lemma 28. □

Remark 13. *As an immediate corollary, we have $V(T) = \overline{W(T)}$ if and only if $\dim(V(T, h_T)) < d$. There are many ways to compute the dimension of an algebraic set. In particular, this dimension can be determined by computing a Kalkbrener triangular decomposition. We denote by* IsClosure *a procedure to test $V(T) = \overline{W(T)}$, by applying Theorem 15.*

Example 13. *Consider the regular chain* $T := \{x_1x_2 + x_1, x_1x_3 + 1\}$ *of* $\mathbb{Q}[x_1 < x_2 < x_3]$. *Since the first polynomial is not primitive w.r.t.* x_2, T *is not a primitive regular chain in the sense of [57]. Since* $V(T, x_1) = \emptyset$ *holds, applying Theorem 15, we have* $\sqrt{\langle T\rangle} = \sqrt{\mathrm{sat}(T)}$. *Actually* $\langle T\rangle = \mathrm{sat}(T)$ *also holds.*

Theorem 16. *Let* T *be a regular chain of* $\mathbf{k}[\mathbf{x}]$ *with free variables* $x_1, \ldots, x_d$. *Let* $C_1, \ldots, C_s \subset \mathbf{k}[\mathbf{x}]$. *Assume that* $\langle C_i\rangle \subseteq \sqrt{\mathrm{sat}(T)}$ *holds, for all* $i = 1, \ldots, s$. *Let* $\mathcal{I} = \langle T, C_1, \ldots, C_s\rangle$. *Then* $\sqrt{\mathrm{sat}(T)} = \sqrt{\mathcal{I}}$ *if and only if there exist regular chains* T_i, $i = 1, \ldots, t$, *such that each of the following properties hold:*

(*i*) $\sqrt{\mathcal{I}} = \cap_{i=1}^{t} \sqrt{\mathrm{sat}(T_i)}$,

(*ii*) $|T_1| = \cdots = |T_t| = n - d$,

(*iii*) h_T *is regular modulo all* $\sqrt{\mathrm{sat}(T_i)}$.

Proof. The direction "⇒"obviously holds. Next we prove the direction "⇐". By (*i*) and (*ii*), we know that $\sqrt{\mathcal{I}}$ is an unmixed ideal of dimension d. Since h_T is regular modulo all $\sqrt{\mathrm{sat}(T_i)}$, by Lemma 28, we have $\dim(V(h_T, \mathrm{sat}(T_i))) < d$. Thus $\dim(V(\mathcal{I}, h_T)) < d$ holds. Applying Lemma 28 again, we know that h_T is regular modulo $\sqrt{\mathcal{I}}$. Thus $\sqrt{\mathcal{I}} = \sqrt{\mathcal{I}} : h_T^{\infty} = \sqrt{\mathcal{I} : h_T^{\infty}}$ holds. On the other hand, we have $\langle T\rangle \subseteq \mathcal{I}$, thus we deduce that $\sqrt{\mathrm{sat}(T)} \subseteq \sqrt{\mathcal{I}}$. Since $\mathcal{I} = \langle T, C_1, \ldots, C_s\rangle$ and $\langle C_i\rangle \subseteq \sqrt{\mathrm{sat}(T)}$, we also have $\mathcal{I} \subseteq \sqrt{\mathrm{sat}(T)}$. The theorem is proved. □

Remark 14. *In Theorem 16, if* $s = 0$, *then the theorem trivially holds for* $t = 1$ *and* $T_1 = T$. *In practice, for example in Algorithm 15, the polynomial sets* C_i, *for all* $i = 1, \ldots, s$, *are regular chains for different orderings such that* $\sqrt{\mathrm{sat}(C_i)} = \sqrt{\mathrm{sat}(T)}$ *holds. Let* $T_1, \ldots, T_t$ *be regular chains in the output of* Triangularize$(\mathcal{I})$. *Then* (*i*) *automatically holds. If condition* (*ii*) *is satisfied, then* $\overline{W(T)} = V(\mathcal{I})$ *holds if and only if* (*iii*) *holds, which is easy to check by computing iterated resultants of* h_T *w.r.t. the regular chains* T_i. *Thus, this theorem provides an algorithmic recipe which may compute* $\overline{W(T)}$ *in somes cases, see Algorithm 15.*

Example 14. *We illustrate Algorithm 15 on one example. Consider the regular chain* $T := \{x_2^5 - x_1^2, x_1x_3 - x_2^2(x_2 + 1)\}$ *of* $\mathbb{Q}[x_1 < x_2 < x_3]$. *Then* $V(T, x_1) := \{(x_1, x_2, x_3) \mid x_1 = x_2 = 0\}$, *whose dimension is* 1. *By Theorem 15, we know that* $V(T) \neq V(sat(T))$. *Let* $C := \{x_2x_3^2 - x_2^2 - 2x_2 - 1, x_3x_1 - x_2^3 - x_2^2\}$ *be another regular chain of* $\mathbb{Q}[x_2 < x_3 < x_1]$. *One can verify that* $\mathrm{sat}(C) = \mathrm{sat}(T)$ *holds. Let* $\mathcal{I} := \langle C, T\rangle$. *A Kalkbrener triangular decomposition of* $\mathcal{I}$ *w.r.t. the order* $x_1 < x_2 < x_3$ *consists only of one regular chain, which is* T *itself. Thus by Theorem 16, we have* $V(\mathrm{sat}(T)) = V(\mathcal{I})$.

Remark 15. *We selected* 22 *one-dimensional non-primitive regular chains to test Algorithm 15. For* 10 *of them, the algorithm could successfully compute* $\overline{W(T)}$. *We also tested some random examples. The random regular chains are generated as follows. We choose a pair of random polynomials with* 4 *variables and of total degree* 2. *Then we apply* Triangularize *to this pair, thus obtaining* 2*-dimensional regular chains. In this way, we generated* 20 *regular chains, out of which* 16 *turned out to be non-primitive regular chains. Algorithm 15 successfully computed* $\overline{W(T)}$ *for* 10 *of those* 16 *examples.*

Algorithm 15: Closure(T)

Input: A non-empty regular chain T of $\mathbf{k}[x_1 < \cdots < x_n]$.

Output: Return $\emptyset$ or a polynomial set G such that $\overline{W(T)} = V(G)$. If $\emptyset$ is returned, this means that the algorithm fails to compute $\overline{W(T)}$.

```
1  begin
2      G := ∅;
3      for i from 1 to n do
4          if i = 1 then
5              C := T;
6          else
7              let R be the ordering x_i < x_{i+1} < ··· < x_n < x_1 ··· < x_{i-1};
8              D := PALGIE(T, R);
9              if |D| ≠ 1 then
10                 return ∅
11             else
12                 let C be the only regular chain in D;
13         if IsClosure(C) then
14             return C;
15         else
16             G := G ∪ C;
17             D := Triangularize(G, mode = K)// compute a Kalkbrener
                   triangular decomposition of V(G)
18             if all regular chains in D have dimension d and h_T is regular w.r.t. each of
               them then
19                 return G
20     return ∅;
21 end
```

Lemma 29. *Let $T = \{t_2(x_1, x_2), t_3(x_1, x_3), \ldots, t_s(x_1, x_s)\}$ be a regular chain of $\mathbf{k}[x_1 < \cdots < x_s]$. Assume that for all $i = 2, \ldots, s$, the polynomial t_i is a primitive polynomial w.r.t. its main variable x_i. Then, the regular chain T generates its saturated ideal.*

Proof. To prove this lemma, it is enough to prove by induction that $\mathrm{sat}(T_i) = \langle T_i \rangle$, for $i = 2 \ldots, s$, where $T_i := \{t_2, \ldots, t_i\}$. The lemma clearly holds for $i = 2$. Assume that the regular chain T_{i-1} is generating its saturated ideal. If $\mathrm{tail}(t_i)$ is invertible modulo $\langle \mathrm{init}(t_i)\} \cup T_{i-1} \rangle$, then $\langle T_i \rangle = \mathrm{sat}(T_i)$ holds (see [57]). Suppose that $\mathrm{tail}(t_i)$ is not invertible modulo $\langle \{\mathrm{init}(t_i)\} \cup T_{i-1} \rangle$, then $\langle \{\mathrm{init}(t_i)\} \cup T_{i-1} \rangle$ generates a proper zero-dimensional ideal, since $\mathrm{init}(t_i)$ is regular modulo $\langle T_{i-1} \rangle$. Let $\mathfrak{p}$ be an associated prime of this ideal. If $\mathrm{tail}(t_i)$ is not regular modulo $\mathfrak{p}$, then all the coefficients of t_i belong to $\mathfrak{p}$. On the other hand, since $t_s(x_1, x_i)$ is primitive, the ideal formed by the coefficients of t_i is the field $\mathbf{k}$, a contradiction. □

Remark 16. *If a regular chain T has the same shape as in Lemma 29, except that the polynomials t_i are not necessarily primitive, for $i = 2, \ldots, s$, then by making all the polynomials t_i primitive, we obtain a new regular chain T' such that we have $\langle T' \rangle = \mathrm{sat}(T') = \mathrm{sat}(T)$.*

Example 15. *Let $T := \{x_3^2 - 2\,x_1, 3\,x_2^3 + 4\,x_1^2\} \subset \mathbb{Q}[x_1 < x_2 < x_3]$ be a 1-dimensional regular chain. As you can see both elements of T are primitive bivariate polynomials. Then Lemma 29 implies that T generates its saturated ideal.*

Example 16. *The above lemma clearly does not hold for regular chains with more than one free variable. Consider for example the regular chain $T := \{x_1x_3 + x_2, x_1x_4 + x_2\}$, where $x_1 < x_2 < x_3 < x_4$. It is clear that $x_4 - x_3 \notin \langle T \rangle$. However, one can prove that $x_4 - x_3 \in \mathrm{sat}(T)$ because $x_1x_4 + x_2 = x_1(x_4 - x_3)$ modulo $\langle x_1x_3 + x_2 \rangle$.*

Lemma 30. *Let $T \subset \mathbf{k}[\mathbf{x}]$ be a regular chain with free variable x_1. Let $C_2 = T$ and let C_i, for $3 \leq i \leq n$, be regular chains w.r.t. the order $x_1 < x_i < \mathbf{x} \setminus \{x_1, x_i\}$ such that $\sqrt{\mathrm{sat}(C_i)} = \sqrt{\mathrm{sat}(T)}$. Assume that all the polynomials of C_i are primitive w.r.t. their main variables for $i = 2, \ldots, n$. Then $\dim(V(C_2, \ldots, C_n)) = 1$ holds.*

Proof. By the fact that $\overline{W(T)} = \overline{W(C_i)}$, we know that $\overline{W(T)} \subseteq V(C_2, \ldots, C_n)$, which implies that $\dim(V(C_2, \ldots, C_n)) \geq 1$. Let c_i be the polynomial in C_i with the main variable x_i. Then the set $C := \{c_2, \ldots, c_n\}$ is clearly a regular chain since $\mathrm{init}(c_i) \in \mathbf{k}[x_1]$ holds for each $i = 2, \ldots, s$. Moreover C generates its saturated ideal by Lemma 29. Thus $\dim(V(C)) = 1$. Since $V(C_2, \ldots, C_n) \subseteq V(C)$, we know that $\dim(V(C_2, \ldots, C_n)) \leq 1$. Thus the lemma holds. □

Example 17. *Let $T := \{x_2^5 - x_1^4, x_1x_3 - x_2^2\}$ be a regular chain of $\mathbb{Q}[x_1 < x_2 < x_3]$. Let also $C := \{x_3^5 - x_1^3, x_3^2x_2 - x_1^2\}$ be a regular chain of $\mathbb{Q}[x_1 < x_3 < x_2]$ for which we have $\mathrm{sat}(C) = \mathrm{sat}(T)$. One can verify that $\dim(V(T, C)) = 1$. Indeed a Kalkbrener triangular decomposition of $T \cup C$ computed by the* Triangularize *command of* `RegularChains` *library w.r.t. the order $x_1 < x_2 < x_3$ is $\{T, D\}$, where $D := \{x_1, x_2, x_3\}$.*

It is easy to observe that the decomposition computed by Triangularize *is redundant, that is we have $\mathrm{sat}(T) \subseteq \mathrm{sat}(D)$ holds. By Theorem 16, we conclude that $\sqrt{\langle T, C \rangle} = \sqrt{\mathrm{sat}(T)}$. However, for this example, Algorithm 15 fails to compute the set G such that $\overline{W(T)} = V(G)$, since T and D do not have the same height.*

Lemma 30, Example 17 and Theorem 16 show that it is possible to compute $\mathrm{sat}(T)$ by a change of order of the variables. One might wonder if this is always true. In particular, we ask the following two questions.

Question 1. *Let $C_1, \ldots, C_n$ be regular chains of $\mathbf{k}[\mathbf{x}]$ w.r.t. the order $x_i < x_{i+1} < \cdots < x_n < x_1 \cdots < x_{i-1}$, for $i = 1, \ldots, n$. Assume that $\sqrt{\mathrm{sat}(C_1)} = \cdots = \sqrt{\mathrm{sat}(C_n)}$. Does $\sqrt{\mathrm{sat}(C_1)} = \sqrt{\langle \cup_{i=1}^{n} C_i \rangle}$ always hold?*

Question 2. *Let $C_1, \ldots, C_n$ be polynomial sets of $\mathbf{k}[\mathbf{x}]$ such that C_i is a regular chain for the order $x_i < x_{i+1} < \cdots < x_n < x_1 \cdots < x_{i-1}$, for $i = 1, \ldots, n$. Assume that $\sqrt{\mathrm{sat}(C_i)} = \sqrt{\mathrm{sat}(C_j)}$ for all $1 \leq i < j \leq n$. Let $P_i \in C_i$ be the polynomial of least rank. Let H_1 be the product of the initials of C_1. Does the relation*

$$\lim(W(C_1)) \;=\; V(C_1 \cup \{P_1, \ldots, P_n, H_1\})$$

always hold?

To answer the two questions, we investigated over 35 different polynomial systems, and all of them succeeded but two of which failed. Here is one of them.

Example 18. *Suppose $T := \{t_1, t_2\} \subset \mathbb{Q}[x_1 < x_2 < x_3 < x_4]$ is a regular chain of dimension two, where $t_1 = -93\,x_1x_2^2 + (53\,x_1 - 35)\,x_2 + 93\,x_1^3 - 26\,x_1^2 - 57\,x_1$ and $t_2 = 93\,x_1x_4 + \left((3233\,x_1 - 2135)\,x_2 + 5673\,x_1^3 + 213\,x_1^2 - 3477\,x_1\right)x_3 + \left(-530\,x_1^2 - 3091\,x_1\right)x_2 - 930\,x_1^4 + 6119\,x_1^3 + 570\,x_1^2 - 1767\,x_1$. One can verify that T does not generate its saturated ideal.*

Following the notations of Question 1, using `PALGIE`*, we will be able to compute regular chains C_i for $i = 1, \ldots, 4$ w.r.t the orders mentioned in Question 1. To see whether the statement of Question 1 is true or not, on one hand, we can find the Kalkbrener triangular decomposition $\{C_1, R_1, R_2\}$ for $V(\cup_{i=1}^{4} C_i)$ where $C_1 = T$, $R_1 := \{x_4 - 19, x_2, x_1\}$, and $R_2 := \{961\,x_4^2 + 42428\,x_4 + 279756, x_3, x_2, x_1\}$.*

On the other hand, using methods based on Gröbner bases computations to find a generator for $\mathrm{sat}(C_1)$, one can find the Kalkbrener triangular decomposition $\{C_1, R_1\}$ for $V(\mathrm{sat}(C_1))$.

Therefore, we have

$$V(\mathrm{sat}(C_1)) \;=\; W(C_1) \cup W(R_1) \;\neq\; V(\cup_{i=1}^{4} C_i) \;=\; W(C_1) \cup W(R_1) \cup W(R_2).$$

This shows that the statement of Question 1 is not true.

Furthermore,

$$V(C_1 \cup \{P_1, \ldots, P_4, H_1\}) \;=\; W(R_1) \;\cup\; W(R_2)$$

where H_1 is the product of the initials of C_1 and P_i is the polynomial in C_i with least rank for $i = 1, \ldots, 4$. But the correct limit points are only represented by R_1 which means $\lim(W(C_1)) \;\neq\; V(C_1 \cup \{P_1, \ldots, P_4, H_1\})$. Cosequently, for this example, the answer to both Questions 1 and 2 is negative.

In Example 18, as one can see, we computed the limit points plus some extra points. The extra component R_2 in this example is of dimension 0 while the limit points we are expecting are of dimension 1.

Proposition 19. *Let T be a regular chain such that* $\mathrm{sat}(T)$ *has dimension d and let $F \subset \mathrm{sat}(T)$ such that $V(T \cup F \cup \{h_T\})$ has dimension $d-1$ and is irreducible. Suppose also that* $\lim(W(T))$ *is not empty. Then, we have* $\lim(W(T)) = V(T \cup F \cup \{h_T\})$.

Proof. The proof is straightforward. □

Example 19. *Consider the regular chain $T := \{x_1\, x_3 + x_2, x_2\, x_4 + x_1\} \subset \mathbb{Q}[x_1 < x_2 < x_3 < x_4]$. One can consider F to be the regular chain computed by applying* `PALGIE` *to T w.r.t. the variable order $x_3 < x_4 < x_1 < x_2$ and consequently, "fish" the polynomial $x_3\, x_4 - 1 \in \mathrm{sat}(T)$. Then*

$$\begin{aligned} V(T \cup F \cup \{h_T\}) &= V(x_1\, x_3 + x_2, x_2\, x_4 + x_1, x_3\, x_4 - 1, x_1\, x_2) \\ &= V(x_1, x_2, x_3\, x_4 - 1) \\ &= \lim(W(T)). \end{aligned}$$

6.7 Conclusion

Among all the methods we have considered for computing $\lim(W(T))$ and $\mathrm{sat}(T)$, those based on linear changes of coordinates seem very promising. They are a good trick for finding a subset $F \subset \mathrm{sat}(T)$ such that $F \cup T$ is a basis of $\mathrm{sat}(T)$, see Proposition 19. To develop that direction further, we are currently investigating the following related questions:

- decide whether $\lim(W(T))$ is empty
- decide whether $W(R) \subseteq \lim(W(T))$ for a given regular chain.

Chapter 7

Tangent Cones of Space Curves

7.1 Introduction

Traditionally, standard bases, Gröbner bases and cylindrical algebraic decomposition are the fundamental tools of computational algebraic geometry. The computer algebra systems CoCoA, Macaulay 2, Magma, Reduce, Singular have well-developed packages for computing standard bases or Gröbner bases, on which they rely in order to provide powerful toolkits to algebraic geometers.

Recent progress in the theory of regular chains has exhibited efficient algorithms for doing local analysis on algebraic sets. One of the algorithmic strengths of the theory of regular chains is its *regularity test* procedure. In algebraic terms, this procedure decides whether a hypersurface contains at least one irreducible component of the zero set of the saturated ideal of a regular chain. Broadly speaking, this procedure separates the zeros of a regular chain that belong to a given hypersurface from those which do not. This regularity test permits to extend an algorithm working over a field into an algorithm working over a direct product of fields. Or, to phrase it in another way, it allows one to extend an algorithm working at a point into an algorithm working at a group of points.

Following that strategy, the authors of [62] have proposed an extension of Fulton's algorithm for computing the intersection multiplicity of two plane curves at the origin. To be precise, this chapter extends Fulton's algorithm in two ways. First, thanks to the regularity test for regular chains, the construction is adapted such that it can work correctly at any point in the intersection of two plane curves, whether this point has rational coordinates or not. Secondly, an algorithmic criterion, see Theorem 17, is proposed for reducing intersection multiplicity computation in arbitrary dimension to the case of two plane curves. This algorithmic criterion requires to compute the tangent cone $TC_p(C)$ of a space curve C at one of its points p. In principle, this latter problem can be handled by means of standard basis (or Gröbner basis) computation. Available implementations (like those in Magma or Singular) require that the point p is uniquely determined by the values of its coordinates. However, when decomposing a polynomial system, a point may be defined as one of the roots of a particular sub-system (typically a regular chain $\boldsymbol{h}$). Therefore, being able to compute the tangent cones of C at all its points defined by a given regular chain $\boldsymbol{h}$, becomes a desirable operation. Similarly, and as discussed in [62], another desirable operation is the computation of the intersection multiplicity

of a zero-dimensional algebraic set V at all its points defined by a given regular chain $\boldsymbol{h}$. This type of tangent cone computation is addressed in the present Chapter (see also [9]).

Tangent cone computations can be approached at least in two ways. First, one can consider the formulation based on homogeneous components of least degree, see Definition 21. The original algorithm of Mora [65] follows this point of view. Secondly, one can consider the more "intuitive" characterization based on limits of secants, see Lemma 31. This second approach, that we follow in this chapter, requires to compute limits of algebraic functions. For this task, we take advantage of [6] where the authors show how to compute the limit points of the quasi-component of a regular chain. This type of calculation can be used for computing the Zariski closure of a constructible set. In the present Chapter, it is used for computing tangent cones of space curves, thus providing an alternative to the standard approaches based on Gröbner bases and standard bases.

The contributions of the present chapter are as follows

1. In Section 7.3, we present a proof of our algorithmic criterion for reducing intersection multiplicity computation in arbitrary dimension to the plane case; this criterion was stated with no justification in [62].

2. In Section 7.4.1, with Lemma 32, under a smoothness assumption, we establish a natural method for computing $TC_p(C)$; as limit of intersection of tangent spaces.

3. In Section 7.4.2, we relax the assumption of Section 7.4.1 and exhibit an algorithm for computing $TC_p(C)$.

This latter algorithm is implemented, in the `AlgebraicGeometryTools` subpackage [5] of the `RegularChains` library which is available at `www.regularchains.org`. Section 7.4.4 offers different examples for computing tangent cones of space curves.

7.2 Preliminaries

Throughout this article, we denote by $\mathbf{k}$ a field with algebraic closure $\overline{\mathbf{k}}$, and by $\mathbb{A}^{n+1}(\overline{\mathbf{k}})$ the $(n+1)$-dimensional affine space over $\overline{\mathbf{k}}$, for some positive integer n. Let $\boldsymbol{x} := x_0, \ldots, x_n$ be $n+1$ variables ordered as $x_0 \succ \cdots \succ x_n$. We denote by $\mathbf{k}[\boldsymbol{x}]$ the corresponding polynomial ring. Let $\boldsymbol{h} \subset \mathbf{k}[\boldsymbol{x}]$ be a subset and $h \in \mathbf{k}[\boldsymbol{x}]$ be a polynomial. We say that h is *regular* modulo the ideal $\langle \boldsymbol{h} \rangle$ of $\mathbf{k}[\boldsymbol{x}]$ whenever h does not belong to any prime ideals associated with $\langle \boldsymbol{h} \rangle$, thus, whenever h is neither null nor a zero-divisor modulo $\langle \boldsymbol{h} \rangle$. The *algebraic set* of $\mathbb{A}^{n+1}(\overline{\mathbf{k}})$ consisting of the common zeros of the polynomials in $\boldsymbol{h}$ is written as $V(\boldsymbol{h})$. For $\mathbf{W} \subset \mathbb{A}^{n+1}(\overline{\mathbf{k}})$, we denote by $\mathbf{I}(\mathbf{W})$ the ideal of $\mathbf{k}[\boldsymbol{x}]$ generated by the polynomials vanishing at every point of $\mathbf{W}$. The ideal $\mathbf{I}(\mathbf{W})$ is radical and when $\overline{\mathbf{k}} = \mathbf{k}$ holds, Hilbert's Nullstellensatz states that $\sqrt{\langle \boldsymbol{h} \rangle} = \mathbf{I}(V(\boldsymbol{h}))$.

In the next two sections, we review the main concepts used in this chapter, namely tangent cones and regular chains. For the former, we restrict ourselves to tangent cones of a space curve and refer to [29] for details and the general[1] case. For the latter concept, we refer to [25], in particular for the specifications of the basic operations on regular chains.

[1] Note that in the book [25], and other classical algebraic geometry textbooks like [90], the tangent cone of an algebraic set at one of its points, is also an algebraic set. Two equivalent definitions appear in [25] and are recalled in Definition 21 and Lemma 31.

7.2.1 Tangent cone of a space curve

As above, let $\boldsymbol{h} \subset \mathbf{k}[\boldsymbol{x}]$. Define $\mathbf{V} := V(\boldsymbol{h})$ and let $p := (p_0, \ldots, p_n) \in \mathbf{V}$ be a point. We denote by $\dim_p(\mathbf{V})$ the maximum dimension of an irreducible component C of $\mathbf{V}$ such that we have $p \in C$. Recall that the *tangent space* of $V(\boldsymbol{h})$ at p is the algebraic set given by

$$T_p(\boldsymbol{h}) := V(\, \mathbf{d}_p\, (f) : f \in \mathbf{I}(\mathbf{V}))$$

where $\mathbf{d}_p\, (f)$ is the *linear part* of f at p, that is, the affine form $\frac{\partial f}{\partial x_0}(p)(x_0 - p_0) + \cdots + \frac{\partial f}{\partial x_n}(p)(x_n - p_n)$. Note that $T_p(\boldsymbol{h})$ is a linear space. We say that $V(\boldsymbol{h})$ is *smooth* at p whenever the dimension of $T_p(\boldsymbol{h})$ is $\dim_p(\mathbf{V})$ and *singular* otherwise. The *singular locus* of $V(\boldsymbol{h})$, denoted by $\mathrm{sing}(\boldsymbol{h})$, is the set of the points $p \in V(\boldsymbol{h})$ at which $V(\boldsymbol{h})$ is singular.

Let $f \in \mathbf{k}[\boldsymbol{x}]$ be a polynomial of total degree d and $p := (p_0, \ldots, p_n) \in \mathbb{A}^{n+1}(\overline{\mathbf{k}})$ be a point such that $f(p) = 0$ holds. Let $\boldsymbol{\alpha} = (\alpha_0, \ldots, \alpha_n) \in {N_{\geq}}^{n+1}$ be a $(n+1)$-tuple of non-negative integers. Denote: $(\boldsymbol{x} - p)^{\alpha} := (x_0 - p_0)^{\alpha_0} \cdots (x_n - p_n)^{\alpha_n}$, where $|\alpha| = \alpha_0 + \cdots + \alpha_n$ is the total degree of $\boldsymbol{x} - p$. Since the polynomial $f \in \mathbf{k}[\boldsymbol{x}]$ has total degree d, it writes as a $\mathbf{k}$-linear combination of the form:

$$f = \sum_{|\alpha|=0} c_{\alpha}(\boldsymbol{x} - p)^{\alpha} + \cdots + \sum_{|\alpha|=d} c_{\alpha}(\boldsymbol{x} - p)^{\alpha}$$

with all coefficients c_{α} belonging to $\mathbf{k}$. Each summand $\mathrm{hc}_p\, (f;\ j) := \sum_{|\alpha|=j} c_{\alpha}(\boldsymbol{x} - p)^{\alpha}$ is called the *homogeneous component in $\boldsymbol{x} - p$ of f in degree j* . Moreover, the *homogeneous component of least degree* of f in $\boldsymbol{x} - p$ is given by $\mathrm{hc}_p\, (f;\ \mathrm{min}) := \mathrm{hc}_p\, (f;\ j_{\mathrm{min}})$ where $j_{\mathrm{min}} = \mathrm{min}(j \in N_{\geq} : \mathrm{hc}_p\, (f;\ j) \neq 0)$.

Definition 21 (Tangent Cone of a Curve)**.** *Let $C \subset \mathbb{A}^{n+1}(\overline{\mathbf{k}})$ be a curve and $p \in C$ be a point. The* tangent cone of C *at a point p is the algebraic set denoted by $TC_p(C)$ and defined by $TC_p(C) = V(\mathrm{hc}_p\, (f;\ \mathrm{min}) : f \in \mathbf{I}(C))$.*

One can show that $TC_p(C)$ consists of finitely many lines, all intersecting at p.

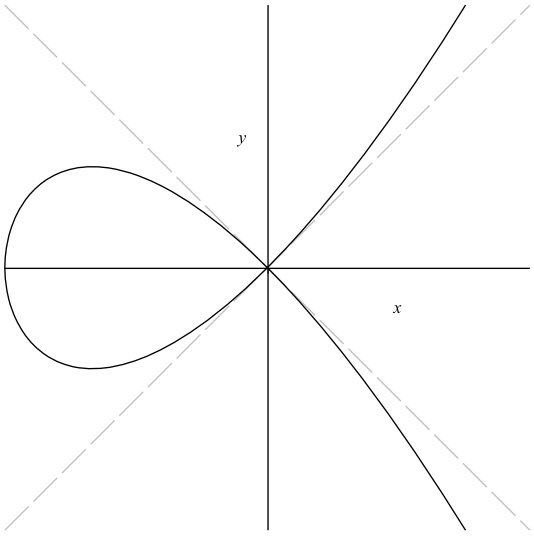


Figure 7.1: A figure displays the typical "fish" curve, which is a planar curve given by $h = y^2 - x^2(x+1) \in \mathbb{Q}[x, y]$. Clearly, two tangent lines are needed to form a "linear approximation" of the curve at the origin. Elementary calculations show these two lines actually form the tangent cone of the fish curve at the origin.

If $\mathbf{I}(C)$ is generated by a single polynomial then computing $TC_p(C)$ is easy. Otherwise, this is a much harder computation. Let $\boldsymbol{h} \subset \mathbf{k}[\boldsymbol{x}]$ be such that $V(\boldsymbol{h}) = C$. As pointed out by Mora et al. in [66], one can compute $\langle \text{HC}_p(f; \min) : f \in \mathbf{I}(C) \rangle$ via a graded Gröbner basis, say $\boldsymbol{G}$, of the *homogenization* of $\boldsymbol{h}$ (a process where an additional variable x_{n+1} is used to make every $h \in \boldsymbol{h}$ a homogeneous polynomial in $\mathbf{k}[\boldsymbol{x}][x_{n+1}]$). *Dehomogenizing* $\boldsymbol{G}$ by letting $x_{n+1} = 1$ produces the tangent cone of $\boldsymbol{h}$ (see Chapter 9.7, Proposition 4 in [29]).

Tangent cones are intimately related to the notion of intersection multiplicity that we review below. As mentioned in the introduction, computing intersection multiplicities is the main motivation of the algorithm presented in this chapter.

Definition 22. *Let* $\boldsymbol{h} \subset \mathbf{k}[\boldsymbol{x}]$. *The* intersection multiplicity *of* p *in* $V(\boldsymbol{h})$ *is defined by* $\text{im}(p; \boldsymbol{h}) := \dim_{vec}(\mathcal{O}/\langle \boldsymbol{h} \rangle)$ *where* $\mathcal{O} := \left\{ f/g : f, g \in \overline{\mathbf{k}}[\boldsymbol{x}],\ g(p) \neq 0 \right\}$ *is the* localization ring of $\mathbf{k}[\boldsymbol{x}]$ at p *and* $\dim_{vec}(\mathcal{O}/\langle \boldsymbol{h} \rangle)$ *is the dimension of* $\mathcal{O}/\langle \boldsymbol{h} \rangle$ *as a vector space over* $\mathbf{k}$. *Note by [30, Chapter 4.2, Proposition 11] we may substitute the power series ring* $\mathbf{k}[[\boldsymbol{x} - p]]$ *for* $\mathcal{O}$.

Example 20. *Let* $\boldsymbol{x} = [x, y, z]$ *and* $\boldsymbol{h} = \{x, y - z^3, z^2\left(z^4 + 1\right)\}$. *We have:*

$$\mathbf{k}[[\boldsymbol{x}]]/\langle \boldsymbol{h} \rangle = \mathbf{k}[[\boldsymbol{x}]]/\langle x,\ y - z^3,\ z^2 \rangle = \mathbf{k}[[\boldsymbol{x}]]/\langle x,\ y,\ z^2 \rangle = \{a + bz : a,\ b \in \mathbf{k}\}$$

implying $\text{im}(\mathbf{0}; \boldsymbol{h}) = 2$.

7.2.2 Regular chains

Broadly speaking, a *regular chain* of $\mathbf{k}[\boldsymbol{x}]$ is a system of equations and inequations defined by polynomials in $\mathbf{k}[\boldsymbol{x}]$ such that each equation specifies, in an implicit manner, the possible values of one of the variables x_i as a function of the variables of least rank, namely $x_{i+1}, \ldots, x_n$. Regular chains are a convenient way to describe the solution set of a polynomial system. More precise statements follow.

Let $h \in \mathbf{k}[\boldsymbol{x}]$ be a non-constant polynomial. The *main variable* of h is the largest variable $x \in \boldsymbol{x}$ (for the ordering $x_0 \succ \cdots \succ x_n$) such that h has a positive degree in x. The *initial* of h, denoted $\text{init}(h)$, is the *leading coefficient* of h w.r.t. its main variable. For instance the initial of $zx + t$ is x in $\mathbb{Q}[x \succ y \succ z \succ t]$ and 1 in $\mathbb{Q}[t \succ z \succ y \succ x]$.

Let $T \subset \mathbf{k}[\boldsymbol{x}]$ consist of non-constant polynomials. Then, the set T is said *triangular* if any two polynomials in T have different main variables. When T is a triangular set, denoting by I_T the product of the initials $\text{init}(f)$ for $f \in T$, we call *saturated ideal* of T, written $\text{sat}(T)$, the column ideal $\text{sat}(T) = \langle T \rangle : \text{I}_T^\infty$ and we call *quasi-component* of T the basic constructible set $\mathbf{W}(T) := V(T) \setminus V(\text{I}_T)$.

Definition 23 (Regular Chain)**.** *The triangular set* $T \subset \mathbf{k}[\boldsymbol{x}]$ *is a* regular chain *if either* T *is empty or* $T \setminus \{f\}$ *is a regular chain and the initial of* f *is regular modulo* $\text{sat}(T \setminus \{f\})$*, where* f *is the polynomial in* T *with largest main variable.*

Regular chains are used to decompose both algebraic sets and radical ideals, leading to two types of decompositions called respectively *Wu-Lazard* and *Kalkbrener* decompositions. More precisely, we have the following definition.

Finitely many regular chains $T_0, \ldots, T_e \subset k[\boldsymbol{x}]$ form a Kalkbrener decomposition of $\sqrt{\langle \boldsymbol{h} \rangle}$ (resp. a Wu-Lazard decomposition of $V(\boldsymbol{h})$) whenever we have $\sqrt{\langle \boldsymbol{h} \rangle} = \sqrt{\text{sat}(T_0)} \cap \cdots \cap$

$\sqrt{\text{sat}(T_e)}$ (resp. $V(\boldsymbol{h}) = \mathbf{W}(T_0) \cup \cdots \cup \mathbf{W}(T_e)$). These two types are different since the quasi-component of a regular chain T may not be an algebraic set. One should note that the Zariski closure of $\mathbf{W}(T)$ (that is, the intersection of all algebraic sets containing $\mathbf{W}(T)$) is the zero set (i.e. algebraic set) of $\text{sat}(T)$. One should observe, however, that if $\text{sat}(T)$ is zero-dimensional then the quasi-component $\mathbf{W}(T)$ and the algebraic set $V(T)$ coincide. Practically efficient algorithms computing both types of decompositions appear in [25].

Regular chains enjoy important algorithmic properties. One of them is the ability to test whether a given polynomial $f \in \mathbf{k}[\boldsymbol{x}]$ is regular or not modulo the saturated ideal of a regular chain $T \subset \mathbf{k}[\boldsymbol{x}]$. This allows us to specify an operation, called Regularize, as follows. The function call Regularize(f, T) computes regular chains $T_0, \ldots, T_e \subset \mathbf{k}[\boldsymbol{x}]$ such that $\sqrt{\text{sat}(T)} = \sqrt{\text{sat}(T_0)} \cap \cdots \cap \sqrt{\text{sat}(T_e)}$ holds and for $i = 0, \ldots, e$, either f is zero modulo $\text{sat}(T_i)$ or f is regular modulo $\text{sat}(T_i)$. When $\text{sat}(T)$ is zero-dimensional, one can give a simple geometrical interpretation to Regularize: this operation separates the points of $V(T)$ belonging to $V(f)$ from those which do not lie on $V(f)$.

7.3 Computing intersection multiplicities in higher dimension

Our interest in a standard-basis free algorithm for computing tangent cones comes by way of an overall goal to compute intersection multiplicities in arbitrary dimension. As mentioned in the introduction, in a previous paper [62], relying on the book of Fulton [36] and the theory of regular chains, authors derived an algorithm for computing intersection multiplicities of planar curves. They also sketched an algorithm criterion, see Theorem 17 below, for reducing the computation of intersection multiplicities in arbitrary dimension to computing intersection multiplicities in lower dimension. When applicable, successive uses of this criterion reduces intersection multiplicity computation in arbitrary dimension to the bivariate case.

Theorem 17. *For $\boldsymbol{h} = h_0, \ldots, h_{n-1}, h_n \in \mathbf{k}[\boldsymbol{x}]$ such that $V(h_0, \ldots, h_{n-1}, h_n)$ is zero-dimensional, for $p \in V(h_n)$, if the hyper-surface $V(h_n)$ is not singular at p and if that the tangent space π of $V(h_n)$ at p intersects transversally[2] the tangent cone of the curve $V(h_0, \ldots, h_{n-1})$ at p, then we have*

$$\text{im}(p;\, h_0, \ldots, h_{n-1},\, h_n) = \text{im}(p;\, h_0, \ldots, h_{n-1},\, \pi)\,,$$

hence, there is a polynomial map which takes $\boldsymbol{h}$ to a lower dimensional subspace while leaving the intersection multiplicity of $V(\boldsymbol{h})$ at p invariant.

Checking whether this criterion is applicable, requires to compute the tangent cone of the curve $V(h_0, \ldots, h_{n-1})$ at p, which motivates the present chapter. This algorithmic criterion was stated in [62] without justification, although the authors had a long and technical proof available in a technical report extending [62]. In the PhD thesis of the fourth author [100], a simpler proof was obtained.

[2] Two algebraic sets V_0 and V_1 in $\mathbb{A}^{n+1}(\overline{\mathbf{k}})$ *transversally intersect* at a point $p \in V_0 \cap V_1$ whenever their tangent cones intersect at $\{p\}$ *only* once or not at all. Note that if one of V_0 is a linear space, then it is its own tangent cone at p. Note also that, for a sake of clarity, we have restricted Definition 21 to tangent cones of curves, although tangent cones of algebraic sets of higher dimension are defined similarly, see [29].

7.4 Computing tangent lines as limits of secants

From now on, the coefficient field $\mathbf{k}$ is the field $\mathbb{C}$ of complex numbers and the affine space $\mathbb{A}^{n+1}(\mathbb{C})$ is endowed with both Zariski topology and the Euclidean topology. While Zariski topology is coarser than the Euclidean topology, we have the following key result (Corollary 1 in Section I.10 of Mumford's book [70]): For an irreducible algebraic set $\mathbf{V}$ and a subset $U \subseteq \mathbf{V}$ open in the Zariski topology induced on $\mathbf{V}$, the closure of U in Zariski topology and the closure of U in the Euclidean topology are both equal to $\mathbf{V}$. It follows that, for a regular chain $T \subset \mathbb{C}[\boldsymbol{x}]$ the closure of $\mathbf{W}(T)$ in Zariski topology and the closure of $\mathbf{W}(T)$ in the Euclidean topology are equal, thus both equal to $V(\mathrm{sat}(T))$. This result provides a bridge between techniques from algebra and techniques from analysis. The authors of [6] take advantage of Mumford's result to tackle the following problem: given a regular chain $T \subset \mathbb{C}[\boldsymbol{x}]$, compute the (non-trivial) limit points of the quasi-component of T, that is, the set $\lim(\mathbf{W}(T)) := \overline{\mathbf{W}(T)} \setminus \mathbf{W}(T)$.

In the present chapter, we shall obtain the lines forming the tangent cone of a space curve at a point by means of a limit computation process. And in fact, this limit computation will reduce to computing $\lim(\mathbf{W}(T))$ for some regular chain T. To this end, we start by stating the principle of our method in Section 7.4.1. Then, we turn this principle into an actual algorithm in Section 7.4.2 via an alternative characterization of a tangent cone, based on *secants*.

7.4.1 An algorithmic principle

Let $\boldsymbol{h} = \{h_0, \ldots, h_{n-1}\} \subset \mathbb{C}[\boldsymbol{x}]$ be n polynomials such that $C = V(\boldsymbol{h})$ is a curve, that is, a one-dimensional algebraic set. Let $p \in C$ be a point. The following proposition is well-known, see Theorem 6 in Chapter 9 of [29].

Lemma 31. *A line L through p lies in the tangent cone $TC_p(C)$ if and only if there exists a sequence $\{q_k : k \in N_{\geq}\}$ of points on $C \setminus \{p\}$ converging to p and such that the secant line L_k containing p and q_k becomes L when q_k approaches p.*

Under some mild assumption, we derive from Lemma 31 a method for computing $TC_p(C)$. We assume that for each $h \in \boldsymbol{h}$, the hyper-surface $V(h)$ is non-singular at p. This assumption allows us to approach the lines of $TC_p(C)$ with the intersection of the tangent spaces $T_q(h_0), \ldots, T_q(h_{n-1})$ when $q \in C$ is an sufficiently small neighborhood of p. A more precise description follows.

For each branch of a connected component $\mathcal{D}$ through p of $C = V(\boldsymbol{h})$ there exists a neighborhood B about p (in the Euclidean topology) such that $V(h_0), \ldots, V(h_{n-1})$ are all non-singular at each $q \in (B \cap \mathcal{D}) \setminus \{p\}$. Observe also that the singular locus $\mathrm{sing}(\mathcal{D})$ contains a *finite* number of points. It follows that we can take B small enough so that $B \cap \mathrm{sing}(\mathcal{D})$ is either empty or $\{p\}$. Define

$$v(q) := T_q(h_0) \cap \cdots \cap T_q(h_{n-1}),$$

where $T_q(h_i)$ is the tangent space of $V(h_i)$ at q.

Lemma 32 states that we can obtain $TC_p(C)$ by finding the limits of $v(q)$ as q approaches p. Since $TC_p(C)$ is the union of all the $TC_p(\mathcal{D})$, this yields a method for computing $TC_p(C)$.

Lemma 32. *The collection of limits of lines $v(q)$ as q approaches p in $(B \cap \mathcal{D}) \setminus \{p\}$ gives the tangent cone of $\mathcal{D}$ at q. That is to say*

$$TC_p(\mathcal{D}) = \lim_{q \to p} v(q) = \lim_{q \to p} T_q(h_0) \cap \cdots \cap T_q(h_{n-1}) \,.$$

Proof. There are two cases, either

1. $\mathcal{D}$ is *smooth* at p and $B \cap \operatorname{sing}(\mathcal{D}) = \emptyset$, or

2. $\mathcal{D}$ is *singular* at p and $B \cap \operatorname{sing}(\mathcal{D}) = \{p\}$.

Case 1. Assume $q \in B \cap \mathcal{D}$ is arbitrary and observe $\mathcal{D}$ is smooth within B and thereby the tangent cone of $\mathcal{D}$ is simply the tangent space (i.e. $TC_q(\mathcal{D}) = T_q(\mathcal{D})$).

Notice $T_q(\mathcal{D})$ is a sub-vector space of $v(q)$. Indeed, let $w \in T_q(\mathcal{D})$ be any tangent vector to $\mathcal{D}$ at q. As $\mathcal{D}$ is a curve in each $V(h)$ for $h \in \boldsymbol{h}$ it follows w is a vector tangent to each $V(h)$ as well. Correspondingly $w \in T_q(h)$ for any $h \in \boldsymbol{h}$ and thus $w \in v(q)$.

Finally, since $h_0, \ldots, h_{n-1}$ form a local complete intersection in B, we know $v(q)$ is a one-dimensional subspace of each $T_q(h_0)$. Since $w \in T_q(h)$ for each $h \in \boldsymbol{h}$, the vector w must span this subspace. Thus, for each $q \in B \cap \mathcal{D}$, we have

$$T_q(\mathcal{D}) = T_q(h_0) \cap \cdots \cap T_q(h_{n-1}) \,.$$

Taking the limit of each side of the above equality, when q approaches p and using again the fact that $\mathcal{D}$ is smooth at $q = p$, we obtain the desired result, that is, $TC_p(\mathcal{D}) = \lim_{q \to p} v(q)$.

Case 2. Assume $\mathcal{D} \cap B - \{p\}$ is a finite union of smooth curves $\mathcal{D}_0, \ldots, \mathcal{D}_j$. These are the smooth branches of $\mathcal{D} \cap B$ meeting at the singular point p. Each j corresponds to a unique line

$$L_j = \lim_{q \to p} v(q) \subset T_p(\mathcal{D})$$

as q approaches p *along* $\mathcal{D}_j$.

By Lemma 31 the tangent cone $TC_p(\mathcal{D})$ is the collection of limits to p of secant lines through p in $\mathcal{D}$. Such lines given by secants along $\mathcal{D}_j$ must coincide with L_j. More precisely

$$L_0 \cup \cdots \cup L_j \subset TC_p(\mathcal{D}) \,.$$

Because each $\mathcal{D}_j$ is smooth there is only one secant line for each j and thereby

$$L_0 \cup \cdots \cup L_j = TC_p(\mathcal{D})$$

as desired. □

7.4.2 Algorithm

Under a smoothness assumption, Lemma 32 states a principle for computing $TC_p(C)$. Let us now turn this principle into a precise algorithm and relax this smoothness assumption as well. To this end, we make use of Lemma 31.

Let q be a point on the curve $C = V(\boldsymbol{h})$ with coordinates $\boldsymbol{x}$. Further let $\widehat{pq}$ be a unit vector in the direction of $\overline{pq}$ (i.e. the line through p and q). To exploit Lemma 31 we must calculate the set

$$\left\{ \lim_{\substack{q \to p \\ q \neq p}} \widehat{pq} \right\},$$

which is indeed a set because C may have several branches through p yielding several lines in the tangent cone $TC_p(C)$.

Let $T \subset \mathbb{C}[\boldsymbol{y}][\boldsymbol{x}]$ be a zero-dimensional regular chain encoding[3] the point p, that is, such that we have $V(T) = \{p\}$. Note that the introduction of $\boldsymbol{y}$ for the coordinates of p is necessary because the "moving point" q is already using $\boldsymbol{x}$ for its own coordinates. Consider the polynomial set

$$\boldsymbol{s} = T \cup \boldsymbol{h}.$$

and observe that the ideal $\langle \boldsymbol{s} \rangle$ is one-dimensional in the polynomial ring $\mathbb{C}[x_{n-1} \succ \cdots \succ x_0 \succ y_{n-1} \succ \cdots \succ y_0]$. Let $T_0, \ldots, T_e \subset \mathbb{C}[\boldsymbol{y}][\boldsymbol{x}]$ be one-dimensional regular chains forming a Kalkbrener decomposition of $\sqrt{\langle \boldsymbol{s} \rangle}$. Thus we have

$$V(\boldsymbol{s}) = \overline{\mathbf{W}(T_0)} \cup \cdots \cup \overline{\mathbf{W}(T_e)}.$$

Computing with the normal vector $\widehat{pq}$ is unnecessary and instead we divide the vector $\overrightarrow{pq}$ by $x_n - y_n$. Since the n-th coordinate of $\frac{\overrightarrow{pq}}{x_n - y_n}$ is 1, this vector remains non-zero when q approaches p. However, this trick leads to a valid limit computation provided that $x_n - y_n$ vanishes finitely many times in $V(\boldsymbol{s})$. When this is the case, the lines of the tangent cone, that are not contained in the hyperplane $y_n = x_n$, can be obtained via limits of meromorphic functions (namely Puiseux series expansions) by letting x_n approach y_n and using the techniques of [6]. As we shall argue below, an ordering of $\boldsymbol{x}$, for which $x_n - y_n$ is regular, always exists. Hence, up to variable re-ordering, this tricks applies.

Since the tangent cone may have lines contained in the hyperplane $y_n = x_n$, additional computations are needed to capture them. There are essentially two options:

1. Perform a random linear change of the coordinates so as to assume that, generically, $y_n = x_n$ contains no lines of $TC_p(C)$.

2. Compute in turn the lines not contained in the hyperplane $y_i = x_i$ for all $i = 0, \ldots, n$ and remove the duplicates; indeed no lines of the tangent cone can simultaneously satisfy $y_i = x_i$ for all $i = 0, \ldots, n$.

Our experiments with theses two approaches suggest that, although the second one seems computationally more expensive, it avoids expression swell of the first one and is practically more efficient.

[3] In practice, we may use a zero-dimensional regular chain $T \subset \mathbb{C}[\boldsymbol{y}][\boldsymbol{x}]$ such that $\{p\} \subseteq V(T) \subseteq C$ holds. Then, the following discussion will bring the tangent cone at several points of C instead of p only.

From now on, we focus on computing the lines of the tangent cone *not* contained in the hyperplane $y_n = x_n$. We note that, deciding whether $x_n - y_n$ vanishes finitely many times in $V(\boldsymbol{s})$ can be done algorithmically by testing whether $x_n - y_n$ is regular modulo the saturated ideal of each regular chain $T_0, \ldots, T_e$. The operation Regularize described in Section 7.2 performs this task.

Consider now T_j, that is, one of the regular chains $T_0, \ldots, T_e$. Thanks to the specifications of Regularize, we may assume w.l.o.g. that either $x_n - y_n$ is regular modulo $\mathrm{sat}(T_j)$ or that $x_n - y_n \equiv 0 \mod \mathrm{sat}(T_j)$ holds.

Consider the latter case first. If $x_n - y_n \equiv 0 \mod \mathrm{sat}(T_j)$ then $\overline{\mathbf{W}(T_j)} \subseteq V(x_n - y_n)$ holds and we try to divide each component of $\overline{pq}$ by $x_{n-1} - y_{n-1}$ instead of $x_n - y_n$. A key observation is that there is $d \in [0, n]$ such that $x_d - y_d \not\equiv 0 \mod \mathrm{sat}(T_j)$ necessarily holds. Indeed, if $x_i - y_i \equiv 0 \mod \mathrm{sat}(T_j)$ would hold for all $i \in [0, n]$ then $\overline{\mathbf{W}(T_j)} \subset V(x_0 - y_0) \cap \cdots \cap V(x_n - y_n)$ would hold as well. Since the $\boldsymbol{y}$ coordinates are fixed by T, the algebraic set $\overline{\mathbf{W}(T_j)}$ would be zero-dimensional—a contradiction.

Hence, up to a variable renaming, we can assume that $x_n - y_n$ is regular modulo $\mathrm{sat}(T_j)$. Therefore, the algebraic set $V(x_n - y_n) \cap \overline{\mathbf{W}(T_j)}$ is zero-dimensional, thus, each component of $\overline{pq}$ is divisible by $x_n - y_n$, when q is close enough to p, with $q \neq p$. Define

$$m_0 = \frac{x_0 - y_0}{x_n - y_n}, \ldots, m_n = \frac{x_n - y_n}{x_n - y_n}.$$

and regard $\boldsymbol{m} = m_0, \ldots, m_n$ as new variables, that we call *slopes*, for clear reasons. Observe that the vector of coordinates $(m_0, \ldots, m_n, 1)$ is a normal vector of the secant line $\overline{pq}$. Thus, our goal is to "solve for" $\boldsymbol{m}$ when x_n approaches y_n with $(y_0, \ldots, y_n, x_0, \ldots, x_n) \in \mathbf{W}(T_j)$.

We turn this question into one computing the limit points of a one-dimensional regular chain, so as to use the algorithm of [6]. To this end, we extend the regular chain T_j to the regular chain $M_j \subset \mathbb{C}[\boldsymbol{m}][\boldsymbol{y}][\boldsymbol{x}]$ given by

$$M_j = T_j \cup \begin{cases} m_0(x_n - y_n) - (x_0 - y_0) \\ \vdots \\ m_n(x_n - y_n) - (x_n - y_n) \end{cases}.$$

Note that M_j is one-dimensional in this extended space and computing $\lim(\mathbf{W}(M_j))$, using the algorithm of [6], solves for $\boldsymbol{m}$ when $x_n \to y_n$ with $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbf{W}(T_0)$. Therefore and finally, the desired set $\{\lim_{q \to p, q \neq p} \widehat{pq}\}$ is obtained as the limit points of the quasi-components of $M_0, \ldots, M_n$.

Algorithm 16 describes the algorithm for computing the tangent cone of the curve C at the point p. Note that the function LimitPoints, is the function for computing $\lim(\mathbf{W}(M_j))$ which is described in Section 7.4.1, Algorithm 5.

Remark 17. *Observe that the above process determines the slopes $m_0, \ldots, m_n$ as roots of the top n polynomials of zero-dimensional regular chains in the variables $m_n \succ \cdots \succ m_0 \succ x_n \succ \cdots \succ x_0 \succ y_n \cdots \succ y_0$. Performing a change of variable ordering to $\boldsymbol{x} \succ \boldsymbol{m} \succ \boldsymbol{y}$ expresses $m_0, \ldots, m_{n-1}$ as functions of the coordinates of the point p only. We consider this a more desirable output.*

Algorithm 16: TangentCone

Input: The 0-dimensional regular chain $T \subset \mathbf{k}[x_0, \ldots, x_n]$ representing the point p on the curve C; the set $\boldsymbol{h}$ containing n polynomials and representing the curve C.

Output: The tangent cone of the curve C at the point p.

```
1  begin
2      Consider new independent variables y_0, ..., y_n;
3      Substitute the variables x_0, ..., x_n in T with y_0, ..., y_n, respectively;
4      {T_0, ..., T_e} := Triangularize(h ∪ T, R), where R is a variable ordering defined as
       R := x_n > ··· > x_0 > y_n > ··· > y_0;
5      lm := {};
6      for each T_i do
7          for ℓ from 0 to n do
8              if x_ℓ − y_ℓ is regular modulo T_i then
9                  Let m_j := (x_j − y_j)/(x_ℓ − y_ℓ), for j = 0, ..., n;
10                 Extend T_i with (x_ℓ − y_ℓ) m_j − (x_j − y_j) to a regular chain M_{i,ℓ};
11                 lm := lm ∪ LimitPoints(M_{i,ℓ}, R'), where R' is a variable ordering defined
                   as R' := m_n > ··· > m_0 > x_n > ··· > x_0 > y_n > ··· > y_0;
12     Remove redundant vectors such that all direction vectors in lm are pairwise colinear;
13     return (lm)
14 end
```

7.4.3 Equations of tangent cones

In the previous section, we saw how to compute the tangent cone $TC_p(C)$ in the form of the slopes of vectors defining the lines of $TC_p(C)$. Instead, one may prefer to obtain $TC_p(C)$ in the form of the equations of the lines of $TC_p(C)$. We explain below how to achieve this. Let S be an arbitrary point with coordinates $(X_0, \ldots, X_n)$. This point belongs to one of the lines of the tangent cone (corresponding to the branches of the curve defined by $\overline{\mathbf{W}(T_j)}$) if and only if the vectors

$$\frac{\overrightarrow{pq}}{x_n - y_n} = \begin{pmatrix} 1 \\ m_{n-1} \\ \vdots \\ m_0 \end{pmatrix} \quad \text{and} \quad \overline{pS} = \begin{pmatrix} X_n - y_n \\ X_{n-1} - y_{n-1} \\ \vdots \\ X_0 - y_0 \end{pmatrix}$$

are collinear. That is, if and only if we have the following relations

$$\begin{cases} X_n &= m_n(x_n - y_n) + y_n \\ &\vdots \\ X_0 &= m_0(x_n - y_n) + y_0. \end{cases} \tag{7.1}$$

Consider a regular chain (obtained with the process described in Remark 17) thus expressing the slopes $m_0, \ldots, m_{n-1}$ as functions of the coordinates $y_0, \ldots, y_n$ of p. Let us extend this regular chain with the relations from Equation (7.1), so as to obtain a one-dimensional regular chain in the variables $X_n \succ \cdots \succ X_0 \succ m_{n-1} \succ \cdots \succ m_0 \succ y_n \cdots \succ y_0$. Next, we eliminate the

variables $m_0, \ldots, m_{n-1}$, with the above equations. This is, indeed, legal since the only point of a line of the tangent cone where the equation $x_n = y_n$ holds is p itself. Finally, this elimination process consists simply of substituting $\frac{X_i - y_i}{x_n - y_n}$ for m_i into the equations defining $m_0, \ldots, m_n$.

7.4.4 Examples

The following examples illustrates our technique for computing tangent cones as limits.

Example 21. *Consider calculating the tangent cone of the fish in Figure 7.1 represented by the the set* $\boldsymbol{h} := \{x_1^2 - x_0^2(x_0 + 1)\}$ *at the origin, given by regular chain* $T := \{x_1, x_0\}$. *Based on Algorithm 16 we have:*

1. *First do a change of variables on* T *and obtain* $T' := \{y_1, y_0\}$.
2. *Then we obtain regular chain*

$$T_1 := \begin{cases} x_1^2 - x_0^2(x_0 + 1) \\ y_1 \\ y_0 \end{cases}$$

   which is a Kalkbrener decomposition of the set $\boldsymbol{h} \cup T'$ *w.r.t variable ordering* $x_1 \succ x_0 \succ y_1 \succ y_0$ *(Line 4 of Algorithm 16).*
3. *One can check that both* $x_0 - y_0$ *and* $x_1 - y_1$ *are regular modulo* T_1. *This can be done by using the* IsRegular *command of* `RegularChains` *Library.*
4. *Extend* T_1 *by the set* $\{m_0 - 1, (x_0 - y_0)m_1 - (x_1 - y_1)\}$ *and obtain the regular chain*

$$M_{1,0} := \begin{cases} (x_0 - y_0)m_1 - (x_1 - y_1) \\ m_0 - 1 \\ x_1^2 - x_0^2(x_0 + 1) \\ y_1 \\ y_0 \end{cases}$$

   w.r.t variable ordering $m_1 \succ m_0 \succ x_1 \succ x_0 \succ y_1 \succ y_0$ *(Line 10 of Algorithm 16).*
5. *Then by applying* LimitPoints *command on* $M_{1,0}$ *we obtain the following regular chains:*

$$lm_1 := \begin{cases} m_1 + 1 \\ m_0 - 1 \\ x_1 \\ x_0 \\ y_1 \\ y_0 \end{cases} \qquad lm_2 := \begin{cases} m_1 - 1 \\ m_0 - 1 \\ x_1 \\ x_0 \\ y_1 \\ y_0 \end{cases}$$

6. *By applying the same process for* $x_1 - y_1$, *we obtain the same set of regular chains* lm_1, lm_2.

Thus the fish curve represented by the set $\boldsymbol{h}$ *has two lines in its tangent cone at the origin with the slopes* 1 *and* -1.

Example 22. *Consider Figure 7.2, which is given by the zero set of* $\boldsymbol{h} = \{x_2^2 + x_1^2 + x_0^2 - 1,\ x_2^2 - x_1^2 - x_0\} \subset \mathbf{k}[x_2, x_1, x_0]$. *Let also* p *is the point represented by the zero dimensional regular chain* $T = \{x_2 + x_1,\ 2x_2^2 - 1,\ x_0\}$. *We want to compute the tangent cone of the curve represented by* $\boldsymbol{h}$ *at the point* p. *Based on Algorithm 16 we have:*

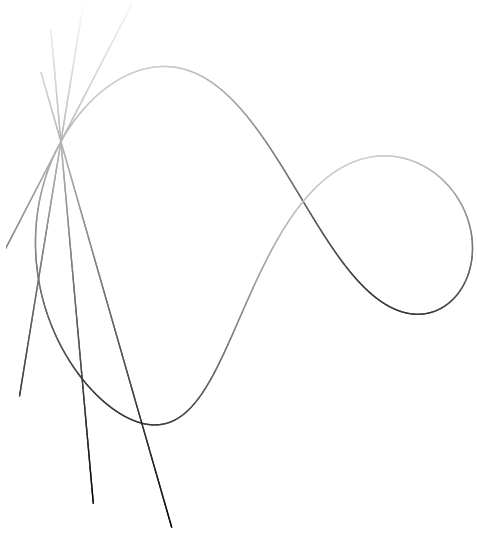


Figure 7.2: Limiting secants along $V(x^2 + y^2 + z^2 - 1,\ x^2 - y^2 - z)$.

1. *Substitute the variables x_2, x_1, x_0 in T with y_2, y_1, y_0 and obtain $T' := \{y_2+y_1,\ 2y_2^2-1,\ y_0\}$.*
2. *Then we obtain the regular chain*

$$T_1 := \begin{cases} 2x_2^2 + x_0^2 - x_0 - 1 \\ 2x_1^2 + x_0^2 + x_0 - 1 \\ y_2 + y_1 \\ 2y_2^2 - 1 \\ y_0 \end{cases}$$

which is a Kalkbrener decomposition of the set $\boldsymbol{h} \cup T'$ w.r.t variable ordering $x_2 \succ x_1 \succ x_0 \succ y_2 \succ y_1 \succ y_0$ (Line 4 of Algorithm 16).
3. *One can check that $x_j - y_j$, for $j = 1, 2, 3$, is regular modulo T_1.*

   - *Extend T_1 by the set $\{m_0 - 1, (x_0 - y_0)m_1 - (x_1 - y_1), (x_0 - y_0)m_2 - (x_2 - y_2)\}$ and obtain the regular chain*

$$M_{1,0} := \begin{cases} (x_0 - y_0)m_2 - (x_2 - y_2) \\ (x_0 - y_0)m_1 - (x_1 - y_1) \\ m_0 - 1 \\ x_1^2 - x_0^2(x_0 + 1) \\ y_1 \\ y_0 \end{cases}$$

   w.r.t variable ordering $m_2 \succ m_1 \succ m_0 \succ x_2 \succ x_1 \succ x_0 \succ y_2 \succ y_1 \succ y_0$ (Line 10 of Algorithm 16).

   - *Then by applying* LimitPoints *command on $M_{1,0}$ we obtain the following regular chains:*

$$lm_1 := \begin{cases} 2m_2 + y_1 \\ 2m_1 + y_1 \\ m_0 - 1 \\ x_2 + y_1 \\ x_1 + y_1 \\ x_0 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}$$

- *By applying the same process for $x_1 - y_1$, we obtain:*

$$lm_2 := \begin{cases} m_2 + 1 \\ m_1 - 1 \\ m_0 \\ x_2 - y_1 \\ x_1 + y_1 \\ x_0 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}, \quad lm_3 := \begin{cases} m_2 - 1 \\ m_1 - 1 \\ m_0 + 4y_1 \\ x_2 + y_1 \\ x_1 - y_1 \\ x_0 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}, \quad lm_4 := \begin{cases} m_2 \\ m_1 - 1 \\ m_0 \\ x_2 + y_1 \\ x_1 + y_1 \\ x_0 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}, \quad lm_5 := \begin{cases} 2m_2 + 2x_2y_1 + 1 \\ m_1 - 1 \\ m_0 - y_1 \\ 2x_2 - 1 \\ x_1 + y_1 \\ x_0 + 1 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}$$

- *And finally by applying the same process for $x_2 - y_2$ we obtain:*

$$lm_6 := \begin{cases} m_2 - 1 \\ m_1 - 1 \\ m_0 + 4y_1 \\ x_2 + y_1 \\ x_1 - y_1 \\ x_0 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}, \quad lm_7 := \begin{cases} m_2 - 1 \\ m_1 \\ m_0 \\ x_2 - y_1 \\ x_1 - y_1 \\ x_0 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}, \quad lm_8 := \begin{cases} m_2 - 1 \\ m_1 + 1 \\ m_0 \\ x_2 - y_1 \\ x_1 + y_1 \\ x_0 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}, \quad lm_9 := \begin{cases} m_2 - 1 \\ 2m_1 - 2x_1y_1 + 1 \\ m_0 - y_1 \\ x_2 - y_1 \\ 2x_1 + 1 \\ x_0 - 1 \\ y_2 + y_1 \\ 2y_1^2 - 1 \\ y_0 \end{cases}$$

In fact, in the computations above, there are some redundancies for computing the slopes. After removing the redundancies, we obtain the following slopes:

$$s_1 := \begin{cases} m_2 + 1 = 0 \\ m_1 - 1 = 0 \\ m_0 = 0 \end{cases}, \quad s_2 := \begin{cases} m_2 - 1 = 0 \\ m_1 + 1 = 0 \\ m_0 = 0 \end{cases}, \quad s_3 := \begin{cases} 2m_2x_1 = 0 \\ 2m_1 + x_1 = 0 \\ m_0 - 1 = 0 \end{cases}, \quad s_4 := \begin{cases} m_2 - 1 = 0 \\ m_1 - 1 = 0 \\ m_0 - 4x_1 = 0 \end{cases}$$

where x_1 corresponds to the x_1 coordinate of the solutions of the regular chain T which is $\{\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\}$.

Furthermore, based on Remark 17, one can obtain the equations of the lines contained in the tangent cone of $\boldsymbol{h}$ *as following:*

$$\begin{cases} X_0 = 0 \\ X_2 + X_1 = 0 \end{cases}, \quad \begin{cases} 4X_1x_1 + X_0 - 2 = 0 \\ X_2 - X_1 + 2x_1 = 0 \end{cases}$$

As one can see, since $x_1 \in \{\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\}$*, thus we have the equations of three different lines in the tangent cone of the curve represented by* $\boldsymbol{h}$ *at the point p. Nevertheless, we deduced four different direction vectors* s_1, s_2, s_3, s_4*. In fact, the vectors* s_1 *and* s_2 *are colinear and thus one of them is redundant. This justifies why we obtained three equations for the lines contained in tangent cone of* $\boldsymbol{h}$*.*

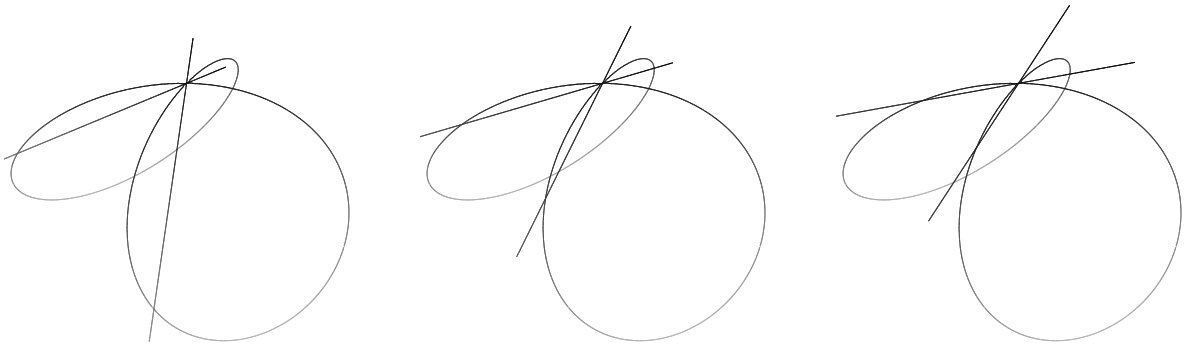


Figure 7.3: Secants along $V(x^2 + y^2 + z^2 - 1) \cap V(x^2 - y^2 - z(z-1))$ limiting to (0, 0, 1).

Example 23. *Consider the curve C in Figure 7.3 which is represented by* $\boldsymbol{h} = \{x_2^2 + x_1^2 + x_0^2 - 1,\ x2^2 - x_1^2 - x_0(x_0 - 1)\} \subset \mathbf{k}[x_2, x_1, x_0]$*. Consider the point* (0, 0, 1) *on this curve. Then using* TangentCone *command of* `RegularChains` *library, one can compute the slopes of the lines contained in the tangent cone of the curve C at the point* (0, 0, 1)*. To do this, it is required to pass the option* slopes *to the command* TangentCone*. These slopes are given by the systems below:*

$$\begin{cases} 3\,m_2^2 - 1 = 0 \\ m_1 - 1 = 0 \\ m_0 = 0 \end{cases}, \quad \begin{cases} m_2 - 1 = 0 \\ m_1^2 - 1 = 0 \\ m_0 = 0 \end{cases}$$

The command TangentCone *also returns the equations of the lines contained in the tangent cone of the curve C, by default, as following:*

$$\begin{cases} X_0 - 1 = 0 \\ X_1^2 - 3\,X_2^2 = 0 \end{cases}$$

7.5 Conclusion

We presented an alternative and Gröbner-free method for calculating the tangent cone of a space curve at any of its points. In essence, this is done by simulating a limit calculation along

a curve using variable elimination. From this limit we can construct each line of the tangent cone by solving for the vector of instantaneous slope along each tangents corresponding secant lines. Finally, this slope vector can be converted into equations of lines.

Chapter 8

Computing Limits of Multivariate Rational Functions

8.1 Introduction

Computing limits of functions is a basic task in multivariate calculus and many fundamental mathematical concepts are defined in terms of such limits. The case of univariate functions, including transcendental ones, has been well studied [40, 41, 80] and the corresponding algorithms are available in popular computer algebra systems. Surprisingly, the case of multivariate rational functions (i.e. quotients of polynomials) is still an active research area. Undergraduate students learn heuristics, like the adaptation of *L'hospital's rule* presented in [54] by G.R. Lawlor, but no general procedures for handling limits of multivariate rational functions.

Two recent papers have revitalized the search for such general procedures. In [104] S.J. Xiao and G.X. Zeng propose a first algorithm that, given a multivariate rational function $q \in \mathbb{Q}(X_1, \ldots, X_n)$, decides whether $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q$ is zero or not. The "not-case" includes the situation where $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q$ does not exist as well as the case where it exists but it is not zero. Their algorithm is based on the observation that the posed question can be phrased as a quantifier elimination problem, that the authors solve using triangular decomposition of algebraic systems, rational univariate representation as well as adjoining infinitesimal elements to the base field. A second algorithm and a result in the paper reduce the question of deciding whether $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q$ exists or not (and computing it, when it exists) to calling the first algorithm.

In [21], C. Cadavid, S. Molina and J.D. Vélez propose an algorithm, now available in Maple as the `limit/multi` command, for determining the existence and possible value of limits of the form $\lim_{(x,y)\to(0,0)} q$, where q is a bivariate rational function, and such that $(0, 0)$ is an isolated zero of the real algebraic set defined by the denominator of q. In a follow-up preprint [98], J.D. Vélez, P. Hernández and C. Cadavid extend the method of [21] to rational functions in three variables, still assuming that the origin is an isolated zero of the denominator. Both papers [21] and [98] rely on the key observation that, for determining the existence and possible value of limits of the form $\lim_{(x,y)\to(0,0)} q$ and $\lim_{(x,y,z)\to(0,0,0)} q$, it is sufficient to study *limits along a real algebraic set* $\chi(q)$, that is, limits of the form $\lim_{(x,y)\to(0,0),(x,y)\in\chi(q)} q$ and $\lim_{(x,y,z)\to(0,0,0),(x,y,z)\in\chi(q)} q$. This latter notion is defined in Section 8.3 of the present chapter . In the three variable case, the

method of [98] requires to compute the singular locus of $\chi(q)$ and the irreducible components of the algebraic set over $\mathbb{C}$ associated with $\chi(q)$.

The method of S.J. Xiao and G.X. Zeng [104] has the advantage of not making any assumptions on the number of variables nor the zero set of the denominator. Meanwhile, the works of C. Cadavid, S. Molina, J.D. Vélez and P. Hernández avoid the use of infinitesimal elements and rely on a deeper geometrical insight, through a notion of *discriminant variety*, see Notation 4; unfortunately, the recourse to singular loci and irreducible decomposition is a limitation in view of an implementation.

In this chapter , we propose an algorithm for determining the existence and possible value of $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q$, for an arbitrary number n of variables. As in [21] and [98], we firstassume that the origin is an isolated zero of the denominator of the rational function q, for Section 8.3 to Section 8.4. However, we avoid the computation of singular loci and decompositions into irreducible components of the real and complex algebraic sets involved in the method. Instead, we take advantage of the theory of regular chains and the RealTriangularize algorithm [23, 22] for decomposing semi-algebraic systems. The experimental results reported in Section 8.7 suggest that our algorithm can solve more problems than the algorithm of S.J. Xiao and G.X. Zeng, in particular when the number n of variables increases. For computing limit of real multivariate rational functions, when the origin is not an isolated zero of the denominator, we describe several results including multivariate L'Hospital rule in Section 8.6. However, the non-isolated case is work is progress We illustrate our algorithm for the case the origin is an isolated zero of the denominator with two examples.

Example 24. *Let $q \in \mathbb{Q}(x, y, z, w)$ be the rational function defined by $q(x, y, z, w) = \frac{z\,w+x^2+y^2}{x^2+y^2+z^2+w^2}$. We aim at computing $\lim_{(x,y,z,w)\to(0,0,0,0)} q$. A first step of the procedure consists in calculating the real algebraic set $\chi(q)$ such that our limit problem reduces to compute $\lim_{(x,y,z,w)\to(0,0,0,0),(x,y,z,w)\in\chi(q)} q$. The set $\chi(q)$, defined in Notation 4, is obtained by the method of Lagrange multipliers, see Section 8.2.1 and the proof of Lemma 33. The* RealTriangularize *algorithm yields the following decomposition: $\chi(q) = Z_{\mathbb{R}}(R_1) \cup Z_{\mathbb{R}}(R_2) \cup Z_{\mathbb{R}}(R_3) \cup Z_{\mathbb{R}}(R_4)$, where R_1, R_2, R_3, R_4 are respectively given by the* regular semi-algebraic systems *(see Section 8.2.4 for this term):*

$$\left\{\begin{array}{l} x = 0 \\ y = 0 \\ z = 0 \\ w = 0 \end{array}\right. , \left\{\begin{array}{l} z = 0 \\ w = 0 \end{array}\right. , \left\{\begin{array}{l} x = 0 \\ y = 0 \\ z + w = 0 \end{array}\right. \text{ and } \left\{\begin{array}{l} x = 0 \\ y = 0 \\ z - w = 0 \end{array}\right. .$$

For our purpose of limit computation, only R_2, R_3, R_4 are interesting, since they define either a curve or a surface passing through the origin, whereas R_1 is simply the origin.

Computing the limit of q along each of the curves $Z_{\mathbb{R}}(R_3)$ and $Z_{\mathbb{R}}(R_4)$ is achieved by a specific procedure presented in Section 5.2, based on Puiseux series. This procedure extends to the real case a technique presented in [6] for the complex case. On this particular example, evaluating $q(x, y, z, w)$ at $Z_{\mathbb{R}}(R_3)$ and $Z_{\mathbb{R}}(R_4)$, immediately yields the value of the limit in each case, which are $-\frac{1}{2}$ and $\frac{1}{2}$, respectively.

Now we focus on R_2 which consists simply of a regular chain, namely $T := \{t_1, t_2\}$ with $t_1 = z$ and $t_2 = w$. In order to compute the limit of $q(x, y, z, w)$ along $Z_{\mathbb{R}}(R_2)$, we apply again the method of Lagrange multipliers. More precisely, we wish to optimize $q(x, y, z, w)$ along $t_1(x, y, z, w) = t_2(x, y, z, w) = 0$ intercepted with a family of ellipsoids

$E_r(x,y,z,w) = 0$ with $E_r := A_1\,x^2 + A_2\,y^2 + A_3 z^2 + A_4\,w^2 - r^2$, where A_1, A_2, A_3, A_4 are positive values to be determined.

By definition of $\chi(q)$, the gradient $\nabla_{x,y,z,w} q$ is proportional to (x,y,z,w) along $\chi(q)$. Hence, in order to apply the Lagrange multipliers method, we need to check that the vectors $\nabla_{x,y,z,w} t_1$, $\nabla_{x,y,z,w} t_2$ and (x,y,z,w) are linearly independent almost everywhere on $Z_{\mathbb{R}}(R_2)$. By considering the following Jacobian matrix

$$\begin{bmatrix} x & y & z & w \\ \frac{\partial t_1}{\partial x} & \frac{\partial t_1}{\partial y} & \frac{\partial t_1}{\partial z} & \frac{\partial t_1}{\partial w} \\ \frac{\partial t_2}{\partial x} & \frac{\partial t_2}{\partial y} & \frac{\partial t_2}{\partial z} & \frac{\partial t_2}{\partial w} \end{bmatrix} = \begin{bmatrix} x & y & z & w \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

we see that the vectors $\nabla_{x,y,z,w} t_1$, $\nabla_{x,y,z,w} t_2$ and (x,y,z,w) are linearly independent as long as $x \neq 0$ or $y \neq 0$ holds. We choose to impose the constraint $y \neq 0$; using the incremental version of RealTriangularize*, we compute the intersection $Z_{\mathbb{R}}(R_2) \cap \{y \neq 0\}$ and obtain $Z_{\mathbb{R}}(R_5)$ where $R_5 := \{z = 0, w = 0, y \neq 0\}$. Now we choose $(A_1, A_2, A_3, A_4) = (3, 1, 2, 3)$ such that*

$$\begin{bmatrix} A_1 x & A_2 y & A_3 z & A_4 w \\ x & y & z & w \end{bmatrix}$$

is full rank. We are ready to apply the method of Lagrange multipliers, considering the following matrix

$$\begin{bmatrix} A_1 x & A_2 y & A_3 z & A_4 w \\ x & y & z & w \\ \frac{\partial t_1}{\partial x} & \frac{\partial t_1}{\partial y} & \frac{\partial t_1}{\partial z} & \frac{\partial t_1}{\partial w} \\ \frac{\partial t_2}{\partial x} & \frac{\partial t_2}{\partial y} & \frac{\partial t_2}{\partial z} & \frac{\partial t_2}{\partial w} \end{bmatrix} = \begin{bmatrix} 3x & 1y & 2z & 3w \\ x & y & z & w \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

which has a single non-zero minor, namely $m = 4xy$. Using again the incremental version of RealTriangularize*, we compute $Z_{\mathbb{R}}(R_5) \cap Z_{\mathbb{R}}(m = 0)$ and obtain $Z_{\mathbb{R}}(R_6)$, where $R_6 := \{x = 0, z = 0, w = 0, y \neq 0\}$. We are now in dimension one. Using the procedure of Section 5.2 (or, on this particular example, using substitution and elementary calculations) yields 1 as the limit along $Z_{\mathbb{R}}(R_6)$.*

Putting everything together, we have three different values for the limit of $q(x,y,z,w)$ along the three curves $Z_{\mathbb{R}}(R_3)$, $Z_{\mathbb{R}}(R_4)$ and $Z_{\mathbb{R}}(R_6)$. The corresponding values are $-\frac{1}{2}, \frac{1}{2}, 1$, which shows that the limit of q at the origin does not exist.

Example 25. *Let $q \in \mathbb{Q}(x,y,z)$ be the rational function defined by $q(x,y,z) = \frac{x^2 y z^2}{x^4 + z^4 + y^4}$. Here again, we aim at computing $\lim_{(x,y,z)\to(0,0,0)} q$.* RealTriangularize *produces the following decomposition of the real algebraic set $\chi(q)$: $\chi(q) = Z_{\mathbb{R}}(R_1) \cup Z_{\mathbb{R}}(R_2) \cup Z_{\mathbb{R}}(R_3) \cup Z_{\mathbb{R}}(R_4)$, where R_1, R_2, R_3, R_4 are respectively given by the regular semi-algebraic systems:*

$$\left\{ x = 0 \right. , \left\{ z = 0 \right. , \left\{ \begin{array}{l} x^2 - z^2 = 0 \\ y^6 + 3\,y^4\,z^2 - 2\,z^6 = 0 \\ z \neq 0 \end{array} \right. \text{ and } \left\{ \begin{array}{l} x = 0 \\ y = 0 \\ z = 0 \end{array} \right. .$$

The system R_4 can be discarded and the system R_3 has dimension one, hence the limit of q along $Z_{\mathbb{R}}(R_3)$ is handled by the procedure of Section 5.2, which yields 0.

We focus on R_1 and R_2. Similarly to the previous example, we consider the non-linear program consisting of optimizing $q(x, y, z)$ subject to $(x, y, z) \in Z_{\mathbb{R}}(R_1)$ (resp. $(x, y, z) \in Z_{\mathbb{R}}(R_2)$) and $E_r(x, y, z) = 0$ with $E_r := A_1\, x^2 + A_2\, y^2 + A_3 z^2 - r^2$, where A_1, A_2, A_3 are positive values to be determined.

Let $T := \{t_1\} = \{x\}$ be the regular chain part of R_1. Recall that $\nabla_{x,y,z} q$ is proportional to (x, y, z) along $\chi(q)$. Hence, we first determine when the following matrix

$$\begin{bmatrix} x & y & z \\ \frac{\partial t_1}{\partial x} & \frac{\partial t_1}{\partial y} & \frac{\partial t_1}{\partial z} \end{bmatrix} = \begin{bmatrix} x & y & z \\ 1 & 0 & 0 \end{bmatrix}$$

is full rank. The set of its 2-by-2 minors is $\{y, z, 0\}$; hence this matrix is full rank whenever $y \neq 0$ or $z \neq 0$ holds. Since $Z_{\mathbb{R}}(R_1) \not\subseteq Z_{\mathbb{R}}(y)$ holds, we impose the constraint $y \neq 0$ and compute $Z_{\mathbb{R}}(R_1) \cap \{y \neq 0\}$ yielding $Z_{\mathbb{R}}(R_5)$ with $R_5 := \{x = 0, y \neq 0\}$. Next, we let $(A_1, A_2, A_3) = (9, 10, 2)$ such that

$$\begin{bmatrix} A_1 x & A_2 y & A_3 z \\ x & y & z \end{bmatrix}$$

has at least one non-zero minor. Putting the three gradient vectors together, we form the following matrix

$$\begin{bmatrix} A_1 x & A_2 y & A_3 z \\ x & y & z \\ \frac{\partial t_1}{\partial x} & \frac{\partial t_1}{\partial y} & \frac{\partial t_1}{\partial z} \end{bmatrix} = \begin{bmatrix} 9x & 10y & 2z \\ x & y & z \\ 1 & 0 & 0 \end{bmatrix}.$$

Its determinant is $8\, y\, z$ and we compute $Z_{\mathbb{R}}(R_5) \cap \{y\, z = 0\}$ yielding $Z_{\mathbb{R}}(R_6)$ with $R_6 := \{x = 0, z = 0, y \neq 0\}$. The regular semi-algebraic system R_6 represents a space curve and the procedure of Section 5.2 computes the limit of q at the origin along $Z_{\mathbb{R}}(R_6)$, yielding 0. We proceed similarly with R_2. In this case, the non-linear programming trick yields the following space curve $\{y = 0, z = 0, x \neq 0\}$ along which the limit of q at the origin is also 0. Finally, the limit of q at origin exists and is equal to 0.

The chapter is organized as follows. In Section 8.2, we review different mathematical concepts and techniques used throughout the chapter . Regular chains (Section 8.2.2) and regular semi-algebraic systems (Section 8.2.4) are the key concepts. They have rich properties. A regular semi-algebraic system R can be seen as a parametrization of its zero $Z_{\mathbb{R}}(R)$. Therefore, the fundamental properties of parametric polynomial systems (Section 8.2.3) are essential to take full advantage of the theory of regular semi-algebraic systems.

Section 8.3 gathers lemmas which together with the proportions of Section 8.2, support the correctness and termination of our algorithm, which is presented in Sections 8.4 and 5.2. Section 8.6 targets comuting limits of real multivariate rational functions, when the origin is a non-isolated zero of the denominator. Sections 8.7 and 8.8 offer experimental results and conclusions.

8.2 Preliminaries

8.2.1 Lagrange multipliers

The following review is based on [96]. Let n, m be positive integers. Let Ω be an open set of $\mathbb{R}^n$, let $f, g_1, \ldots, g_m : \Omega \to \mathbb{R}$ be C^1 functions, let $b = (b_1, \ldots, b_m) \in \mathbb{R}^m$ and let x^* be a

point of Ω. Define $\Sigma_b := \{y \in \mathbb{R}^n \ : \ g_1(y) = b_1, \ldots, g_m(y) = b_m\}$. The point $x^\star$ is called a *local conditional extremal point of f under the constraints $g_1 = b_1, \ldots, g_m = b_m$*, whenever there exists a neighbourhood U of $x^\star$ such that $f(x)$ takes an extremal value (maximum or minimum) at $x = x^*$ on $\Sigma_b \cap U$. When this holds, the gradients $\nabla f(x^\star), \nabla g_1(x^\star), \ldots, \nabla g_m(x^\star)$ are linearly dependent. The above theorem is, in fact, usually stated when $m < n$ holds and $\nabla g_1(x^\star), \ldots, \nabla g_m(x^\star)$ are linearly independent, i.e. when $g(x^\star)$ is a regular value of the function $g = (g_1, \ldots, g_m)$ (at least if we restrict g to some neighbourhood V of $x^\star$). Then, the necessary condition of the theorem can be translated into the identity

$$\nabla f(x^\star) = \lambda_1^\star \nabla g_1(x^\star) + \ldots + \lambda_m^\star \nabla g_m(x^\star)\,, \tag{8.1}$$

for some real numbers $\lambda_1^\star, \ldots, \lambda_m^\star$ called *Lagrange multipliers*. Observe that, if we define the *Lagrange function*

$$F(x, \lambda) = f(x) + \sum_{i=1}^{m} \lambda_i (b_i - g_i(x))\,, \tag{8.2}$$

then the conditions (8.1) and $x^\star \in \Sigma_b$, for some neighbourhood U of $x^\star$ are equivalent to the fact that $(x^\star, \lambda^\star)$ is a critical point of F.

8.2.2 Regular chain theory

This section is a brief summary of concepts and algorithms for which details can be found in [25]. Throughout this chapter, $\mathbf{k}$ is a field of characteristic 0 and $\overline{\mathbf{k}}$ is its algebraic closure. We say that $\mathbf{k}$ is an *algebraic number field* if it is a finite degree field extension of the field $\mathbb{Q}$ of rational numbers. Here degree refers to the dimension of $\mathbf{k}$ as a vector space over $\mathbb{Q}$. Let $\mathbf{k}[X]$ be the polynomial ring over $\mathbf{k}$ and with ordered variables $X = X_1 < \cdots < X_n$. Let $p \in \mathbf{k}[X]$. Assume that $p \notin \mathbf{k}$. Denote by $\mathrm{mvar}(p)$, $\mathrm{init}(p)$, and $\mathrm{mdeg}(p)$ respectively the greatest variable appearing in p (called the *main variable* of p), the leading coefficient of p w.r.t. $\mathrm{mvar}(p)$ (called the *initial* of p), and the degree of p w.r.t. $\mathrm{mvar}(p)$ (called the *main degree* of p); denote by $\mathrm{discrim}(p)$ the discriminant of p w.r.t. $\mathrm{mvar}(p)$. For $F \subset \mathbf{k}[X]$, we denote by $\langle F \rangle$ and $V(F)$ the ideal generated by F in $\mathbf{k}[X]$ and the algebraic set of $\overline{\mathbf{k}}^n$ consisting of the common roots of the polynomials of F.

Triangular set. Let $T \subset \mathbf{k}[X]$ be a *triangular set*, that is, a set of non-constant polynomials with pairwise distinct main variables. Denote by $\mathrm{mvar}(T)$ the set of main variables of the polynomials in T. A variable $v \in X$ is called *algebraic* w.r.t. T if $v \in \mathrm{mvar}(T)$, otherwise it is said *free* w.r.t. T. If no confusion is possible, we shall always denote by $\underline{U} = U_1, \ldots, U_d$ and $\underline{Y} = Y_1, \ldots, Y_m$ the free variables and the main variables of T, respectively. We let $d = 0$ whenever T has no free variables. For $v \in \mathrm{mvar}(T)$, we denote by T_v and T_v^- the polynomial $f \in T$ with $\mathrm{mvar}(f) = v$ and the polynomials $f \in T$ with $\mathrm{mvar}(f) < v$, respectively. Let h_T be the product of the initials of the polynomials in T. We denote by $\mathrm{sat}(T)$ the *saturated ideal* of T: if T is the empty triangular set, then $\mathrm{sat}(T)$ is defined as the trivial ideal $\langle 0 \rangle$, otherwise it is the ideal $\langle T \rangle : h_T^\infty$. The *quasi-component* $W(T)$ of T is defined as $V(T) \setminus V(h_T)$. The Zariski closure of $W(T)$ in $\overline{\mathbf{k}}^n$, denoted by $\overline{W(T)}$, is the intersection of all algebraic sets $V \subseteq \overline{\mathbf{k}}^n$ such that $W(T) \subseteq V$ holds; moreover we have $\overline{W(T)} = V(\mathrm{sat}(T))$. For $f \in \mathbf{k}[X]$, we denote by $\mathrm{res}(f, T)$ the *iterated resultant* of f w.r.t. T, that is, f itself, if f is constant, or $\mathrm{res}(\mathrm{res}(f, T_v, v), T_v^-)$ if $v \in \mathrm{mvar}(T)$ and $v = \mathrm{mvar}(f)$ hold, or $\mathrm{res}(f, T_v^-)$ otherwise.

Regular chain. A triangular set $T \subset \mathbf{k}[X]$ is a *regular chain* if either T is empty, or letting v be the largest variable occurring in T, the set T_v^- is a regular chain, and the initial of T_v is regular (that is, neither zero nor zero divisor) modulo $\mathrm{sat}(T_v^-)$. Let $H \subset \mathbf{k}[X]$. The pair $[T, H]$ is a *regular system* if each polynomial in H is regular modulo $\mathrm{sat}(T)$. If H consists of a single polynomial h, then we also write $[T, h]$, for short, instead of $[T, H]$. The *dimension* of T is the dimension of its saturated ideal. A regular chain T, or a regular system $[T, H]$, is *square-free* if for all $t \in T$, the polynomial $\mathrm{der}(t)$ is regular w.r.t. $\mathrm{sat}(T)$, where $\mathrm{der}(t) = \frac{\partial t}{\partial v}$ and $v = \mathrm{mvar}(t)$. By $[T, H_{\neq}]$, we denote the algebraic system consisting of the equations $f = 0$ for all $f \in T$ and the inequations $h \neq 0$ for $h \in H \cup \{h_T\}$.

Triangular decomposition. Let $F \subset \mathbf{k}[X]$. Regular chains $T_1, \ldots, T_e$ of $\mathbf{k}[X]$ form a *triangular decomposition* of $V(F)$ in the sense of Kalkbrener (resp. Wu and Lazard) whenever we have $V(F) = \cup_{i=1}^{e} \overline{W(T_i)}$ (resp. $V(F) = \cup_{i=1}^{e} W(T_i)$). We denote by Triangularize an algorithm, such as the one of [25], computing a Kalkbrener triangular decomposition.

Regularization. Let $p \in \mathbf{k}[X]$ and $T \subset \mathbf{k}[X]$ be a regular chain. The function call Regularize(p, T) computes a set of regular chains $\{T_1, \ldots, T_e\}$ such that: (1) for each $i = 1, \ldots, e$, either $p \in \mathrm{sat}(T_i)$ holds or p is regular w.r.t. $\mathrm{sat}(T_i)$; (2) we have $\overline{W(T)} = \overline{W(T_1)} \cup \cdots \cup \overline{W(T_e)}$, and $\mathrm{mvar}(T) = \mathrm{mvar}(T_i)$ holds for each $i = 1, \ldots, e$.

Good specialization. Let $[T, H]$ be a square-free regular system of $\mathbf{k}[X]$. Recall that $\underline{Y}$ and $\underline{U} = U_1, \ldots, U_d$ stand respectively for $\mathrm{mvar}(T)$ and $X \setminus \underline{Y}$. Let $a = (a_1, \ldots, a_d)$ be a point of $\overline{\mathbf{k}}^d$. We say that $[T, H]$ *specializes well* at a if: (i) for each $t \in T$ the polynomial $\mathrm{init}(t)$ is not zero modulo the ideal $\langle U_1 - a_1, \ldots, U_d - a_d \rangle$; (ii) the image of $[T, H]$ modulo $\langle U_1 - a_1, \ldots, U_d - a_d \rangle$ is a square-free regular system.

Border polynomial [105]. Let $[T, H]$ be a square-free regular system of $\mathbf{k}[X]$. Let bp be the primitive and square free part of the product of all $\mathrm{res}(\mathrm{der}(t), T)$ $\mathrm{res}(h, T)$ for $h \in H$ and $t \in T$. We call bp the *border polynomial* of $[T, H]$. Proposition 20 follows from the specialization property of sub-resultants and states a fundamental property of border polynomials.

Proposition 20. *The system $[T, H]$ specializes well at $a \in \overline{\mathbf{k}}^d$ if and only if the border polynomial $bp(a) \neq 0$.*

8.2.3 Parametric polynomial systems

The following is based on [55] and [68]. In the sequel of this section, the field $\mathbf{k}$ is either $\mathbb{R}$ or $\mathbb{C}$. Let $f_1, \ldots, f_s, p_1, \ldots, p_r \in \mathbb{Q}[X]$, with, as before $X = X_1 < \cdots < X_n$. Consider the constructible set $C = \{x \in \mathbb{C}^n : f_1(x) = \cdots = f_s(x) = 0, p_1(x) \neq 0, \ldots, p_r(x) \neq 0\}$ and the semi-algebraic set $\mathcal{S} = \{x \in \mathbb{R}^n : f_1(x) = \cdots = f_s(x) = 0, p_1(x) > 0, \ldots, p_r(x) > 0\}$. Let $1 \leq d < n$. We view the variables $X_1, \ldots, X_d$ as *parameters* and we rename them as $\underline{U} = U_1, \ldots, U_d$. We denote by $\Pi_{\underline{U}}$ the canonical projection on the parameter space.

Discriminant variety in the complex case. Let δ be the dimension of $\overline{\Pi_{\underline{U}}(C)} = \overline{\Pi_{\underline{U}}(\overline{C})}$. An algebraic set $W \subset \mathbb{C}^d$ is a *discriminant variety* of C w.r.t. $\Pi_{\underline{U}}$ if the following four conditions hold:

1. $W \subseteq \overline{\Pi_{\underline{U}}(C)}$ holds,
2. $W = \overline{\Pi_{\underline{U}}(C)}$ holds if and only if $\Pi_{\underline{U}}^{-1}(u) \cap C$ is infinite for almost all $u \in \overline{\Pi_{\underline{U}}(C)}$,

3. the connected components $\mathcal{U}_1, \ldots, \mathcal{U}_k$ of $\overline{\Pi_{\underline{U}}(C)} \setminus W$ are analytic sub-manifolds of dimension δ, and
4. for all $1 \leq i \leq k$, the pair $(\Pi_{\underline{U}}^{-1}(\mathcal{U}_i), \Pi_{\underline{U}})$ is an analytic covering of $\mathcal{U}_i$.

This latter condition implies that there exists finitely many disjoint connected subsets $C_1, \ldots, C_{i_k}$ of $\mathbb{C}^n$ such that their union equals $\Pi_{\underline{U}}^{-1}(\mathcal{U}_i) \cap C$ and $\Pi_{\underline{U}}$ is a local diffeomorphism from C_j onto $\mathcal{U}_i$, for $1 \leq j \leq i_k$ and $1 \leq i \leq k$. Moreover, W contains the union of the critical values of the restriction of $\Pi_{\underline{U}}$ to the regular locus of C, as well as the projection of the singular locus of C.

Proposition 21 (Theorem 4, [68]). *Let $[T, H]$ be a square-free regular system of $\mathbf{k}[X]$ and bp its border polynomial. Then, the zero set of bp in $\mathbf{k}^d$ is the $\subseteq$-minimal discriminant variety of $[T, H_{\neq}]$ regarded as a parametric polynomial system, for which the parameters are the free variables of T.*

The real case. In practice, studying the parametric semi-algebraic system $\mathcal{S}$ can be done by

1. computing a discriminant variety W of the parametric constructible set C, and
2. applying the following proposition.

Proposition 22 (Corollary 1, [55]). *Assume that $W \neq \overline{\Pi_{\underline{U}}(C)}$ holds. Then, $(\overline{\Pi_{\underline{U}}(C)} \setminus W) \cap \mathbb{R}^d$ has finitely many connected components, $\mathcal{U}_1, \ldots, \mathcal{U}_e$, which are real analytic manifolds. Moreover, for each $i = 1, \ldots, e$, the number of points of $\mathcal{S}$ over $\mathcal{U}_i$ is constant, and if $\Pi_{\underline{U}}^{-1}(\mathcal{U}_i) \cap \mathcal{S}$ is not empty, then $(\Pi_{\underline{U}}^{-1}(\mathcal{U}_i) \cap \mathcal{S}, \Pi_{\underline{U}})$ is a real analytic covering of $\mathcal{U}_i$.*

8.2.4 Triangular decomposition of semi-algebraic sets

In this section, we recall that any semi-algebraic system decomposes into finitely many *regular semi-algebraic systems* (see Definition 24 for this term). For coherency with our software implementation, we assume the base field $\mathbf{k}$ of our polynomial coefficients is $\mathbb{Q}$ instead of $\mathbb{R}$. See [22] for details. Nevertheless, one can easily reduce the case where $\mathbf{k}$ is a real algebraic extension of $\mathbb{Q}$ to the case $\mathbf{k} = \mathbb{Q}$ by encoding this extension with a *regular semi-algebraic system* given by polynomials with coefficients in $\mathbb{Q}$.

Semi-algebraic system. Let us consider four finite polynomial subsets $F = \{f_1, \ldots, f_s\}$, $N = \{n_1, \ldots, n_t\}$, $P = \{p_1, \ldots, p_r\}$ and $H = \{h_1, \ldots, h_\ell\}$ of $\mathbb{Q}[X]$, where, as before, X stands for n ordered variables $X_1 < \cdots < X_n$. Let $N_{\geq}$ denote the set of the inequalities $\{n_1 \geq 0, \ldots, n_t \geq 0\}$. Let $P_{>}$ denote the set of the inequalities $\{p_1 > 0, \ldots, p_r > 0\}$. Let $H_{\neq}$ denote the set of inequations $\{h_1 \neq 0, \ldots, h_\ell \neq 0\}$. We will denote by $[F, P_{>}]$ the *basic semi-algebraic system* $\{f_1 = 0, \ldots, f_s = 0, p_1 > 0, \ldots, p_r > 0\}$ and by $S := [F, N_{\geq}, P_{>}, H_{\neq}]$ the semi-algebraic system (SAS) which is the conjunction of the following conditions: $f_1 = 0, \ldots, f_s = 0, n_1 \geq 0, \ldots, n_t \geq 0$, $p_1 > 0, \ldots, p_r > 0$ and $h_1 \neq 0, \ldots, h_\ell \neq 0$. The semi-algebraic set consisting of the zeros of S in $\mathbb{R}^n$ is denoted by $Z_{\mathbb{R}}(S)$ while the constructible set consisting of the zeros of $[F, N_{\geq}]$ in $\mathbb{C}^n$ is denoted by $Z_{\mathbb{C}}([F, N_{\geq}])$; if $N_{\geq}$ is empty we simply write $V(F)$ instead of $Z_{\mathbb{C}}([F, N_{\geq}])$. For an algebraic set $W \subseteq \mathbb{C}^n$, we denote by $W \cap \mathbb{R}^n$ the subset of $\mathbb{R}^n$ consisting of the points of W with real coordinates.

Definition 24. *Let $T \subset \mathbb{Q}[X]$ be a square-free regular chain. As before, let $\underline{U} = U_1, \ldots, U_d$ and $\underline{Y} = Y_1, \ldots, Y_{n-d}$ designate respectively the variables of X that are free w.r.t. T, and those that are algebraic w.r.t. T. With $P \subset \mathbb{Q}[X]$ as above, assume that each polynomial in P is regular*

w.r.t. sat(T). *Let Q be a quantifier-free formula over $\mathbb{Q}[X]$ involving only the $\underline{U}$ variables. Let O be the semi-algebraic subset of $\mathbb{R}^d$ defined by Q. We say that $R := [Q, T, P_>]$ is a* regular semi-algebraic system *if either $d = 0$ and the semi-algebraic system $[T, P_>]$ admits real solutions, or $d > 0$ and the following conditions hold:*

(*i*) *O is a non-empty open subset in $\mathbb{R}^d$,*
(*ii*) *the regular system $[T, P]$ specializes well at every point a of O,*
(*iii*) *at each point a of O, the specialized system $[T(a), P(a)_>]$ admits real solutions.*

The zero set *of R, denoted by $Z_{\mathbb{R}}(R)$, is the set of points $(a, \zeta) \in \mathbb{R}^d \times \mathbb{R}^{n-d}$ such that $Q(a)$ holds, and $t(a, \zeta) = 0$, $p(a, \zeta) > 0$ both hold for all $t \in T$ and all $p \in P$.*

Remark 18. *Using the notations of Definition 24, let $R = [Q, T, P_>]$ be a regular semi-algebraic system. Since O is open, each connected component C of O in $\mathbb{R}^d$ is locally homeomorphic to the hyper-cube $(0, 1)^d$. From Property* (*ii*)*, the zero set $Z_{\mathbb{R}}(R)$ consists of disjoint graphs of continuous semi-algebraic functions defined on each such C. Moreover, from Property* (*iii*)*, there is at least one such graph. For these reasons, the regular semi-algebraic system R can be understood as a parameterization of the set $Z_{\mathbb{R}}(R)$. Clearly, the dimension of $Z_{\mathbb{R}}(R)$ is d.*

Moreover, from Property (*ii*)*, together with Proposition 20 and Proposition 22, we deduce that for every connected component C of O, $(\Pi_{\underline{U}}^{-1}(C) \cap Z_{\mathbb{R}}(R), \Pi_{\underline{U}})$ is a real analytic covering of C. This implies that, at each point a of O, the Jacobian matrix of $T(a)$ is full rank.*

Proposition 23. *As above, let $S := [F, N_\geq, P_>, H_{\neq}]$ be a semi-algebraic system. Then, there exists a finite family of regular semi-algebraic systems $R_1, \ldots, R_e$ such that $Z_{\mathbb{R}}(S)$ equals the union of $Z_{\mathbb{R}}(R_1), \ldots, Z_{\mathbb{R}}(R_e)$. We call $R_1, \ldots, R_e$ a* triangular decomposition *of S and we denote by* RealTriangularize *an algorithm computing such a decomposition.*

Remark 19. *Expanding Remark 18, recall that we have observed that the dimension of $Z_{\mathbb{R}}(R)$ is d. In practice, this number is immediately deduced from the number of polynomials in the regular chain T. Indeed, we have $d = n - \#(T)$, where $\#(T)$ denotes the number of elements of T.*

An important feature of the RealTriangularize *algorithm [22] is the fact that triangular decompositions of semi-algebraic sets can be computed incrementally. Indeed, this algorithm relies on a procedure, called* Intersect*, such that, for a given semi-algebraic constraint (that is, either a polynomial equation, or a polynomial inequality) C, the function call* Intersect(R, C) *returns regular semi-algebraic systems $R_1, \ldots, R_e$ such that $Z_{\mathbb{R}}(R) \cap Z_{\mathbb{R}}(C)$ equals the union of $Z_{\mathbb{R}}(R_1), \ldots, Z_{\mathbb{R}}(R_e)$.*

The algorithms of Section 8.4 use another important procedure: for a regular semi-algebraic system R, one needs to check whether the origin $\underline{o}$ of $\mathbb{R}^n$ belongs or not to the closure of $Z_{\mathbb{R}}(R)$ in the Euclidean topology. The fact that the closure of $Z_{\mathbb{R}}(R)$ is a semi-algebraic set can be proved by constructing this set from other semi-algebraic sets by means if set-theoretic operations, as well as projection. Algorithms for those latter operations are described in [22] and [23]; this leads naturally to an algorithm for deciding whether or not $\underline{o}$ belongs to the closure of $Z_{\mathbb{R}}(R)$.

8.2.5 Puiseux series

This section is devoted to concepts and notations related to Puiseux series, taken from [35]. Let $\mathbf{k}$ be an algebraic number field and $\overline{\mathbf{k}}$ its algebraic closure. We denote by $\mathbf{k}[[X_1, \ldots, X_n]]$ and $\mathbf{k}\langle X_1, \ldots, X_n\rangle$ the respective rings of formal power series and convergent power series in $X_1, \ldots, X_n$ with coefficients in $\mathbf{k}$. When $n = 1$, we write U instead of X_1. Thus $\mathbf{k}[[U]]$ and $\mathbf{k}\langle U\rangle$ are the rings of formal and convergent univariate power series in U and coefficients in $\mathbf{k}$.

Puiseux series. We denote by $\mathbf{k}[[U^*]] = \bigcup_{\ell=1}^{\infty} \mathbf{k}[[U^{\frac{1}{\ell}}]]$ the ring of *formal Puiseux series*. Hence, given $\varphi \in \mathbf{k}[[U^*]]$, there exists $\ell \in \mathbb{N}_{>0}$ such that $\varphi \in \mathbf{k}[[U^{\frac{1}{\ell}}]]$ holds. Hence, we can write $\varphi = \sum_{m=0}^{\infty} a_m U^{\frac{m}{\ell}}$, for some $a_0, \ldots, a_m, \ldots \in \mathbf{k}$. We denote by $\mathbf{k}((U^*))$ the quotient field of $\mathbf{k}[[U^*]]$. Let $\varphi \in \mathbf{k}[[U^*]]$ and $\ell \in \mathbb{N}$ such that $\varphi = f(U^{\frac{1}{\ell}})$ holds for some $f \in \mathbf{k}[[U]]$. We say that the Puiseux series φ is *convergent* if we have $f \in \mathbf{k}\langle U\rangle$. Convergent Puiseux series form an integral domain denoted by $\mathbf{k}\langle U^*\rangle$; its quotient field is denoted by $\mathbf{k}(\langle U^*\rangle)$.

Puiseux parameterization. Let $f \in \mathbf{k}\langle X_1\rangle[X_2]$ be of positive degree d in X_2. A *Puiseux parametrization* of f is a pair $(\psi(U), \varphi(U))$ of elements of $\overline{\mathbf{k}}\langle U\rangle$ for some new variable U, such that

1. $\psi(U) = U^{\varsigma}$, for some $\varsigma \in \mathbb{N}_{>0}$;
2. $f(\psi(U), \varphi(U)) = 0$ holds in $\overline{\mathbf{k}}\langle U\rangle$, and
3. there is no integer $\ell > 1$ such that both $\psi(U)$ and $\varphi(U)$ are in $\overline{\mathbf{k}}\langle U^{\ell}\rangle$.

The index ς is called the *ramification index* of the parametrization $(U^{\varsigma}, \varphi(U))$. Assume that f is *general* in X_2 of order $k \geq 1$, that is, $f(0, X_2) \neq 0$ and the minimum degree of a term in $f(0, X_2)$ is k. Then, Puiseux's theorem guarantees that f admits Puiseux parameterizations and Newton-Puiseux's algorithm computes them. Assume further that f is monic in X_2. Then, there exist $\varphi_1, \ldots, \varphi_d \in \mathbf{k}\langle U\rangle$ such that we have $f(U^d, X_2) = (X_2 - \varphi_1(U)) \cdots (X_2 - \varphi_d(U))$.

8.3 Basic lemmas

Fix a real number $\rho > 0$ and let D_{ρ}^* be the punctured ball

$$D_{\rho}^* = \{(x_1, \ldots, x_n) \in \mathbb{R}^n \mid 0 < \sqrt{x_1^2 + \cdots + x_n^2} < \rho\}.$$

Let $q : \mathbb{R}^n \longrightarrow \mathbb{R}$ be a rational function defined on D_{ρ}^*.

Notation 4. *Let $\chi(q)$ be the subset of $\mathbb{R}^n$ (regarded as an affine space) where the gradient $\nabla_{x_1,\ldots,x_n} q$ of q at $(x_1, \ldots, x_n)$ and the vector $(x_1, \ldots, x_n)$ of $\mathbb{R}^n$ (regarded as a vector space) are co-linear. For $n = 2$, writing (x, y) for (x_1, x_2), we have*

$$\chi(q) = \{(x, y) \in \mathbb{R}^2 \mid y\tfrac{\partial q}{\partial x} - x\tfrac{\partial q}{\partial y} = 0\}.$$

In higher dimension, using McCoy theorem, the real algebraic set $\chi(q)$ is the vanishing locus of all 2-by-2 minors of the 2-by-n matrix which rows are $\nabla_{x_1,\ldots,x_n} q$ and $(x_1, \ldots, x_n)$.

Definition 25. *Let S be a semi-algebraic set of dimension at least 1 and such that the origin of $\mathbb{R}^n$ belongs to the closure $\overline{Z_{\mathbb{R}}(S)}$ of $Z_{\mathbb{R}}(S)$ in the Euclidean topology. Let $L \in \mathbb{R}$. We say that, when $(x_1, \ldots, x_n) \in \mathbb{R}^n$ approaches the origin along S, the limit of the rational function*

$q(x_1, \ldots, x_n)$ exists and equals L, whenever for all $\varepsilon > 0$, there exists $0 < \delta$ such that for all $(x_1, \ldots, x_n) \in S \cap D^*_\delta$ the inequality $|q(x_1, \ldots, x_n) - L| < \varepsilon$ holds. When this holds, we write

$$\lim_{\substack{(x_1,\ldots,x_n) \to (0,\ldots,0) \\ (x_1,\ldots,x_n) \in S}} q(x_1, \ldots, x_n) = L$$

Lemma 33 is a direct generalization of Proposition 1 in [21] and Lemma 36 is a less direct generalization of one of the properties established in Proposition 17 of [21]. We provide proofs for those results since they are essential for understanding the algorithm presented in Section 8.4. Meanwhile, Lemma 34 follows from Lemma 33 and elementary reasoning about limits; hence we omit its proof.

Lemma 33. *For all $L \in \mathbb{R}$ the following two assertions are equivalent:*

(*i*) $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q(x_1, \ldots, x_n)$ *exists and equals L,*

(*ii*) $\lim\limits_{\substack{(x_1,\ldots,x_n) \to (0,\ldots,0) \\ (x_1,\ldots,x_n) \in \chi(q)}} q(x_1, \ldots, x_n)$ *exists and equals L.*

PROOF. Clearly the first assertion implies the second one. Next, we assume that the second one holds and we prove that the first one holds as well. Hence, we assume that for all $\varepsilon > 0$, there exists $0 < \delta < \rho$ such that for all $(x_1, \ldots, x_n) \in \chi(q) \cap D^*_\delta$ the inequality $|q(x_1, \ldots, x_n) - L| < \varepsilon$ holds. We fix $\varepsilon > 0$. For every $r > 0$, we define $C_r = \{(x_1, \ldots, x_n) \in \mathbb{R}^n \mid \sqrt{x_1^2 + \cdots + x_n^2} = r\}$. For all $0 < r < \rho$, we choose $t_1(r)$ (resp. $t_2(r)$) minimizing (resp. maximizing) q on C_r. Applying the method of Lagrange multipliers, we have $t_1(r), t_2(r) \in \chi(q)$, for all $0 < r < \rho$. Observe that for all $(x_1, \ldots, x_n) \in \mathbb{R}^n$, with $r := \sqrt{x_1^2 + \cdots + x_n^2} < \rho$, we have $q(t_1(r)) - L \leq q(x_1, \ldots, x_n) - L \leq q(t_2(r)) - L$. From the assumption and the definitions of $t_1(r), t_2(r)$, there exists $0 < \delta < \rho$ such that, for all $r < \delta$, we have $-\varepsilon < q(t_1(r)) - L$ and $q(t_2(r)) - L < \varepsilon$. Therefore, there exists $0 < \delta < \rho$ such that for all $(x_1, \ldots, x_n) \in D^*_\delta$ the inequality $|q(x_1, \ldots, x_n) - L| < \varepsilon$ holds. □

Lemma 34. *Let $R_1, \ldots, R_e$ be regular semi-algebraic systems forming a triangular decomposition of $\chi(q)$ in the sense of Proposition 23. Then, for all $L \in \mathbb{R}$ the following two assertions are equivalent:*

(*i*) $\lim\limits_{\substack{(x_1,\ldots,x_n) \to (0,\ldots,0) \\ (x_1,\ldots,x_n) \in \chi(q)}} q(x_1, \ldots, x_n)$ *exists and equals L.*

(*ii*) *for all $i \in \{1, \ldots, e\}$ such that $Z_{\mathbb{R}}(R_i)$ has dimension at least 1 and the origin belongs to $\overline{Z_{\mathbb{R}}(R_i)}$, we have* $\lim\limits_{\substack{(x_1,\ldots,x_n) \to (0,\ldots,0) \\ (x_1,\ldots,x_n) \in Z_{\mathbb{R}}(R_i)}} q(x_1, \ldots, x_n)$ *exists and equals L.*

Notation 5. *For any complex algebraic set (resp. real algebraic set) $S \subseteq \mathbb{C}^n$ (resp. $S \subseteq \mathbb{R}^n$) we denote by* $\mathrm{Sing}(S)$ *the singular locus of S.*

Lemma 35. *Let $h \in \mathbb{R}[X_1, \ldots, X_n]$ be of positive degree in X_n. Let $U_0 \subset \mathbb{R}^{n-1}$ be a neighbourhood of the origin such that there exists a real number λ such that $\nabla h(p) = \lambda p$ holds for all $p \in U_0$. Furthermore assume that both the leading coefficient c of h in X_n and the discriminant Δ of h in X_n vanish nowhere on U_0. Then, for every smooth function $u : U_0 \longrightarrow \mathbb{R}$ for which $h(x_1, \ldots, x_{n-1}, u(x_1, \ldots, x_{n-1})) = 0$ holds, for all $(x_1, \ldots, x_{n-1}) \in U_0$, then, the graph of u is contained in a sphere centred at the origin.*

Proof. We view h as a parametric polynomial in X_n with $X_1, \ldots, X_{n-1}$ as parameters. Since the leading coefficient c of h in X_n and the discriminant Δ of h in X_n vanish nowehere on U_0, it follows from Section 8.2.3 that the intersection of U_0 and the discriminant variety of h is empty. Therefore, there exists a smooth analytic function $u : U_0 \longrightarrow \mathbb{R}$ such that Property (i) holds. Let

$$W = \{(X_1, \ldots, X_{n-1}, X_n) \mid X_1, \ldots, X_{n-1} \in U_0 \text{ and } X_n = u(X_1, \ldots, X_{n-1})\}.$$

Thus, the set W is the graph of u. For any $t \in W$, the normal vector at t is given by

$$n(t) = \frac{(-\partial u/\partial X_1, \ldots, -\partial u/\partial X_{n-1}, 1)}{\sqrt{(\partial u/\partial X_1)^2 + \cdots + (\partial u/\partial X_{n-1})^2 + 1}}.$$

Now, let $\mu := \lambda/\|\nabla h(t)\|$, then $\nabla h(t) = \mu\|\nabla h(t)\|t$. Therefore, from $t \in W$, we deduce that $n(t)$ can also be written as

$$n(t) = \frac{(X_1, \ldots, X_{n-1}, u(X_1, \ldots, X_{n-1}))}{\sqrt{X_1^2 + \cdots + X_{n-1}^2 + u^2(X_1, \ldots, X_{n-1})}}$$

which results in the following equalities:

$$\begin{cases} \frac{X_i}{\sqrt{X_1^2+\cdots+X_{n-1}^2+u^2(X_1,\ldots,X_{n-1})}} = -\frac{\partial u/\partial X_i}{\sqrt{(\partial u/\partial X_1)^2+\cdots+(\partial u/\partial X_{n-1})^2+1}} \quad , \quad i = 1, \ldots, n-1 \\ \frac{u(X_1,\ldots,X_{n-1})}{\sqrt{X_1^2+\cdots+X_{n-1}^2+u^2(X_1,\ldots,X_{n-1})}} = \frac{1}{\sqrt{(\partial u/\partial X_1)^2+\cdots+(\partial u/\partial X_{n-1})^2+1}} \end{cases} \tag{8.3}$$

The last equality in (8.3) implies that we have:

$$u(X_1, \ldots, X_{n-1}) = \frac{\sqrt{X_1^2 + \cdots + X_{n-1}^2 + u^2(X_1, \ldots, X_{n-1})}}{\sqrt{(\partial u/\partial X_1)^2 + \cdots + (\partial u/\partial X_{n-1})^2 + 1}}$$

Consequently, we obtain the following system of PDEs:

$$\left\{ u(X_1, \ldots, X_{n-1})\, \partial u/\partial X_i = -X_i \quad , \text{ for } i = 1, \ldots, n-1. \right.$$

Now for $i = 1$, we integrate both sides of Equation (8.3) with respect to X_1. There exits a differentiable function $F_2(X_2, \ldots, X_{n-1})$ such that

$$\frac{u^2(X_1, \ldots, X_{n-1})}{2} = \frac{-X_1^2}{2} + F_2(X_2, \ldots, X_{n-1}). \tag{8.4}$$

Now by taking derivative of both sides of Equation (8.4) with respect to X_2, we have $u\, \partial u/\partial X_2 = \partial F_2/\partial X_2$. After substitution of the latter equality in the equation $u\, \partial u/\partial X_2 = -X_2$, there exits a differentiable function $F_3(X_3, \ldots, X_{n-1})$ such that

$$\frac{-X_2^2}{2} = F_2(X_2, \ldots, X_{n-1}) + F_3(X_3, \ldots, X_{n-1}).$$

By continuing the same process, we have

$$\frac{-X_{i-1}^2}{2} = F_{i-1}(X_{i-1}, \ldots, X_{n-1}) + F_i(X_i, \ldots, X_{n-1}).$$

for $i = 2, 3, \ldots n-2$. But for $i = n-1$, we have $u\,\partial u/\partial X_{n-1} = \partial F_{n-1}/\partial X_{n-1}$. After substitution of the latter equality in $u\,\partial u/\partial X_{n-1} = -X_{n-1}$, there exits a constant C such that

$$\frac{-X_{n-1}^2}{2} = F_{n-1}(X_{n-1}) + C.$$

Therefore

$$\frac{u^2(X_1, \ldots, X_{n-1})}{2} = -\frac{X_1^2}{2} - \cdots - \frac{X_{n-1}^2}{2} + C.$$

Let $\alpha = (\alpha_1, \ldots, \alpha_{n-1}, \alpha_n)$ be a point of W. Since $u(\alpha_1, \ldots, \alpha_{n-1}) = \alpha_n$, we have $C = 1/2(\alpha_1^2 + \cdots + \alpha_n^2)$. So

$$u(X_1, \ldots, X_{n-1}) = \sqrt{r^2 - X_1^2 - \cdots - X_{n-1}^2},$$

where $r^2 := \alpha_1^2 + \cdots + \alpha_n^2$. Then we conclude W is a neighbourhood of $p \in D_\rho^*$ which is part of a sphere centred at the origin. □

Lemma 36. *Assume $n \geq 3$. Let $S = [Q, \{t_n\}, P_>]$ be a regular semi-algebraic system of $\mathbb{Q}[X_1, \ldots, X_n]$ such that $Z_\mathbb{R}(S)$ has dimension $d := n-1$, and the closure $\overline{Z_\mathbb{R}(S)}$ contains the origin. W.l.o.g. we assume that* $\mathrm{mvar}(t_n) = X_n$ *holds. Let $\mathcal{M}$ be the $2 \times n$ matrix with the vector $(X_1, \ldots, X_n)$ as first row and the gradient vector $\nabla t_n = \left(\frac{\partial t_n}{\partial X_1} \;\cdots\; \frac{\partial t_n}{\partial X_n}\right)$ as second row. Then, there exists a non-empty set $O \subset D_\rho^* \cap Z_\mathbb{R}(S)$, which is open relatively to $Z_\mathbb{R}(S)$, such that $\mathcal{M}$ is full rank at any point of O, and the origin is in the closure of O.*

PROOF. Consider first the case $d = n-1$, that is, T consists of the single polynomial t_n. Assume that there exists $0 < r < \rho$ such that $\mathcal{M}$ is not full rank at any point of $D_r^* \cap Z_\mathbb{R}(S)$. Then, the system of partial differential equations (PDEs) $X_j \frac{\partial t_n}{\partial X_i} - X_i \frac{\partial t_n}{\partial X_j} = 0$, for $1 \leq i < j \leq n$ holds at any point of $D_r^* \cap Z_\mathbb{R}(S)$. Lemma 35 implies that $t_n = u(X_1^2 + \cdots + X_n^2)$ where u is a univariate polynomial. Let $0 < r' < r$ and $S_{r'}$ be the sphere centred at the origin, with radius r'. Our hypotheses and the previous PDE argument imply that $Z_\mathbb{R}(S)$ contains a non-empty set $V_{r'} \subset S_{r'}$ which is an open set in the Euclidean topology induced on $S_{r'}$. The fact that this holds for all r', with $0 < r' < r$, contradicts the fact that $Z_\mathbb{R}(S)$ is a semi-algebraic set of dimension less than n.

Therefore, for all $r > 0$ small enough, the set $D_r^* \cap Z_\mathbb{R}(S)$ contains a point p_r, as well as a neighbourhood N_r of p_r (due to the full rank characterization in terms of minors) such that N_r is open relatively to $Z_\mathbb{R}(S)$ and $\mathcal{M}$ is full rank at any point of N_r. From there, the desired conclusion follows. □

8.4 Main Algorithm

We describe in this section our procedure for determining the existence and the possible value of limits of the form $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q(x_1, \ldots, x_n)$. Recall that q is a rational function in the n ordered variables $X_1 < \cdots < X_n$ and with rational number coefficients. We assume that the

origin is an isolated zero of the denominator. The pseudo-code for this procedure is stated in Algorithms 20, 19, 18 and 17.

Algorithm 17: LimitAlongCurve

Input: $q \in \mathbb{Q}(X_1, \ldots, X_n)$; real curve C given by regular chain T.
Output: Limit of rational function q at origin along real curve C.

begin
  Let f, g be the numerator and denominator of q;
  Let $R := \{gX_{n+1} - f\} \cup T$ with X_{n+1} a new variable;
  `// R is a regular chain for` $X_1 < \cdots < X_{n+1}$`;`
  Compute the limit points of $Z_{\mathbb{R}}(R) \setminus Z_{\mathbb{R}}(h_R)$ in $\mathbb{R}^n$ for the Euclidean topology;
  If there is only one such point $(x_1, \ldots, x_n, x_{n+1})$ with $x_1 = \cdots = x_n = 0$, then x_{n+1} is the desired limit of q;
  Return `no_limit` since q has no limit along C at the origin;
end

Algorithm 18: RandomEllipse

Input: $n \in \mathbb{N}$
Output: Ellipse in $\mathbb{R}^n$ randomly generated

begin
  repeat
    choose $A_1, \ldots, A_n, r$ randomly with $r > 0$;
    let $E_r := \sum_{i=1}^{n} A_i X_i^2 - r^2$;
    $\mathcal{S} := \begin{bmatrix} \frac{\partial E_r}{\partial X_1} & \cdots & \frac{\partial E_r}{\partial X_n} \\ X_1 & \cdots & X_n \end{bmatrix}$;
  until *$\mathcal{S}$ has at least one non-zero minor* ;
  return $(A_1, \ldots, A_n, r)$;
end

Proposition 24. *Algorithm 20 terminates and returns finitely many pairs $(L_1, S_1), \ldots, (L_e, S_e)$ where each of $L_1, \ldots, L_e$ is either a real number, or the flag* `no_limit`*, and $S_1, \ldots, S_e$ are all regular semi-algebraic systems such that each of the sets $Z_{\mathbb{R}}(S_1), \ldots, Z_{\mathbb{R}}(S_e)$ has dimension one and contains the origin in its closure. Moreover, if $L_i \in \mathbb{R}$, then L_i is the limit of q at the origin along the $Z_{\mathbb{R}}(S_i)$ for all $i = 1, \ldots, e$. In addition, the rational function q admits a finite limit at the origin if and only if $L_1, \ldots, L_e$ are all real and equal; if this holds, then this common value is the limit of q at the origin.*

Proof. Algorithm 20 applies Lemma 33 as follows. At Line (1), it computes the real algebraic set $\chi(q)$ defined in Notation 4 and at Line (2) computes a triangular decomposition $\mathcal{D}$ of $\chi(q)$ as defined in Proposition 23. Following Lemma 34, each regular semi-algebraic system $S \in \mathcal{D}$ which is zero-dimensional, or such that the origin is not contained in the closure of $Z_{\mathbb{R}}(S)$, is discarded at Line (6). For all the other regular semi-algebraic systems (RSASs), one runs Limitinner(q, S) at Line (7), that is, makes a call to Algorithm 19.

Algorithm 19 is the core routine. It first checks whether $Z_{\mathbb{R}}(S)$ has dimension one or not. If $\dim(Z_{\mathbb{R}}(S)) = 1$ holds, one runs $\mathsf{LimitAlongCurve}(q, S)$, that is, Algorithm 17. Applying Algorithm 19 with RSASs of dimension one can be seen as the *base case* of that recursive routine while the rest of that routine reduces computation with RSASs of dimension higher than one to the one-dimensional case.

This reduction is performed by repeated applications of the Lagrange multipliers trick, as in the proof of Lemma 33. It follows from Lemma 36 that there exists a minor $m \in \mathsf{Minors}(\mathcal{M})$ (where $\mathcal{M}$ is defined at Line (6) of Algorithm 19) such that we have $Z_{\mathbb{R}}(S) \not\subseteq Z_{\mathbb{R}}(m)$. Note that we know that the Jacobian of T (that is, the matrix formed with the ∇t, for $t \in T$) is full rank. This follows from Proposition 22 and explains why we do not need to compute any singular loci.

Once such a minor $m \in \mathsf{Minors}(\mathcal{M})$ is found, we compute $Z_{\mathbb{R}}(S) \cap Z_{\mathbb{R}}(\{m \neq 0\})$ using the $\mathsf{Intersect}$ command defined in Remark 19; this is done at Line (12). The resulting triangular decomposition consists of RSASs with the same dimension as S. The goal of Line (13) is to remove any RSAS S' such that $\overline{Z_{\mathbb{R}}(S')}$ does not contain the origin; see also Remark 19 for that test.

At Lines (16) to (19), we prepare for applying the Lagrange multipliers trick: since $\nabla_{(X_1,\ldots,X_n)} q$ is proportional to $(X_1, \ldots, X_n)$ along $\chi(q)$ we cannot re-use the family of circles C_r as in the proof of Lemma 33; instead, we use a family of ellipsoids, given by E_r; this idea was introduced in [98]. In particular, at Line (16), we determine values for the coefficients $A_1, \ldots, A_n$ of the polynomial E_r such that at least one minor of $\mathcal{M}'$ is not zero. This task is delegated to Algorithm 18: in practice, choosing $A_1, \ldots, A_n$ all positive at random works; if this would not work, $A_1, \ldots, A_n$ would be determined by solving polynomial systems.

The for-loop located between Lines (21) and (43) runs until we find a minor m' of the matrix $\mathcal{M}'$ (where $\mathcal{M}'$ is defined at Line (18) of Algorithm 19) such that the dimension of $Z_{\mathbb{R}}(S) \cap Z_{\mathbb{R}}(\{m' = 0\})$ is less than that of $Z_{\mathbb{R}}(S)$. This search is expected to be *successful* because the non-linear programs consisting of minimizing/maximizing $q(x_1, \ldots, x_n)$ under the constraints $(x_1, \ldots, x_n) \in Z_{\mathbb{R}}(S) \cap \{E_r = 0\}$ have solutions, necessarily. However, this search depends on the minor m, as well. In fact, what the previous non-linear optimization argument guarantees is the existence of a pair of minors (m, m') such that $\mathcal{M}$ is full rank for $m \neq 0$ while $\mathcal{M}'$ is not full rank for $m' = 0$. For this reason, for certain m, the search for m', or the recursive call at Line (31), may fail. Such a situation leads the algorithm to try the next m from $\mathsf{Minors}(\mathcal{M})$. It follows that Algorithm 19 must implement a *backtracking* mechanism.

This backtracking feature is achieved by enhancing the algorithm with a *state machine*. Note that at Lines (7), (20), (32), (40), (45), and (50) a variable called state is assigned in order to record the *new* state of the algorithm. Observe that, if the variable state never receives the value `backtrack` during the execution of Algorithm 19, then only the first minor $m \in \mathsf{Minors}(\mathcal{M})$ and the first minor $m' \in \mathsf{Minors}(\mathcal{M}')$ are considered by the algorithm.

Observe that, during one iteration of the for-loop located between Lines (21) and (43), if the variable L receives the value `backtrack`, or the variable $\mathcal{I}$ remains empty, then this iteration failed to find a minor m'; as a consequence either this for-loop goes for another iteration, or, if all iterations have been executed, the variable state will receive the value `backtrack` implying that the current value of the minor m cannot lead to find a minor m' with the desired properties.

Finally, observe that the execution of the for-loop located between Lines (8) and (54) terminates either with state reaching the value `found_second_minor` (implying that a pair of minors

(m, m') with the desired properties has been found) or with state having the value `backtrack` (implying that no such pair was found).

It follows from the above discussion that Algorithm 19 always terminates and so does Algorithm 20. Moreover, and as mentioned above, since the non-linear programs consisting of minimizing/maximizing q under the constraints $(x_1, \ldots, x_n) \in Z_{\mathbb{R}}(S) \cap \{E_r = 0\}$ necessarily have solutions (where q and S are the input of Algorithm 19), the calls that Algorithm 20 makes to Algorithm 19 will ultimately produce an answer of the form $(L_1, S_1), \ldots, (L_e, S_e)$ with the desired properties. □

Algorithm 19: LimitInner

Input: $q \in \mathbb{Q}(X_1, \ldots, X_n)$; a given regular semi-algebraic system S.
Output: Limit of rational function q at origin along zero set of S.
begin

```
1: procedure LimitInner(q, S)
2:     if dim(Z_R(S)) = 1 then
3:         return (LimitAlongCurve(q, S), S);
4:     end if
5:     let [Q, T, P_>] := S;
6:     M := [ X_1 ··· X_n ; ∇t, t ∈ T ];
7:     state := search_first_minor;
8:     for m ∈ Minors(M) do
9:         if Z_R(S) ⊆ Z_R(m) then next;
10:        end if
11:        J := ∅;
12:        for S' ∈ Intersect(S, {m ≠ 0}) do
13:            if o ∉ closure(Z_R(S')) or dim(Z_R(S')) = 0
    then
14:                next;
15:            end if
16:            (A_1, ..., A_n, r) := RandomEllipse(n);
17:            let E_r := Σ_{i=1}^n A_i X_i^2 − r^2;
18:            M' := [ ∂E_r/∂X_1 ··· ∂E_r/∂X_n ; X_1 ··· X_n ; ∇t, t ∈ T ];
19:            let [Q', T', P_>'] := S';
20:            state := search_second_minor;
21:            for m' ∈ Minors(M') do
22:                if res(m', T') = 0 then
23:                    next;
24:                end if
25:                I := ∅;
26:                for C ∈ Intersect(S', m' = 0) do
27:                    if o ∉ closure(Z_R(C)) or
    dim(Z_R(C)) = 0 then
28:                        next;
29:                    end if
30:                    L := LimitInner(q, C);
31:                    if L = backtrack then
32:                        state := backtrack;
33:                        break;
34:                    else
35:                        I := I ∪ {L};
36:                    end if
37:                end for
38:                if I ≠ ∅ and state ≠ backtrack
    then
39:                    J := J ∪ I;
40:                    state := found_second_minor;
41:                    break;
42:                end if
43:            end for
44:            if state ≠ found_second_minor then
45:                state := backtrack;
46:                break;
47:            end if
48:        end for
49:        if state ≠ backtrack then
50:            state := found_first_minor;
51:            break;
52:        end if
53:    end for
54:    if state = found_first_minor then
55:        return J;
56:    else
57:        return backtrack;
58:    end if
59: end procedure
```

end

Algorithm 20: Limit

Input: $q \in \mathbb{Q}(X_1, \ldots, X_n)$.
Output: Limit of rational function q at origin.

```
1 begin
2     A := Minors( [ X_1  ...  X_n ; ∂q/∂X_1  ...  ∂q/∂X_n ] );
3     D := RealTriangularize(A);
4     L := ∅;
5     for S ∈ D do
6         if o ∈ closure(Z_R(S)) and dim(Z_R(S)) > 0 then
7             L := L ∪ {LimitInner(q, S)};
8     return L;
9 end
```

8.5 Optimizations

In this section, we discuss how we apply different optimizations on the implementation of Algorithm 20. Two optimizations will be discussed here. The first optimization is given by Lemma 37; see also the Proposition 2.2 of [104].

Lemma 37. *Let f and g be two non-zero polynomials in $\mathbb{R}[X_1, \ldots, X_n]$, and also consider the lexicographic order $X_1 < \cdots < X_n$ on variables. If $\lim_{x \to \underline{o}} \frac{f}{g}$ exists, then the trailing monomial of f is not lower than the trailing monomial of g with respect to the order $X_1 < \cdots < X_n$.*

Lemma 37 implies that if the trailing monomial of f is lower than the trailing monomial of g with respect to any lexicographic variable ordering over $X_1, \ldots, X_n$, then it is guaranteed that $\lim_{x \to \underline{o}} \frac{f}{g}$ does not exist.

The second optimization trick is described in the next lemma. This lemma is taken from [89], Theorem 2.

Lemma 38. *Let $a_1, \ldots, a_n$ be non-negative integers, $m_1, \ldots, m_n$ be positive inetgers and $c_1, \ldots, c_n$ be positive real numbers, where $n > 1$. Then*

$$\lim_{(x_1,\ldots,x_n) \to (0,\ldots,0)} \frac{x_1^{a_1} \cdots x_n^{a_n}}{c_1\, x_1^{2m_1} + \cdots + c_n x_n^{2m_n}} \tag{8.5}$$

exists and equals zero if and only if $\sum_{i=1}^{N} \frac{a_i}{2\, m_i} > 1$.

Since the criterion of Lemma 38 can be applied to a very small set of examples, we have extended this idea and proposed Theorem 18.

Theorem 18. *Let f and g be two polynomials in $\mathbb{R}[X_1, \ldots, X_n]$ and g is of the form $c_1 X_1^{2m_1} + \cdots + c_n X_n^{2m_n}$ for some positive inetgers $m_1, \ldots, m_n$ and positive real numbers $c_1, \ldots, c_n$. Let also $r, q \in \mathbb{R}[X_1, \ldots, X_n]$ be the remainder and quotient in the Euclidean division of f by g w.r.t. X_1. Then $\lim_{(x_1,\ldots,x_n) \to (0,\ldots,0)} \frac{f(x_1,\ldots,x_n)}{g(x_1,\ldots,x_n)}$ exists and equals to L if for each term $X_1^{a_1} \cdots X_n^{a_n}$ (where*

$a_1, \ldots, a_n$ are non-negative integers) of the polynomial $r(X_1, \ldots, X_n)$, we have $\sum_{i=1}^{N} \frac{a_i}{2m_i} > 1$ and also $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q = L$.

Theorem 18 gives a sufficent condition for $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} \frac{f(x_1,\ldots,x_n)}{g(x_1,\ldots,x_n)}$ to exist when g is of the form $c_1 X_1^{2m_1} + \cdots + c_n X_n^{2m_n}$. However, this condition is not necessary, on the contrary to that of Lemma 38. In fact, when the condition of $\sum_{i=1}^{N} \frac{a_i}{2m_i} > 1$ does not hold for some terms of $r(X_1, \ldots, X_n)$, then we can not claim whether or not $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} \frac{f(x_1,\ldots,x_n)}{g(x_1,\ldots,x_n)}$ exist.

8.6 Limit of a multivariate rational function at a point which is not an isolated zero of the denominator

In this section, we discuss how to compute the limit of multivariate rational functions at the origin, when the origin is not an isolated zero of the denominator.

Let again $f, g \in \mathbb{Q}[X_1, \ldots, X_n]$ such that the fraction $q := f/g$ is irreducible and not constant. Let $Z_{\mathbb{R}}(f) := \{(x_1, \ldots, x_n) \in \mathbb{R}^n \mid f(x_1, \ldots, x_n) = 0\}$. Similarly, we define $Z_{\mathbb{R}}(g)$. We assume that $\underline{o} := (0, \ldots, 0) \in Z_{\mathbb{R}}(f) \cap Z_{\mathbb{R}}(g)$ holds. Let $C_{f,\underline{o}}$ and $C_{g,\underline{o}}$ be the connected components of $Z_{\mathbb{R}}(f)$ and $Z_{\mathbb{R}}(g)$ to which $\underline{o}$ belongs. Assume $\dim(C_{g,\underline{o}}) > 0$.

Over $\mathbb{C}$, we have either $W(f) \cap W(g) = \emptyset$ or $\dim(W(f) \cap W(g)) = \dim(W(f)) - 1$, while this phenomenon may break over $\mathbb{R}$.

Example 26. *Consider $f := (X - Y)^2 + (X^2 + Z^2T^2)^2$ and $g := (X - Y)^2 + (Y^2 + Z^2T^2)^2$. Note that f, g are different and irreducible and the euqality $C_{g,\underline{o}} = C_{f,\underline{o}}$ holds for them.*

Example 27. *Consider $f := (X - Y)^2 + (Z - Y)^2$ and $g := (X - Y)^2 + (Z - X)^2$. Then $C_{g,\underline{o}} \cap C_{f,\underline{o}}$ consists of a single point, which is the origin;*

Computing rational function limits often reduces to path tracking whithin semi-algebraic sets. The following lemma gives an interesting result about selecting a real analytic curve going through the origin and lying in a given semi-algebraic set.

Lemma 39 (Curve selection Lemma 3.1,[64])**.** *Let $f_1, \ldots, f_m, g_1, \ldots, g_p \in \mathbb{Q}[X_1, \ldots, X_n]$ such that the origin $\underline{o}$ is in the closure of the semi-algebraic set S defined by:*

$$f_1 = \cdots = f_m = 0,\ g_1 > 0, \ldots, g_p > 0.$$

Then, there exists real analytic curve $\gamma : [0, \varepsilon) \to \mathbb{R}^n$, with $\gamma(0) = \underline{o}$, and $\gamma(t) \in S$ for $t > 0$.

Remark 20. *For a semi-algebraic set S, testing $\underline{o} \in \overline{S}$ can be phrased as a quantifier elimination problem and thus solved by Cylindrical Algebraic Decomposition (CAD):*

$$\underline{o} \in \overline{S} \iff (\forall \varepsilon > 0)\,(\exists \mathbf{x} \in \mathbb{R}^n)\ \|\mathbf{x}\| < \varepsilon \Longrightarrow \mathbf{x} \in S.$$

For $n = 2$, one can use "lighter" methods for this test. For instance, computing the real branches (thus Puiseux series, which form an ordered field) of $f(x_1, x_2) = 0$ about $(x_1, x_2) = (0, 0)$ and check which ones satisfy $g(x_1, x_2) > 0$.

In the following, we present two propositions that studies the limit of real multivariate rational functions at the origin, when $C_{g,\underline{o}} \subseteq C_{f,\underline{o}}$ does not hold.

Proposition 25. *Assume that* $\underline{o} \in \overline{\{g = 0, f > 0\}}$*, then* $\lim_{(x_1,\ldots,x_n)\to\underline{o}} \frac{f(x_1,\ldots,x_n)}{g(x_1,\ldots,x_n)}$ *can not be finite.*

PROOF. Assume by contradiction that $\lim_{(x_1,\ldots,x_n)\to\underline{o}} \frac{f(x_1,\ldots,x_n)}{g(x_1,\ldots,x_n)}$ exists and equals $\ell \in \mathbb{R}$. Let us fix $\varepsilon > 0$. Then, there exists $r > 0$ such that for all $\mathbf{x} \in B(\underline{o}, r)$, we have $\ell - \varepsilon \leq q(\mathbf{x}) \leq \ell + \varepsilon$. Thus, $q(\mathbf{x})$ is bounded on $B(\underline{o}, r)$. From the hypothesis, for all $r' > 0$, we can choose $\mathbf{y} \in B(\underline{o}, r') \cap \{g = 0, f > 0\}$. Using the continuity of f and making r' small enough, we have $B(\mathbf{y}, r') \cap C_{f,\underline{o}} = \emptyset$ as well as $B(\mathbf{y}, r') \subseteq B(\underline{o}, r)$. Observe that $1/(g(\mathbf{x}))$ is arbitrary large (in absolute value) on $B(\mathbf{y}, r')$ while $f(\mathbf{x})$ remains bounded on $B(\mathbf{y}, r')$. This contradicts the fact that $q(\mathbf{x})$ is bounded on $B(\underline{o}, r)$. □

Proposition 26. *Assume that both* $\underline{o} \in \overline{\{f = 0, g > 0\}}$ *and* $\underline{o} \in \overline{\{g = 0, f > 0\}}$ *hold. Then,* $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} \frac{f(x_1,\ldots,x_n)}{g(x_1,\ldots,x_n)}$ *does not exist.*

PROOF. In one hand, from the first assumption and Lemma 39, there exists a path to the origin along which q is identically zero. Hence, $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q(x_1, \ldots, x_n)$ must be null, if it exists. On the other hand, the assumption $\underline{o} \in \overline{\{g = 0, f > 0\}}$ and Proposition 25 imply that $\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q(x_1, \ldots, x_n)$ cannot be finite. □

The results that are obtained from Proposition 25 and Proposition 26 imply that

$$\lim_{(x_1,\ldots,x_n)\to(0,\ldots,0)} q(x_1, \ldots, x_n)$$

does not have any finite limit, when $\underline{o} \in \overline{\{f \neq, g = 0\}}$. Assume from now on that $C_{g,\underline{o}} \subseteq C_{f,\underline{o}}$ holds. Let E be a connected component of $\mathbb{R}^n \setminus C_{f,\underline{o}}$. Theorem 2.1 in [54], suggests that one can apply a multivariate version of L'Hospital's Rule to compute

$$\lim_{\substack{(x_1,\ldots,x_n)\to(0,\ldots,0)\\(x_1,\ldots,x_n)\in E}} \frac{f(x_1, \ldots, x_n)}{g(x_1, \ldots, x_n)},$$

which might cause a recursive call to this procedure. Note that multivariate versions of L'Hospital's Rule have more assumptions than their univariate counterpart. What to do when these assumptions are not met is work in progress.

Theorem 19 (Theorem 10, [106])**.** *Let* $U \subseteq \{(x, y) \in \mathbb{R}^2 \mid 0 \leq x \text{ and } 0 \leq y\}$ *be a neighbourhood of* $\underline{o}$ *such that* $[0, \varepsilon]\times[0, \varepsilon] \subseteq U$ *holds for* $\varepsilon > 0$ *small enough. Let* $f, g : U \to \mathbb{R}$ *such that the partial derivatives* f_{xy} *and* g_{xy} *exist on* $]0, \varepsilon]\times]0, \varepsilon]$*. Assume* $(\lim_{\underline{o}} f_{xy}, \lim_{\underline{o}} g_{xy}) \notin \{(0, 0), (\pm\infty, \pm\infty)\}$*, that is, no indeterminate forms. Then we have*

$$\lim_{\substack{(x,y)\to(0,0)\\(x,y)\in U}} \frac{f(x, y)}{g(x, y)} = \lim_{\substack{(x,y)\to(0,0)\\(x,y)\in U}} \frac{f_{xy}(x, y)}{g_{xy}(x, y)}.$$

Example 28. *Consider* $f(x, y) = x\,y^2 - y$ *and* $g(x, y) = xy - x^3$*. Then one can obtain* $Z_{\mathbb{R}}(f) = \{xy - 1\} \cup \{y = 0\}$ *and* $Z_{\mathbb{R}}(g) = \{x^2 + y = 0, y < 0\} \cup \{x = 0\}$ *by using the command* `RealTriangularize` *of* RegularChains *library. Then we have:*

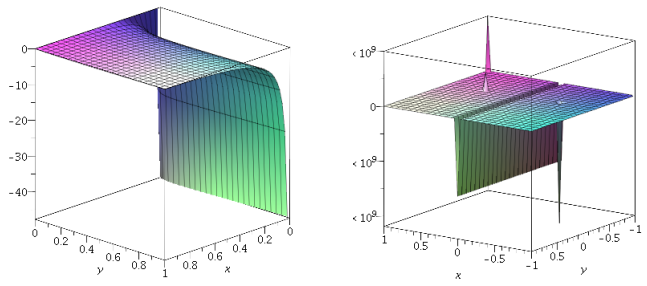


Figure 8.1: The graph corresponding to $\frac{f(x,y)}{g(x,y)} = z$

$$\lim_{\substack{(x,y)\to(0,0)\\(x,y)\in U}} \frac{f(x,y)}{g(x,y)} = \lim_{\substack{(x,y)\to(0,0)\\(x,y)\in U}} \frac{f_{xy}(x,y)}{g_{xy}(x,y)} = \lim_{\substack{(x,y)\to(0,0)\\(x,y)\in U}} \frac{2y}{1} = 0.$$

Based on Theorem 19, the value 0 *is the limit of* $\lim_{(x,y)\to(0,0)} \frac{f(x,y)}{g(x,y)}$ *over the quadrant* $x > 0, y > 0$ *and consequently, the global limit, see Figure 8.1.*

The following theorem is taken from [54], Theorem 2.1. This theorem is the multivariate counterpart of L'hôpital rule in one variable. The method of L'hôpital rule is a way to remove the indetermination that might occur when computing $\lim_{x\to o} \frac{f}{g}$.

Theorem 20 (Bivariate L'hôpital rule)**.** *Let* $\mathcal{N}$ *be a neighbourhood in* $\mathbb{R}^2$ *containing a point* p *at which two differentiable functions* $f : \mathcal{N} \to \mathbb{R}$ *and* $g : \mathcal{N} \to \mathbb{R}$ *are zero. Set* $C := \{(x, y) \in \mathcal{N} : f(x, y) = g(x, y) = 0\}$*, and suppose that* C *is a smooth curve through* p*. Suppose there exists a vector* $\boldsymbol{v}$ *not tangent to* C *at* p *such that the directional derivative* $D_{\boldsymbol{v}}(g)$ *of* g *in the direction of* $\boldsymbol{v}$ *is never zero within* $\mathcal{N}$*. More generally, if* C *consists of a union of two or more smooth curves through* p*, suppose that for each component* E_i *of* $\mathcal{N} \setminus C$ *we can find a vector* $\boldsymbol{v_i}$*, not tangent at* p *to any of the curves comprising* C *such that* $D_{\boldsymbol{v_i}}(g) \neq 0$ *on* E_i*. Then* $\lim_{(x,y)\to p} \frac{f(x,y)}{g(x,y)}$ *exists if the limits* $\lim_{(x,y)\to p,(x,y)\in E_i} \frac{D_{\boldsymbol{v_i}}(f)}{D_{\boldsymbol{v_i}}(g)}$ *exist and are equal for all* i*. Each limit is assumed to be taken over the domain of points where the denominator is nonzero, and we assume in each case that* p *is a limit point of that domain.*

Let $g(x, y) = 0$ be an analytic curve for which $\mathbb{R}^2 \setminus \{g(x, y) = 0\}$ consists two semi-algebraic sets E_1 and E_2, see Figure 8.2. Then, Theorem 20, Proposition 25 and Proposition 26, imply that for computing $\lim_{(x,y)\to p} \frac{f(x,y)}{g(x,y)}$, it is enough to compute $\lim_{(x,y)\to p} \frac{f(x,y)}{g(x,y)}$ within E_1 and E_2.

The same theorem holds for real-valued functions of n variables, with C a union of hypersurfaces.

Theorem 21 (Multivariate L'hôpital rule)**.** *Let* f *and* g *be two polynomials in* $\mathbb{R}[X_1, \ldots, X_n]$ *and the origin is a non-isolated zero of* $Z_{\mathbb{R}}(g)$*. Let also* $\dim(Z_{\mathbb{R}}(g)) = n - 1$ *and* $Z_{\mathbb{R}}(g)$ *is smooth*

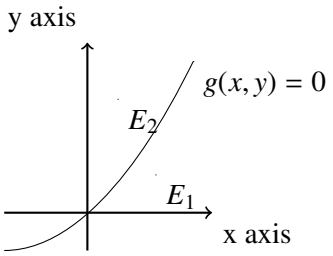


Figure 8.2: Limit of real bivariate rational functions

at the origin. Suppose that $E_1, \ldots, E_e$ *are the connected components of* $\mathbb{R}^n \setminus Z_{\mathbb{R}}(g)$*. For every vector* $\boldsymbol{v}$ *such that* $]0, \boldsymbol{v})$ *intersects* E_i *on a segment, we assume that there exists* $r > 0$ *such that* $D_{\boldsymbol{v}}(g)(\mathbf{x}) \neq 0$ *holds in* $E_i \cap B(\underline{o}, r)$*. Then we have:*

$$\lim_{(x_1,\ldots,x_n)\to p} \frac{f(x_1,\ldots,x_n)}{g(x_1,\ldots,x_n)} = \lim_{(x_1,\ldots,x_n)\to p,(x_1,\ldots,x_n)\in E_i} \frac{D_{\boldsymbol{v}_i}(f(x_1,\ldots,x_n))}{D_{\boldsymbol{v}_i}(g(x_1,\ldots,x_n))}.$$

Example 29. *Consider* $f(x, y) = x^2 - y^2$ *and* $g(x, y) = (x - y)^2 + z^2$*. Then one can compute* $Z_{\mathbb{R}}(f) = \{x = y\} \cup \{x = -y\}$ *and* $Z_{\mathbb{R}}(g) = \{x - y, z = 0\}$ *by using the command* `RealTriangularize` *in* RegularChains *library. Within* $\mathbb{R}^3 \setminus Z_{\mathbb{R}}(g)$*, consider the CAD cell* $E := \{x \neq y\}$*. Choose* $\boldsymbol{v} = (-1, 1, 0)$*, thus* $D_{\boldsymbol{v}}\, g = \boldsymbol{\nabla g} \cdot \boldsymbol{v} = -4x + 4y$*. Observe that* $D_{\boldsymbol{v}}\, g$ *does not vanish within* E*. Then, we have:*

$$\lim_{\substack{\mathbf{x}\to\underline{o}\\ \mathbf{x}\in E}} \frac{f(\mathbf{x})}{g(\mathbf{x})} = \lim_{\substack{\mathbf{x}\to\underline{o}\\ \mathbf{x}\in E}} \frac{D_{\boldsymbol{v}}\, f(\mathbf{x})}{D_{\boldsymbol{v}}\, g(\mathbf{x})} = \lim_{\substack{\mathbf{x}\to\underline{o}\\ \mathbf{x}\in E}} \frac{-2x-2y}{-4x+4y},$$

which implies that the latter limit, and consequently, $\lim_{\mathbf{x}\to\underline{o}} \frac{f(\mathbf{x})}{g(\mathbf{x})}$ *does not exist.*

8.7 Experimentation

This section is devoted to an experimental comparison of various MAPLE's codes for computing limits of multivariate rational functions: MAPLE's built-in command `limit`, the `TestLimit` command presented in [104] and our implementation of the algorithm of Section 8.4 within the `RationalFunctionLimit` command of the `RegularChains` library. We used more than 50 test-examples[1] and a representative subset of them is provided in Table 8.1. The abbreviations LM, TL, and RFL stand for `limit`, `TestLimit`, and `RationalFunctionLimit` commands. Further, NV, TD and LV represent the number of variables, the maximum total degree between numerator and denominator, and the value of the limit, respectively. The timings in columns LM, TL, RFL are in seconds.

For bivariate rational functions (examples 1-5), both LM and TL run faster than RFL, except on example 2. Recall that LM applies to bivariate rational functions only.

[1]The list of the examples and the timings corresponding to Table 8.1 can be found at `www.regularchains.org/RationalLimit/RationalFunctionLimit.zip` and `www.regularchains.org/RationalLimit/Report`, respectively.

Out of the 25 examples in 3 variables or more, TL and RFL solve respectively 9 and 24 examples within the prescribed resource limits of 2 GB of memory and 1800 sec of CPU time. Moreover, out of those 25 examples, TL fails on 8 of them due to a division-by-zero error. For the 17 examples in 3 variables or more, for which TL does not hit such an error, RFL runs faster than TL on 10 examples.

Taking into account the 30 examples: (1) TL and RFL solve respectively 13 and 29 examples, (2) for the 21 examples for which TL does not hit an error, RFL runs faster than TL 11 times, and (3) for the 13 examples for which TL computes the answer, TL is faster than RFL on 10 times.

Using `RegularChains` library one could use `RationalFunctionLimit`$(q(X_1, \ldots, X_n), [X_1 = 0, \ldots, X_n = 0])$ to obtain the limit of q at the origin. These experimental results were obtained on an Ubuntu desktop (3.33GHz Intel Core i5 CPU, 3.7Gb total mem.).

8.8 Conclusion

We have presented a procedure for determining the existence and possible value of limits of n-variate rational function over $\mathbb{Q}$. Our work extends the articles [21] and [98] dedicated to $n = 2$ and $n = 3$, respectively. We rely on the theory of regular chains, which allows us to avoid computing singular loci and decompositions into irreducible components. Our main tool is the RealTriangularize algorithm. We have implemented our procedure within the RegularChains library and our code is available at `www.regularchains.org`.

Experimental results show that our code solves more test cases than the implementation of [104], in particular as variable number or total degree increases. Nevertheless, our code is still under development and many optimizations are planned, including borrowing techniques used in [104].

As in [21] and [98], we assume that the origin is an isolated zero of the denominator. However, relaxing this assumption is work in progress thanks to RealTriangularize and the ideas proposed in [55].

In Chapter5, we have presented an algorithm for determining the real branches of a space curve about one of its point. This is a core routine for computing limits of real multivariate rational functions as well as for addressing topological questions like whether a point belongs to the closure of a CAD cell. To this end, we revisited the Hensel-Sasaki construction and established properties of the Yun-Moses polynomials (see Chapter 3). In Section 8.6, we have presented several results for computing limits of real multivariate rational functions at the origin, when the origin is not an isolated zero of the denominator. The case of computing limits of real multivariate rational functions is work in progress.

Ex	NV	TD	LM	TL	RFL	LV
1	2	4	0.061	0.097	0.134	-1
2	2	4	0.056	wrong answer	0.126	-1
3	2	2	0.015	0.002	0.074	undefined
4	2	4	0.096	0.001	0.906	undefined
5	2	4	0.064	0.089	0.127	-1
6	3	5	N/A	0.508	5.278	0
7	3	8	N/A	$> 2GB$	$> 2GB$	0
8	3	18	N/A	10.422	0.166	0
9	3	18	N/A	0.502	0.128	0
10	4	4	N/A	0.002	1.498	undefined
11	4	2	N/A	0.003	0.205	undefined
12	4	4	N/A	0.002	1.495	undefined
13	4	5	N/A	$> 2GB$	1.991	0
14	4	21	N/A	$> 2GB$	0.545	0
15	4	6	N/A	$> 2GB$	0.526	0
16	5	19	N/A	$> 2GB$	0.376	0
17	5	4	N/A	2.705	1.205	0
18	6	6	N/A	Error	1.228	0
19	6	6	N/A	Error	1.399	undefined
20	6	18	N/A	Error	0.391	0
21	6	10	N/A	$> 2GB$	0.627	0
22	6	10	N/A	$> 2GB$	1.143	0
23	6	6	N/A	Error	3.497	0
24	7	6	N/A	0.002	0.012	undefined
25	8	5	N/A	$> 2GB$	5.445	0
26	8	9	N/A	Error	19.747	undefined
27	9	4	N/A	0.003	3.628	undefined
28	9	10	N/A	Error	36.959	0
29	9	5	N/A	Error	1.193	0
30	10	10	N/A	Error	7.093	0

Table 8.1: Comparisons between three different commands for computing the limit of real multivairate rational functions: limit, TestLimit, and RationalFunctionLimit.

Chapter 9

Conclusion

In this thesis, we have pursued four main different goals. Each Section 9.1, 9.2, 9.3, and 9.4 is dedicated to one of our goals. For each goal, we first illustrate the accomplishments of this thesis to meet that goal; then we explain what has remained unsolved for a future work w.r.t each goal; finally, we report on software presentations of our methods for solving the four main goals of the present thesis.

9.1 Computing limit points of quasi-components of regular chains

For computing limit points corresponding to regular chains, we have proposed two different methods: (1) computing limit points corresponding to regular chains via Puiseux series expansions, and (2) computing limit points corresponding to regular chains via changes of coordinates. The first method for computing such limits only works for regular chains in dimension one. This method is based on a theorem from [70], which relates the Euclidean and Zariski topologies. This theorem enables us to use analytic tools to solve our algebraic problem of computing limit points corresponding to regular chains.

The new techniques proposed by the second method for computing limit points of regular chains can handle cases where the results of our first method (based on Puiseux series expansions) could not apply. One of the main ideas of our new results is to use a linear change of coordinates to replace the input regular chain to one such that the computations of limit points corresponding to the input regular chain can be done by means of standard operations on regular chains. Nevertheless, our proposed techniques do not cover all possible cases and the problem of finding limit points of quasi-components of regular chains remains unsolved.

The algorithm for computing limit points corresponding to regular chains of dimension one is implemented in the `AlgebraicGeometryTools` subpackage [5] of the `RegularChains` library which is available at `www.regularchains.org`. The function that implements the latter algorithm is called LimitPoints. The command LimitPoints can compute both real and complex limit points corresponding to regular chains of dimension one.

We have also implemened the Palgie algorithm for computing the regular chain $\overline{C}$ for a given regular chain C under changes of variable orderings and linear changes of coordinates in

ChainTools subpackage of RegularChains library. The functions that implement the latter algorithms are respectively called ChangeOfOrder and ChangeOfCoordinates.

9.2 Computing Puiseux expansions of bivariate polynomials

One of the essential tools for computing limit points corresponding to regular chains of dimension one is Puiseux series expansions of bivariate polynomials. For computing the Puiseux expansions of bivariate polynomials, we rely on the extended Hensel construction (EHC) method. We have enhanced two steps of the EHC algorithm. First, we proposed a new method for computing the Yun-Moses polynomials using Wronskian matrices. For an input bivariate polynomial $F(X, Y)$ with coefficients in a field $\mathbf{k}$ and total degree d, we show that the Yun-Moses polynomials (needed when applying the EHC to $F(X, Y)$) can be computed within $O(d^3\,\mathsf{M}(d))$ operations in $\mathbf{k}$, where $n \longmapsto \mathsf{M}(n)$ is a (polynomial) multiplication time [99]. In addition, we exhibit a new strategy for performing the lifting steps so that the k-th lifting step of the EHC applied to bivariate polynomial $F(X, Y)$, can be computed within $O(k\,d\,\mathsf{M}(d)^2)$ operations in $\mathbf{k}$ (instead of $O(k^2\,d\,\mathsf{M}(d)^2)$ in a direct approach) or within $O(k\,d\,\mathsf{M}(d))$ operations in the algebraic closure of $\mathbf{k}$.

The EHC algorithm is used for factorizing univariate polynomials with power series coefficients. Our enhancement of the EHC computes all the linear factors in the factorization of such polynomials within $O(d^3\mathsf{M}(d) + k^2\,d\,\mathsf{M}(d))$ operations in $\mathbb{C}$. In [51], H.T. Kung and J.F. Traub propose a method for computing one of the linear factors of bivariate polynomials. Using the method of Kung and Traub, the estimated time for computing all linear factors of $F(X, Y)$ is within $O(d^2\,k\,\mathsf{M}(k))$ (resp. $O(d^2\,\mathsf{M}(k))$) operations over $\mathbb{C}$, using a linear (resp. quadratic) lifting scheme. Furthermore, by the improvments of the method of Kung and Traub, which is given by D. V. Chudnovsky and G. V. Chudnovsky in [28], one can compute all the branches of $F(X, Y)$ within $O(d^2\mathsf{M}(k))$ operations over $\mathbb{C}$.

The EHC currently uses a linear lifting scheme. The experimentation reported in Section 3.5 show that, for problems of practical interest, an EHC implementation can outperform counterparts based on the linear and quadratic lifting schemes of [51]. However, one interesting question can be how to have a quadratic lifting scheme for the EHC algorithm.

The EHC method and the algorithm of Kung and Traub are implemented in PowerSeries library. The functions for latter algorithms are respectively called ExtendedHenselConstruction and KungTraub. The PowerSeries library can be downloaded from www.regularchains.org.

9.3 Computing tangent cones of space curves at their singular points

For computing tangent cones of space curves at their singular points, we have proposed an algorithm based on computing the limit of the secant lines, which are the lines going through the given singular point and an arbitrary moving point on the given space curve. The computation of limit of such secant lines is possible through the method of computing limit points corresponding to regular chains of dimension one.

The function TangentCone implements this algorithm in the `AlgebraicGeometryTools` subpackage [5] of the `RegularChains` library.

9.4 Computing limits of real multivariate rational functions

We have proposed an algorithm for computing limits of real multivariate rational functions at the origin. We first assume that the origin is an isolated zero of the denominator of the given rational function. To do so, we take advantage of the theory of regular chains and the RealTriangularize algorithm for decomposing semi-algebraic systems. For computing limits of real multivariate rational functions, when the origin is not an isolated zero of the denominator, we describe several results including multivariate L'Hospital rule. However, the non-isolated case is still work in progress.

The RationalFunctionLimit implements our algorithm for computing the limit of real multivariate rational functions at the origin, when the origin is an isolated zero of the denominator. This function is integrated into `AlgebraicGeometryTools` subpackage of `RegularChains` library.

Bibliography

[1] S. S. Abhyankar. Irreducibility criterion for germs of analytic functions of two complex variables. *Advances in Mathematics*, 74(2):190 – 257, 1989.

[2] M.E. Alonso, T. Mora, G. Niesi, and M. Raimondo. An algorithm for computing analytic branches of space curves at singular points. In *Proc. of the 1992 International Workshop on Mathematics Mechanization*, pages 135–166. International Academic Publishers, 1992.

[3] P. Alvandi, M. Ataei, and M. Moreno Maza. On the extended Hensel construction and its application to the computation of limit points. In *Symbolic and Algebraic Computation, International Symposium (ISSAC 2017), Kaiserslautern,Germany*, 2017, preprint.

[4] P. Alvandi, C. Chen, A. Hashemi, and M. Moreno Maza. Regular chains under linear changes of coordinates and applications. In *Computer Algebra in Scientific Computing - 17th International Workshop, CASC 2015, Aachen, Germany, September 14-18, 2015, Proceedings*, pages 30–44. Springer, 2015.

[5] P. Alvandi, C. Chen, S. Marcus, M. Moreno Maza, É. Schost, and P. Vrbik. Doing algebraic geometry with the regularchains library. In *Mathematical Software - ICMS 2014 - 4th International Congress, Seoul, South Korea, August 5-9, 2014. Proceedings*, pages 472–479. Springer, 2014.

[6] P. Alvandi, C. Chen, and M. Moreno Maza. Computing the limit points of the quasi-component of a regular chain in dimension one. In *Proc. of CASC*, volume 8136, pages 30–45, 2013.

[7] P. Alvandi, M. Kazemi, and M. Moreno Maza. Computing limits with the `regularchains` and `powerseries` libraries: From rational functions to zariski closure. In Proceedings of International Symposium on Symbolic and Algebraic Computation (ISSAC 2016), Waterloo, Canada, 2016. Refereed software.

[8] P. Alvandi, M. Kazemi, and M. Moreno Maza. Computing limits of real multivariate rational functions. In *Proc. of ISSAC*, pages 39–46. ACM, 2016.

[9] P. Alvandi, M. Moreno Maza, É. Schost, and P. Vrbik. A standard basis free algorithm for computing the tangent cones of a space curve. In *Proc. of CASC*, pages 45–60. Springer, 2015.

[10] C. Andradas and T. Recio. Plotting missing points and branches of real parametric curves. *Appl. Algebra Eng. Commun. Comput.*, 18(1-2):107–126, 2007.

[11] P. Aubry, D. Lazard, and M. Moreno Maza. On the theories of triangular sets. *J. Symb. Comput.*, 28(1-2):105–124, 1999.

[12] C. L. Bajaj and A. V. Royappa. Finite representations of real parametric curves and surfaces. In *Modeling in Computer Graphics, Methods and Applications [selection of papers from the conference held at Genoa, Italy, on June 28-July 1, 1993]*, pages 347–358. Springer, 1993.

[13] E. Becker, T. Mora, M. G. Marinari, and C. Traverso. The shape of the Shape Lemma. In *Proc. of ISSAC'94*, pages 129–133. ACM, 1994.

[14] I. Bermejo and P. Gimenez. Saturation and Castelnuovo-Mumford regularity. *J. Algebra*, 303:592–617, 2006.

[15] L. Bernardin. On bivariate Hensel and its parallelization. In *Proc. of ISSAC*, pages 96–100. ACM, 1998.

[16] M. Bocher. The theory of linear dependence. *Annals of Mathematics, Second Series*, 2(1/4):81–96, 1900.

[17] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot. Representation for the radical of a finitely generated differential ideal. In *Proc. of ISSAC*, pages 158–166. ACM, 1995.

[18] F. Boulier, F. Lemaire, and M. Moreno Maza. Pardi! In *Proc. of ISSAC*, pages 38–47. ACM, 2001.

[19] F. Boulier, F. Lemaire, and M. Moreno Maza. Well known theorems on triangular systems and the D5 principle. In *Proc. of* Transgressive Computing 2006, pages 79–91, Granada, Spain, 2006.

[20] F. Boulier, F. Lemaire, and M. Moreno Maza. Computing differential characteristic sets by change of ordering. *J. Symb. Comput.*, 45(1):124–149, 2010.

[21] C. Cadavid, S. Molina, and J. D. Vélez. Limits of quotients of bivariate real analytic functions. *J. Symb. Comput.*, 50:197–207, 2013.

[22] C. Chen, J. H. Davenport, J. P. May, M. Moreno Maza, B. Xia, and R. Xiao. Triangular decomposition of semi-algebraic systems. *J. Symb. Comput.*, 49:3–26, 2013.

[23] C. Chen, J. H. Davenport, M. Moreno Maza, B. Xia, and R. Xiao. Computing with semi-algebraic sets: Relaxation techniques and effective boundaries. *J. Symb. Comput.*, 52:72–96, 2013.

[24] C. Chen, O. Golubitsky, F. Lemaire, M. Moreno Maza, and W. Pan. Comprehensive triangular decomposition. In *Proc. of CASC*, pages 73–101. Springer, 2007.

[25] C. Chen and M. Moreno Maza. Algorithms for computing triangular decomposition of polynomial systems. *J. Symb. Comput.*, 47(6):610–642, 2012.

[26] C. Chen, M. Moreno Maza, B. Xia, and L. Yang. Computing cylindrical algebraic decomposition via triangular decomposition. In *Proc. of ISSAC*, pages 95–102. ACM, 2009.

[27] S. C. Chou and X. S. Gao. A zero structure theorem for differential parametric systems. *J. Symb. Comput.*, 16(6):585–595, 1993.

[28] D. V. Chudnovsky and G. V. Chudnovsky. On expansion of algebraic functions in power and Puiseux series, I. *J. Complexity*, 2(4):271–294, 1986.

[29] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms*. Spinger-Verlag, 1st edition, 1992.

[30] D. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry*. Graduate Text in Mathematics, 185. Springer-Verlag, New-York, 1998.

[31] X. Dahan, X. Jin, M. Moreno Maza, and É. Schost. Change of order for regular chains in positive dimension. *Theor. Comput. Sci.*, 392(1-3):37–65, 2008.

[32] J. Della Dora, C. Dicrescenzo, and D. Duval. About a new method for computing in algebraic number fields. In *Proc. of EUROCAL' 85*, pages 289–290. Springer, 1985.

[33] D. Duval. Rational Puiseux expansions. *Compos. Math.*, 70(2):119–154, 1989.

[34] D. Eisenbud. *Commutative Algebra with a View toward Algebraic Geometry*. Springer-Verlag, New York, 1995.

[35] G. Fischer. *Plane Algebraic Curves*. American Mathematical Society, 2001.

[36] W. Fulton. *Algebraic curves*. Advanced Book Classics. Addison-Wesley, 1989.

[37] X. S. Gao, J. Van der Hoeven, C. M. Yuan, and G. L. Zhang. Characteristic set method for differential-difference polynomial systems. *J. Symb. Comput.*, 44(9):1137–1163, 2009.

[38] M. Giusti, K. Hägele, G. Lecerf, J. Marchand, and B. Salvy. The projective Nœther Maple package: Computing the dimension of a projective variety. *J. Symbolic Comput.*, 30(3):291–307, 2000.

[39] G. M. Greuel and G. Pfister. *A Singular Introduction to Commutative Algebra*. Springer-Verlag, Berlin, 2002.

[40] D. Gruntz. A new algorithm for computing asymptotic series. In *Proc. of ISSAC*, pages 239–244. ACM, 1993.

[41] D. Gruntz. *On computing limits in a symbolic manipulation system*. PhD thesis, ETZ Zurich, 1996.

[42] A. Hashemi. Efficient algorithms for computing Nœther normalization. In *Asian Symposium on Computer Mathematics (ASCM 2007)*, volume 5081 of *Lecture Notes in Artificial Intelligence*, pages 97–107. Springer, 2007.

[43] A. Hashemi. Effective computation of radical of ideals and its application to invariant theory. In *Proc. of ICMS*, volume 8592 of *Lect. Notes Comput. Sci.*, pages 382–389. Springer, 2014.

[44] D. L. Hilliker and E. G. Straus. Determination of bounds for the solutions to those binary diophantine equations that satisfy the hypotheses of Runge's theorem. *Trans. AMS*, 280(2):637–657, 1983.

[45] E. Hubert. Factorization-free decomposition algorithms in differential algebra. *J. Symb. Comput.*, 29(4-5):641–662, 2000.

[46] D. Inaba. Factorization of multivariate polynomials by extended Hensel construction. *SIGSAM Bull.*, 39(1):2–14, March 2005.

[47] D. Inaba and T. Sasaki. A numerical study of extended Hensel series. In *Symbolic-Numeric Computation, SNC 2007, International Workshop, 25-27 July 2007, University of Western Ontario, London, Ontario, Canada*, pages 103–109. ACM, 2007.

[48] M. Iwami. Analytic factorization of the multivariate polynomial. *Proc. of CASC*, pages 213–225, 2003.

[49] M. Kalkbrener. Algorithmic properties of polynomial rings. *J. Symb. Comput.*, 26(5):525–581, 1998.

[50] T. Krick and A. Logar. An algorithm for the computation of the radical of an ideal in the ring of polynomials. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (AAECC 1991)*, volume 539 of *Lect. Notes Comput. Sci.*, pages 195–205, 1991.

[51] H. T. Kung and J. F. Traub. All algebraic functions can be computed fast. *J. ACM*, 25(2):245–260, 1978.

[52] T. Kuo. Generalized NewtonPuiseux theory and Hensel's lemma in $\mathbb{C}[x, y]$. *Canad. J. Math.*, 41:1101–1116, 1989.

[53] S. Landau and G. L. Miller. Solvability by radicals is in polynomial time. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, STOC '83, pages 140–151. ACM, 1983.

[54] G. R. Lawlor. A L'Hospital's rule for multivariable functions. *ArXiv e-prints*, August 2012.

[55] D. Lazard and F. Rouillier. Solving parametric polynomial systems. *J. Symb. Comput.*, 42(6):636–667, 2007.

[56] G. Lecerf. Computing the equidimensional decomposition of an algebraic closed set by means of lifting fibers. *J. of Complexity*, 19(4):564–596, 2003.

[57] F. Lemaire, M. Moreno Maza, W. Pan, and Y. Xie. When does $\langle T \rangle$ equal sat(T)? *J. Symb. Comput.*, 46(12):1291–1305, 2011.

[58] F. Lemaire, M. Moreno Maza, and Y. Xie. The `RegularChains` library. In *Maple 10*, Maplesoft, Canada, 2005. Refereed software.

[59] Arjen K. Lenstra. Factoring multivariate polynomials over algebraic number fields. *SIAM Journal on Computing*, 16(3):591–598, 1987.

[60] A. Logar. A computational proof of the Noether normalization lemma. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (AAECC 1988)*, volume 357 of *Lect. Notes Comput. Sci.*, pages 259–273, 1988.

[61] B. Mannaa and T. Coquand. Dynamic Newton-Puiseux theorem. *arxiv::1304.6770v2*, 2013.

[62] S. Marcus, M. Moreno Maza, and P. Vrbik. On Fulton's algorithm for computing intersection multiplicities. In *Computer Algebra in Scientific Computing*, pages 198–211. Springer Berlin Heidelberg, 2012.

[63] J. Maurer. Puiseux expansion for space curves. *Manuscripta Math.*, 32:91–100, 1980.

[64] J.W. Milnor. *Singular Points of Complex Hypersurfaces*. Annals of mathematics studies. Princeton University Press, 1968.

[65] F. Mora. An algorithm to compute the equations of tangent cones. In Jacques Calmet, editor, *Computer Algebra*, volume 144 of *Lecture Notes in Computer Science*, pages 158–165. Springer Berlin Heidelberg, 1982.

[66] T. Mora, C. Traverso, and G. Pfister. An introduction to the tangent cone algorithm issues in robotics and non-linear geometry. *Advances in Computing Research*, 6:199–270, 1992.

[67] M. Moreno Maza and R. Rioboo. Polynomial gcd computations over towers of algebraic extensions. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (AAECC 1995)*, volume 948 of *Lect. Notes Comput. Sci.*, pages 365–382, 1995.

[68] M. Moreno Maza, B. Xia, and R. Xiao. On solving parametric polynomial systems. *Mathematics in Computer Science*, 6(4):457–473, 2012.

[69] J. Moses and D.Y.Y. Yun. The EZ-GCD algorithm. In *Proceedings of the ACM Annual Conference*, ACM '73, pages 159–166. ACM, 1973.

[70] D. Mumford. *The Red Book of Varieties and Schemes*. Springer-Verlag, 2nd edition, 1999.

[71] J. R. Munkres. *Topology*. Prentice Hall, 2nd edition, 2000.

[72] A. Parusiński and G. Rond. The Abhyankar-Jung theorem. *J. Algebra*, 365:29–41, 2012.

[73] A. Poteaux and M. Rybowicz. Improving complexity bounds for the computation of Puiseux series over finite fields. In *Proc. of ISSAC*, pages 299–306. ACM, 2015.

[74] Guénaël Renault and Kazuhiro Yokoyama. A modular method for computing the splitting field of a polynomial. In *Proceedings of the 7th International Conference on Algorithmic Number Theory*, ANTS'06, pages 124–140, Berlin, Heidelberg, 2006. Springer-Verlag.

[75] Guénaël Renault and Kazuhiro Yokoyama. Multi-modular algorithm for computing the splitting field of a polynomial. In *Symbolic and Algebraic Computation, International Symposium, ISSAC 2008, Linz/Hagenberg, Austria, July 20-23, 2008, Proceedings*, pages 247–254, 2008.

[76] R. Rioboo. Real algebraic closure of an ordered field: Implementation in *Axiom*. In *ISSAC*, pages 206–215. ACM, 1992.

[77] J. F. Ritt. *Differential Equations from an Algebraic Standpoint*, volume 14. American Mathematical Society, 1932.

[78] D. Robertz. Noether normalization guided by monomial cone decompositions. *J. Symb. Comput.*, 44:1359–1373, 2009.

[79] F. Rouillier. Solving zero-dimensional systems through the rational univariate representation. *Appl. Algebra Eng. Commun. Comput.*, 9(5):433–461, 1999.

[80] B. Salvy and J. Shackell. Symbolic asymptotics: Multiseries of inverse functions. *J. Symb. Comput.*, 27(6):543–563, 1999.

[81] T. Sasaki and D. Inaba. Enhancing the extended Hensel construction by using Gröbner bases. In *Proc. of CASC*, volume 9890, pages 457–472. Springer, 2016.

[82] T. Sasaki and F. Kako. Solving multivariate algebraic equation by Hensel construction. *Japan Journal of Industrial and Applied Mathematics*, 1999.

[83] T. Sasaki and S. Yamaguchi. An analysis of cancellation error in multivariate Hensel construction with floating-point number arithmetic. In *Proc. of ISSAC*, pages 1–8. ACM, 1998.

[84] W. M. Seiler. A combinatorial approach to involution and δ-regularity II: Structure analysis of polynomial modules with Pommaret bases. *Appl. Alg. Eng. Comm. Comp.*, 20:261–338, 2009.

[85] J. R. Sendra. Normal Parametrizations of Algebraic Plane Curves. *J. Symb. Comput*, 33(6):863 – 885", 2002.

[86] J. R. Sendra, D. Sevilla, and C. Villarino. Covering of surfaces parametrized without projective base points. In *Proc. of ISSAC*, pages 375–380. ACM, 2014.

[87] J. R. Sendra, D. Sevilla, and C. Villarino. Covering rational ruled surfaces. *CoRR*, abs/1406.2140, 2014.

[88] J. R. Sendra, C. Villarino, and D. Sevilla. Missing sets in rational parametrizations of surfaces of revolution. *Computer-Aided Design*, 66:55–61, 2015.

[89] A. S. Sertöz. Continuity of multivariate rational functions. *arXiv:1403.7434*, 2010.

[90] I. R. Shafarevich. *Basic algebraic geometry. 1*. Springer-Verlag, Berlin, second edition, 1994.

[91] T. Shimoyama and K. Yokoyama. Localization and primary decomposition of polynomial ideals. *J. Symb. Comput.*, 22(3):247–277, 1996.

[92] A. J. Sommese and J. Verschelde. Numerical homotopies to compute generic points on positive dimensional algebraic sets. *J. Complexity*, 16(3):572–602, 2000.

[93] B. M. Trager. Algebraic factoring and rational function integration. In *Proceedings of the Third ACM Symposium on Symbolic and Algebraic Computation*, SYMSAC '76, pages 219–226. ACM, 1976.

[94] K. Tsuji. An improved ez-gcd algorithm for multivariate polynomials. *J. Symb. Comput.*, 44(1):99–110, 2009.

[95] M. van Hoeij. An algorithm for computing an integral basis in an algebraic function field. *J. Symb. Comput.*, 18(4):353 – 363, 1994.

[96] I. B. Vapnyarskii. *Encyclopedia of Mathematics*, chapter Lagrange multipliers. Springer.

[97] W. V. Vasconcelos. *Computational Methods in Commutative Algebra and Algebraic Geometry*. Springer-Verlag, 1998.

[98] J. D. Vélez, J. P. Hernández, and C. A. Cadavid. Limits of quotients of real polynomial functions of three variables. *ArXiv e-prints*, April 2016.

[99] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, 2 edition, 2003.

[100] P. Vrbik. *Computing Intersection Multiplicity via Triangular Decomposition*. PhD thesis, The University of Western Ontario, 2014.

[101] R. J. Walker. *Algebraic Curves*. Springer-Verlag, 1978.

[102] D. K. Wang. The `Wsolve` package. http://www.mmrc.iss.ac.cn/~dwang/wsolve.html.

[103] D. M. Wang. Epsilon 0.618. http://www-calfor.lip6.fr/~wang/epsilon.

[104] S. J. Xiao and G. X. Zeng. Determination of the limits for multivariate rational functions. *Science China Mathematics*, 57(2):397–416, 2014.

[105] L. Yang, X. R. Hou, and B. Xia. A complete algorithm for automated discovering of a class of inequality-type theorems. *Science in China, Series F*, 44(1):33–49, 2001.

[106] W. H. Young. On indeterminate forms. In *Proc. London Math. Soc.*, volume 8, pages 40–76, 1910.

Curriculum Vitae

Name:	Parisa Alvandi
Post-Secondary Education and Degrees:	The University of Western Ontario London, Ontario, Canada Ph.D. Computer Science, May 2017
	Isfahan University Of Technology Khomeyni Shahr, Isfahan, Iran M.Sc. Pure Mathematics (Geometry), August 2011
	K. N. Toosi University of Technology Tehran, Tehran, Iran B.Sc. Pure Mathematics, August 2009
Related Work Experience:	Internship Maplesoft incorporation, Waterloo, Ontario, Canada July 2013 - October 2013
	Research and Teaching Assistant The University of Western Ontario 2012 - 2017
	Research Assistant Isfahan University Of Technology 2009 - 2011

Publications:

Parisa Alvandi, Masoud Ataei, Marc Moreno Maza.
On the extended Hensel construction and its application to the computation of limit points, ISSAC '17.

Parisa Alvandi, Mahsa Kazemi, Marc Moreno Maza.
Computing Limits of Real Multivariate Rational Functions ISSAC '16.

Parisa Alvandi, Mahsa Kazemi, Marc Moreno Maza.
Computing Limits with the RegularChains and PowerSeries libraries: From Rational Functions to Zariski Closure, ISSAC '16.

Parisa Alvandi, Marc Moreno Maza.
Real limit points of quasi-componenets of regular chains, ISSAC '16.

Parisa Alvandi, Changbo Chen, Amir Hashemi, Marc Moreno Maza.
Regular Chains under Linear Changes of Coordinates and Applications , CASC 2015.

Parisa Alvandi, Marc Moreno Maza, Éric Schost, Paul Vrbik.
A Standard Basis Free Algorithm for Computing the Tangent Cones of a Space Curve, CASC 2015.

Parisa Alvandi, Changbo Chen, Steffen Marcus,
Marc Moreno Maza, Éric Schost, Paul Vrbik.
Doing Algebraic Geometry with the RegularChains Library, ICMS 2014.

Parisa Alvandi, Changbo Chen, Marc Moreno Maza.
Computing the Limit Points of the Quasi-component of a Regular Chain in Dimension One , CASC 2013.

Parisa Alvandi, Amir Hashemi.
Applyling Buchberger's Criteria for Computing Gröbner Bases over Finite Chain Rings, Journal of Algebra and its Applications 2013.

Parisa Alvandi, Amir Hashemi.
Detecting Unnecessary Reductions in Gröbner Bases over Galois Rings , Proceeding of the 42nd Annual Iranian Mathematics Conference, 2011.

Software Packages:

`AlgebraicGeometryTools` subpackage of `RegularChains` library implemented in MAPLE, http://www.regularchains.org/

`PowerSeries` library
implemented in MAPLE, http://www.regularchains.org/